



**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ХЕРСОНСЬКИЙ НАЦІОНАЛЬНИЙ
ТЕХНІЧНИЙ УНІВЕРСИТЕТ**

**ПРОБЛЕМИ
ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ**

Рекомендовано до друку Вченою радою
Херсонського національного технічного університету
(протокол №5 від 30 березня 2018 року)

Журнал «Проблеми інформаційних технологій» включено до Переліку наукових фахових видань України (Наказ Міністерства освіти і науки України №820 від 11.07.2016 р.), в яких можуть публікуватися результати дисертаційних робіт на здобуття наукових ступенів доктора і кандидата наук

Журнал включено до науково-метричних баз даних,
цифрових архівів та бібліотек з безкоштовним on-line-доступом:
Index Copernicus, Google Scholar, Research Bible, Open Academic Journals Index (OAJI), Directory of Open Access Journals (DOAJ), РІНЦ (Російський індекс наукового цитування), AcademicKeys, National Library of Ukraine (Vernadsky)

ISSN 1998-7005 (Print)
ISSN 2313-0687 (Online)

#01(023) травень 2018

РЕДАКЦІЙНА РАДА**Головний редактор**

ХОДАКОВ Віктор Єгорович

/доктор технічних наук, професор, заслужений діяч науки і техніки України/

Заступник головного редактора

ЛЯШЕНКО Олена Миколаївна /кандидат технічних наук, доцент/

Відповідальний секретар

КИБАЛКО Ігор Іванович /кандидат технічних наук, доцент /

РЕДАКЦІЙНА КОЛЕГІЯ

Бардачов Юрій Миколайович засл. діяч науки і техніки України	д.т.н., професор	Херсонський національний технічний університет, Україна
Ходаков Віктор Єгорович засл. діяч науки і техніки України	д.т.н., професор	Херсонський національний технічний університет, Україна
Павлов Олександр Анатолійович засл. діяч науки і техніки України	д.т.н., професор	Національний технічний університет України «Київський політехнічний інститут», Україна
Хомченко Анатолій Никифорович засл. діяч науки і техніки України	д.ф.-м.н., професор	Чорноморський національний університет імені Петра Могили, Україна
Башков Євген Олександрович	д.т.н., професор	Донецький національний технічний університет, Україна
Бурков Володимир Миколайович	д.т.н., професор	Інститут проблем управління ім. В.О. Трапезнікова РАН, Російська Федерація
Бусигін Борис Сергійович	д.т.н., професор	Національний гірничий університет, Україна
Висоцький Юрій Борисович	д.х.н., професор	Донецький національний технічний університет, Україна
Guchek Petro Yosipovich	Dr.Sc., Professor	Institute of Biocybernetics and Biomedical Engineering of the Polish Academy of Sciences, Poland
Долятовський Валерій Анастасійович засл. діяч науки Російської Федерації	д.е.н., професор	Південний федеральний університет, Російська Федерація
Загарій Геннадій Іванович	д.т.н., професор	Українська державна академія залізничного транспорту, Україна
Каргін Анатолій Олексійович	д.т.н., професор	Донецький національний університет, Україна
Коваленко Віктор Федорович	д.т.н., професор	Херсонський національний технічний університет, Україна
Коваленко Ігор Іванович	д.т.н., професор	Чорноморський національний університет імені Петра Могили, Україна
Козак Юрій Олександрович	д.т.н., професор	Одеська національна академія зв'язку ім. О.С. Попова, Україна
Кошкін Костянтин Вікторович	д.т.н., професор	Національний університет кораблебудування ім. адмірала Макарова, Україна
Левикін Віктор Макарович	д.т.н., професор	Харківський національний університет радіоелектроніки, Україна
Лізунов Петро Петрович	д.т.н., професор	Київський національний університет будівництва та архітектури, Україна
Марасанов Володимир Васильович	д.т.н., професор	Херсонський національний технічний університет, Україна
Мельник Анатолій Олексійович	д.т.н., професор	Національний університет «Львівська політехніка», Україна
Михайленко Віктор Мефодійович	д.т.н., професор	Київський національний університет будівництва та архітектури, Україна
Пушкарь Олександр Іванович	д.е.н., професор	Харківський національний економічний університет, Україна
Рожков Сергій Олександрович	д.т.н., професор	Херсонська державна морська академія, Україна
Розов Юрій Георгійович засл. діяч науки і техніки України	д.т.н., професор	Херсонський національний технічний університет, Україна
Рудакова Ганна Володимирівна	д.т.н., професор	Херсонський національний технічний університет, Україна
Рябенський Володимир Михайлович	д.т.н., професор	Національний університет кораблебудування ім. адмірала Макарова, Україна
Садовський Віктор Васильович засл. працівник вищої школи Російської Федерації	д.т.н., професор	Білоруський державний економічний університет, Республіка Білорусь
Синекоп Микола Сергійович	д.т.н., професор	Харківський державний університет харчування та торгівлі, Україна
Філіпенко Ігор Григорович	д.т.н., професор	Харківська державна академія залізничного транспорту, Україна
Фісун Микола Тихонович	д.т.н., професор	Чорноморський державний університет ім. Петра Могили, Україна
Цвіркун Анатолій Данилович	д.т.н., професор	Інститут проблем управління ім. В.О. Трапезнікова РАН, Російська Федерація
Шаронова Наталія Валеріївна	д.т.н., професор	Національний технічний університет «Харківський політехнічний інститут», Україна
Шерстюк Володимир Григорович Lydia Lewin	д.т.н., доцент PhD	Херсонський національний технічний університет, Україна University of Melbourne, Australia



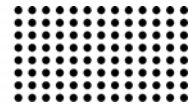
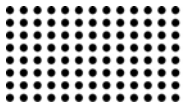
ЗАСНОВНИК

Херсонський національний технічний університет

м. Херсон

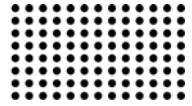
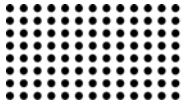
ПРИ ПІДТРИМЦІ:

Донбаської національної академії будівництва та архітектури	м. Макєєвка
Донецького національного технічного університету	м. Красноармійськ
Донецького національного університету	м. Вінниця
Європейського університету	м. Київ
Київського національного університету будівництва і архітектури	м. Київ
Чорноморського державного університету ім. П.Могили	м. Миколаїв
Національного авіаційного університету	м. Київ
Національної гірничої академії України	м. Дніпропетровськ
Національного технічного університету «Київський політехнічний інститут»	м. Київ
Національного технічного університету «Харківський політехнічний інститут»	м. Харків
Національного університету кораблебудування ім. адмірала Макарова	м. Миколаїв
Національного університету «Львівська політехніка»	м. Львів
Одеської національної академії харчових технологій	м. Одеса
Української державної академії залізничного транспорту	м. Харків
Харківської національної академії міського господарства	м. Харків
Харківського державного університету харчування та торгівлі	м. Харків
Харківського державного економічного університету	м. Харків
Харківського національного університету радіоелектроніки	м. Харків
Херсонського державного аграрного університету	м. Херсон
Херсонської державної морської академії	м. Херсон



ЗМІСТ

В. Е. Ходаков, А. Е. Соколов, Г. В. Веселовская ОБУЧЕНИЕ КАК ПРОЦЕСС ИЗМЕНЕНИЯ ПОСЛЕДОВАТЕЛЬНОСТИ СОСТОЯНИЙ ОБУЧАЕМОГО В СИСТЕМЕ "ОБУЧАЮЩИЙ-ОБУЧАЕМЫЙ"	6
Г. В. Веселовська, А. Є. Соколов УДОСКОНАЛЕННЯ МЕТОДІВ УПРАВЛІННЯ СКЛАДНИМИ КОМП'ЮТЕРИЗОВАНИМИ СИСТЕМАМИ ТЕХНІЧНИХ ЗАСОБІВ НАВЧАННЯ НА ЗАСАДАХ КОНЦЕПЦІЙ МЕТОДОЛОГІЇ СИСТЕМ ШТУЧНОГО ІНТЕЛЕКТУ	13
С.Ф. Чалий, В.О. Лещинський, І.О. Лещинська МОДЕЛЮВАННЯ КОНТЕКСТУ В РЕКОМЕНДАЦІЙНИХ СИСТЕМАХ	21
А.Ю. Дорошенко, Н.В. Шаронова, Б.О. Єна, О.В. Янголенко РОЗРОБКА ПРОГРАМНИХ КОМПОНЕНТІВ ІНФОРМАЦІЙНОЇ СИСТЕМИ ЕКСТРАКЦІЇ ФАКТОГРАФІЧНИХ ДАНИХ З ВЕБ-РЕСУРСІВ.....	27
Ю.Б. Бродський, Р.В. Грищук АНАЛІЗ РОЛІ ТА МІСЦЯ, СУТНОСТІ ТА ЗМІСТУ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ НА СУЧАСНОМУ ЕТАПІ РОЗВИТКУ НАУКИ І ТЕХНІКИ	36
В.І. Чупринка, Г.Ю. Зелінський, Н.В. Чупринка АВТОМАТИЧНЕ ПРОЕКТУВАННЯ СХЕМ РОЗКРОЮ ЛИСТОВИХ МАТЕРІАЛІВ НА ПЛОСКІ ГЕОМЕТРИЧНІ ОБ'ЄКТИ	43
П.І. Бідюк, І.В. Пудло, Є.О. Демківський, Т.І. Демківська ПРОГНОЗУВАННЯ ДОХІДНОСТІ БАНКІВСЬКИХ ПРОДУКТІВ З ВИКОРИСТАННЯМ СКОРИНГОВОГО ПІДХОДУ	54
М.В. Деркач, І.С. Скарга-Бандурова ДОСЛІДЖЕННЯ МЕТОДІВ РОЗРАХУНКУ ВІДСТАНІ ДЛЯ ОКРЕМИХ СЕГМЕНТІВ МАРШРУТУ	61
В.А. Дідук МЕТОДИКА ПРИЙНЯТТЯ РІШЕННЯ ПО МІСЦЮ РОЗТАШУВАННЯ ПРОДУКЦІЇ В СКЛАДСЬКИХ ПРИМІЩЕННЯХ	67
И.А. Доровская, Д.В. Доровской СЦЕНАРНО-ПРОЦЕДУРНАЯ МОДЕЛЬ ФУНКЦИОНИРОВАНИЯ ВИРТУАЛЬНЫХ ТРЕНАЖЕРОВ ДЛЯ ПОДГОТОВКИ МЕНЕДЖЕРОВ	73
В.С. Димов, О.О. Боскін ВИКОРИСТАННЯ ЛЮДИНО-МАШИНОГО ІНТЕРФЕЙСУ ДЛЯ УПРАВЛІННЯ ТЕХНОЛОГІЧНИМИ ПРОЦЕСАМИ	81
O.S. Gerasin, O.V. Kozlov, G.V. Kondratenko, H. Mingxin SYNTHESIS AND STUDY OF THE MATHEMATICAL MODEL OF A CATERPILLAR MOBILE ROBOT FOR VERTICAL MOVEMENT	87
Л.М. Захарченко, Р.М. Захарченко, Т.Г. Кірюшатова, К.В. Кірюшатова ПІДГОТОВКА КОРИСТУВАЧІВ НОВОЇ ВЕРСІЇ ПРОГРАМИ 1С: ПІДПРИЄМСТВО 8.3. ДЛЯ УКРАЇНИ	98
О.М. Лобода, В.С. Димов МОДЕЛІ ТА МЕТОДИ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ УПРАВЛІННЯ АГРАРНОГО СЕКТОРУ ЕКОНОМІКИ ЗА ДОПОМОГОЮ ДОСТАТНІХ УМОВ ОПТИМАЛЬНОСТІ	104
А.М. Мацуй, В.О. Кондратець МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ АВТОМАТИЧНОГО КОНТРОЛЮ ЕНЕРГОЕФЕКТИВНОСТІ ПОДРІБНЕННЯ РУДИ КУЛЬОВИМИ МЛИНАМИ ПРИ СПРАЦЮВАННІ ФУТЕРОВКИ	111
S.B. Prykhodko, N.V. Prykhodko, T.G. Smykodub, A.V. Spinov CONSTRUCTING THE NON-LINEAR REGRESSION EQUATION TO ESTIMATE THE SOFTWARE SIZE OF OPEN SOURCE PHP-BASED INFORMATION SYSTEMS.....	118
В.М. Рябенський, І.І. Чудайкін, Ю.Д. Таргунакова РОЗРОБКА ПРОГРАМНОГО КОМПЛЕКСУ ДЛЯ РОЗВ'ЯЗАННЯ ПРЯМИХ ЗАДАЧ МОРСЬКОЇ МАГНІТОМЕТРІЇ.....	126
В.О. Яковенко, Ю.В. Ульяновська АНАЛІЗ ЗАДАЧІ ПОБУДОВИ АВТОМАТИЗОВАНОЇ СИСТЕМИ ВИЯВЛЕННЯ ТА ПОПЕРЕДЖЕННЯ НАДЗВИЧАЙНИХ СИТУАЦІЙ НА ПІДПРИЄМСТВАХ ПІДВИЩЕНОЇ НЕБЕЗПЕКИ.....	134
Т.В. Козуля, Е.В. Ведь ОБЗОР МОДЕЛЕЙ ГЕТЕРОГЕННОГО КАТАЛИЗА.....	140



О.В. Галкін, М.М. Верес, В. О. Ларін, О.В. Бантиш ВИКОРИСТАННЯ ПРЕДМЕТНО – ОРІЄНТОВАНОЇ МОВИ І ВІЗУАЛЬНИХ ПІДХОДІВ ДЛЯ ПРОЕКТУВАННЯ СИСТЕМИ АКТОРІВ(АККА)	148
О.К. Везумський ХАОС ФРАКТАЛ БИРЖА.....	154
М.М. Шевчук, Я.О. Юсин, Т.М. Заболотня, Н.А. Рибачок, А.І. Дичка ПІДХОДИ ДО ОБРОБЛЕННЯ «СИНТАКСИЧНОГО ЦУКРУ» ПРИ ПОШУКУ ПЛАГІАТУ В ПРОГРАМНОМУ КОДІ.....	160
D.O. Zuev, E.V. Dos, A.V. Kropachev, O.V. Babkin, A.A. Varlamov POWER MANAGEMENT FOR SERVER CLUSTERS HARDWARE	166
М.Л. Дворецький, С.В. Дворецька, Є.О. Давиденко РОЗРОБКА СИСТЕМИ УПРАВЛІННЯ ЗНАННЯМИ ОРГАНІЗАЦІЇ НА БАЗІ CMS WORDPRESS	173
І.М. Журавська ГЕНЕРАЦІЯ СУБ-ОПТИМАЛЬНИХ МАРШРУТІВ БЕЗПІЛОТНОГО ЛІТАЛЬНОГО АПАРАТА З ВИКОРИСТАННЯМ НЕЙРОННОЇ МЕРЕЖІ ХОПФІЛДА.....	181
Б.А. Салтан, Д.А. Собко, І.В. Кулаковська МАТЕМАТИЧНА МОДЕЛЬ АЛГОРИТМУ ВИБОРУ ЗБРОЇ БОТОМ В КОМП'ЮТЕРНІЙ ГРІ ЖАНРУ ШУТЕР	186
О.О. Боскін, С.Р. Мазманян, А.М. Левицька АНАЛІЗ БЕЗПЕКИ ІНТЕРФЕЙСУ 802.11(WI-FI З'ЄДНАНЬ), КЛАСИФІКАЦІЯ ПАРОЛІВ	194
Т.В. Горбач, В.Ю. Славгородський, І.Ю. Шубін, А.В. Ковалевська МОДЕЛІ ЕЛЕКТРОННОГО НАВЧАННЯ ТА ВИМОГИ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	205
В.В. Ткаченко, О.Ю. Чередніченко, М.А. Вовк, С.І. Єршова ПІДХІД ДО ЗБОРУ ІНФОРМАЦІЇ ЩОДО ЕКОЛОГІЧНОЇ ОБСТАНОВКИ ПРИ ВИНИКНЕННІ НАДЗВИЧАЙНИХ СИТУАЦІЙ ТЕХНОГЕННОГО ХАРАКТЕРУ	219
О.М. Яшина, Д.І. Братасюк, Р.В. Гремечевський ПОРІВНЯЛЬНА ХАРАКТЕРИСТИКА АКТУАЛЬНИХ ЗАСОБІВ РОЗРОБКИ ПІД МОБІЛЬНІ ПЛАТФОРМИ.....	227
В.Е. Ходаков, С.В. Яцюк КОНЦЕПТУАЛЬНЫЕ ОСНОВЫ ПОСТРОЕНИЯ ПРОДУКЦИОННЫХ СИСТЕМ ЭКСПЕРТНОГО ОЦЕНИВАНИЯ СОЦИАЛЬНО-ЭКОНОМИЧЕСКИХ СИСТЕМ.....	234
Л.О. Катеринич, Ю.Ю. Петелько АЛГОРИТМ ПОШУКУ ЗВ'ЯЗКІВ І ЗАЛЕЖНОСТЕЙ У ДАНИХ ВЕБ-СТОРИНОК	241
Є.В. Лєпа Моделювання Корпоративної Комп'ютерної Мережі	248
Л.В. Колесник, Н.А. Кириченко, І.В. Костоглот РОЗРОБКА ЗАСОБУ ПРОЕКТУВАННЯ ВИСОКОНАВАНТАЖЕНИХ РЕЛЯЦІЙНИХ СИСТЕМ ЗБЕРІГАННЯ ДАНИХ: ОПТИМІЗАЦІЯ СТРУКТУРИ ТА ЗАПИТІВ SQL	253
Ю.О. Накул РОЗРОБКА ІНФОРМАЦІЙНОЇ МОДЕЛІ КОМП'ЮТЕРНОЇ СИСТЕМИ КОНТРОЛЮ ЗАВАНТАЖЕННЯ КОНТЕЙНЕРОВОЗА	261
АНОТАЦІЇ	267



ОБУЧЕНИЕ КАК ПРОЦЕСС ИЗМЕНЕНИЯ ПОСЛЕДОВАТЕЛЬНОСТИ СОСТОЯНИЙ ОБУЧАЕМОГО В СИСТЕМЕ "ОБУЧАЮЩИЙ-ОБУЧАЕМЫЙ"

УДК 539.2

ХОДАКОВ Виктор Егорович

д.т.н., профессор, заслуженный деятель науки и техники Украины, заведующий кафедрой информационных технологий факультета кибернетики и системной инженерии Херсонского национального технического университета, **научные интересы** – прикладной системный анализ, управление социально-экономическими системами.

СОКОЛОВ Андрей Евгеньевич

к.т.н., доцент, доцент кафедры информационных технологий факультета кибернетики и системной инженерии Херсонского национального технического университета, **научные интересы** – компьютеризированные системы обучения.

ВЕСЕЛОВСКАЯ Галина Викторовна

к.т.н., доцент, доцент кафедры информационных технологий факультета кибернетики и системной инженерии Херсонского национального технического университета, **научные интересы** – компьютеризированные системы обучения.

ВВЕДЕНИЕ

Обучение в общем виде можно представить как процесс, то есть как ход, развитие какого-либо явления, как последовательную смену состояний в развитии образования.

XXI век в области обучения характеризуется интенсивным развитием компьютерных средств, информационных технологий, изменяется общая парадигма конструирования и исследования средств вычислительной техники, которая означает переход к технической и программной базе мультимедиа и гипермедиа, в одной среде могут использоваться тексты, графика, звук, видео, 3D и т.п., что приводит к существенной переоценке методов разработки обучающих систем.

Автоматизированные обучающие системы (АОС) сегодня получили более современное название компьютеризированных.

Рассмотрим краткую историю вопроса.

В отечественной литературе еще в 80-90 годах был провозглашен подход к созданию обучающих систем, в

которых процесс обучения зависит от свойств обучаемого.

Большой вклад в развитие теории внесли А. Я. Савельев, А. М. Довгялло [1-3].

Широкое применение нашел подход, учитывающий принцип обратной связи в процессе обучения, когда качество усвоения предшествующего материала учитывается при подаче последующего – обучающие системы ANALYZER, ОККАМ [4].

Такие системы анализировали ошибки и выдавали рекомендации по повторению определенных частей курса.

Но сам процесс обучения в этом случае не составлял непрерывный процесс.

Системы в основном использовались для самоконтроля.

ПОСТАНОВКА ЗАДАЧИ

Большинство обучающих систем [5,6] основано на последовательном изложении материала и контроле-

ных заданий, проверяющих условие материала сравнением ответа с гипотезой.

Последовательное изложение материала и оторванность контроля получаемых результатов от процесса обучения являются основными недостатками этих систем, причина которых кроется в недостаточной разработке моделей, составляющих процесс обучения и использовании модели обучаемого.

Исходя из изложенного, целесообразно продолжить исследования, разрабатывать модели процессов обучения, модели обучаемого (ученика) [7-10].

Целью работы является формализация процесса обучения, выявление основных особенностей этого специфического процесса.

В данном случае, обучение представляется в виде последовательной смены состояний обучаемого в процессе обучения.

РЕШЕНИЕ ЗАДАЧИ

При обучении система "обучающий и обучаемый" (учитель – ученик) эволюционирует, переходя от одного объекта изучения к другому, по-возможности, сохраняя логическую и смысловую связь между изучаемыми вопросами.

Рассмотрим упорядоченное множество изучаемых объектов S , причем порядок определен по смысловой и логической связи.

Тогда переход от s_i к s_{i+1} определяет переход от одной темы к другой.

Для описания всего объема знаний, введем декартово произведение $\Omega_5 = S_1 \times S_2 \times S_3$.

Объем знаний Ω_5 можно представить графически (рис.1.).

Положив, что множество S_1 является множеством отправления, множество S_2 представляет собой множество прибытия, а множество S_3 , определяющее смежные вопросы, связано с этими множествами через проекцию $\text{Pr}S_3 = S_1 \times S_2$, можно определить реальную траекторию обучения – движение в области применения знаний и исполняемую траекторию – движение в области изучаемых знаний.

Графически представить весь объем знаний – задача сложная, однако для конкретной области знаний или учебного предмета это не только возможно, но и повседневно выполняется при анализе и разработке учебных планов.

Теперь можно определить полный объем знаний Ω_1 и говорить о полноте и связности знаний.

Важно учесть, что в силу того, что исполняемая траектория является проекцией реальной траектории, любое нарушение последовательности, связности и истинности на исполняемой траектории – нарушение реальной траектории обучения.

Для области знаний исполняемая траектория обучения определяется более жестко и имеет вид специальной сети, где осуществляются переходы между темами (рис. 2.).

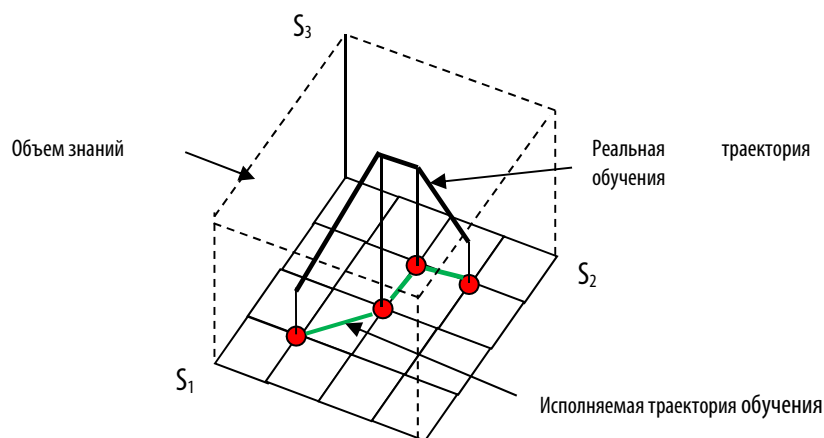


Рисунок 1 – Объем знаний и траектории обучения.

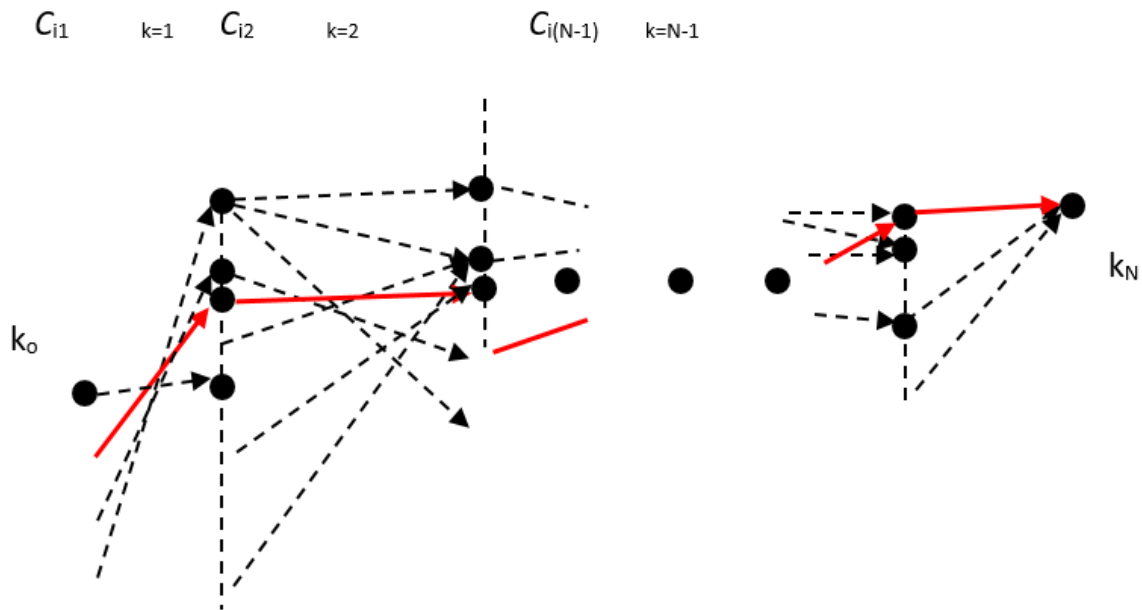
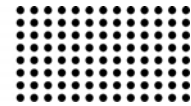
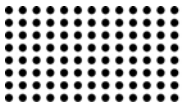


Рисунок 2 – Исполняемая траектория обучения

Затраты на переход C_{ij} , различаются немного в силу длительной работы по оптимизации учебных планов, поэтому функционал цели обучения достаточно близок к линейной функции.

Примем функционал цели на траектории j , принадлежащей оптимальной траектории $J^* = \{j_1^*, j_2^*, \dots, j_N^*\}$, в задаче обучения в виде:

$$C_{j \in J^*} = \frac{1}{N \sum_{\substack{i=1 \\ j \in J^*}}^N (C_{\max} - C_{ij})} \sum_{i=1}^N (C_{\max} - C_{ij}) \theta_i \quad (1)$$

где θ_i – оценка результата обучения на i -м шаге согласно принятой шкалы желательности;

C_{\max} – максимальные затраты;

N – количество переходов на траектории развития.

Собственно, формула (1) – просто привычная уточненная оценка по среднему баллу, что позволяет представить процесс обучения как оптимизационную задачу обучения:

$$J^* \rightarrow \max C_{j \in J^*} \quad (2)$$

С другой стороны, возникает задача построения оптимальной траектории, составляющей минимум затрат на выполнение плана.

В этом случае минимизируется функционал затрат:

$$J^* \rightarrow \min \sum_{i=1}^N C_{i,j} \quad (3)$$

Исходя из изложенного, задача оптимизации траектории обучения может быть рассмотрена как задача динамического программирования.

Метод динамического программирования [11] основан на принципе или методе Беллмана.

В основе этого метода лежит каноническая система уравнений Лагранжа-Эйлера для задачи оптимального управления с целевым функционалом:

$$J(\mathbf{x}, \mathbf{u}, t) = \int_{t_0}^{t_1} f_0(\mathbf{x}, \mathbf{u}, t) dt \quad (4)$$

при ограничениях в виде динамической системы:

$$\frac{d\mathbf{x}(t)}{dt} = \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, t); \quad \dim \mathbf{x} = \dim \mathbf{f} = n, \quad (5)$$

и постановкой задачи с закреплением концов траекторий $\mathbf{x}(t_0) = \mathbf{x}_0, \mathbf{x}(t_1) = \mathbf{x}_1$.

Ставится задача найти оптимальное управление \mathbf{u}^* и оптимальную траекторию \mathbf{x}^* на оптимальном интервале t^*_0, t^*_1 :

$$(\mathbf{x}^*, \mathbf{u}^*, t^*) \rightarrow \inf J(\mathbf{x}, \mathbf{u}, t); \quad (6)$$

Решение данной задачи определяется системой уравнений Лагранжа-Эйлера в канонической форме:

$$\left. \begin{aligned} \frac{\partial H}{\partial \mathbf{x}} &= \frac{d\boldsymbol{\lambda}}{dt} \\ \frac{\partial H}{\partial \boldsymbol{\lambda}} &= -\frac{d\mathbf{x}}{dt} \\ \frac{\partial H}{\partial \mathbf{u}} &= \mathbf{0} \end{aligned} \right\} \quad (7)$$

После стандартного предположения выпуклости задачи, из условий Кунны-Таккера получаем дополнительно два условия:

$$\left. \begin{aligned} \mathbf{x}^* &\xrightarrow{\mathbf{u}=\mathbf{u}^*} \min H \\ \mathbf{u}^* &\xrightarrow{\mathbf{x}=\mathbf{x}^*} \max H \end{aligned} \right\} \quad (8)$$

Первое условие известно как принцип максимума Понтрягина, второе условие – как принцип Беллмана.

Принцип максимума Понтрягина неприменим в случае отсутствия выпуклых свойств в задаче.

Потребуем от динамической системы только выполнения условий существования и единственности на всей траектории или на всех отрезках траекторий.

При этом в окрестности конца траектории существует оптимальная траектория, которую можно определить, постепенно увеличивая ее длину (рис. 3).

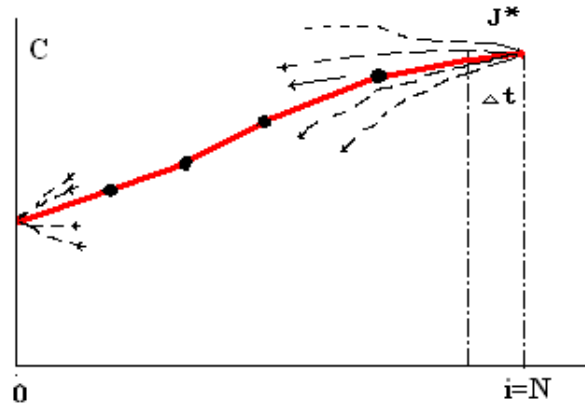


Рисунок 3 – Выделение траектории в T задаче

Решаем задачу минимизации функционала цели на интервале (t, t_1) :

$$\begin{aligned} (\mathbf{x}^*, \mathbf{u}^*, t^*) \rightarrow \inf J(\mathbf{x}, \mathbf{u}, t) &= \int_{t_1}^{t_2} f_0(\mathbf{x}, \mathbf{u}, t) dt, \\ \frac{d\mathbf{x}(t)}{dt} &= \dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}, t); \quad \dim \mathbf{x} = \dim \mathbf{f} = n \\ \mathbf{x}(t) &= \mathbf{x}, \quad \mathbf{x}(t_1) = \mathbf{x}_1. \end{aligned} \quad (9)$$

В полученной, так называемой, T задаче интервал (t, t_1) возможно выбрать малым, и тогда T задача обладает свойством линейности.

Ограничения связаны с выбором траекторий \mathbf{x} , интервала времени (t, t_1) и делятся на фазовые и ограничение по времени.

Функция Гамильтона в этой задаче имеет вид:

$$H(\mathbf{x}, \mathbf{u}, \boldsymbol{\lambda}, t) = \lambda_0 f_0 + \boldsymbol{\lambda}_t \cdot \mathbf{1} + \langle \boldsymbol{\lambda}, \mathbf{f} \rangle \quad (10)$$

Так как множитель Лагранжа определен как чувствительность функции цели $J_t = J_t(\mathbf{x}, \mathbf{u}, t)$ к фазовым и временным ограничениям, и учитывая, что J_t является функцией Белмана V , задача динамического программирования принимает вид:

$$\lambda_t = \frac{dV}{dt}; \quad \lambda_0 = const; \quad \lambda_t = \frac{\partial V}{\partial x_t}. \quad (11)$$

Следовательно, функция Гамильтона в T задаче принимает вид:

$$H(\mathbf{x}, \mathbf{u}, \boldsymbol{\lambda}, t) = \lambda_0 f_0 + \frac{dV}{dt} + \sum_{i=1}^n \frac{\partial V}{\partial x_i} f_i \quad (12)$$

Задача поиска оптимальной траектории при заданных управлениях решается как прямая Т задача:

$$\begin{aligned} \mathbf{x}^* \rightarrow_{u=u^*} \min H(\mathbf{x}, \mathbf{u}, \boldsymbol{\lambda}, t) = \\ = \min_{u=u^*} \left\{ \lambda_0 f_0 + \frac{dV}{dt} + \sum_{i=1}^n \frac{\partial V}{\partial x_i} f_i \right\} \end{aligned} \quad (13)$$

Полученное условие является уравнением Беллмана и, совместно с условием стационарности по управлению, позволяет определить оптимальное управление на всей траектории.

Причем требование существования и единственности решения позволяет утверждать, что оптимальная траектория состоит из оптимальных траекторий.

Дополнительно следует отметить, что данная задача является прямой задачей:

$$\mathbf{x}^* \rightarrow_{u=u^*} \min H(\mathbf{x}^*, \mathbf{u}^*, \boldsymbol{\lambda}^*). \quad (14)$$

Учитывая, что на каждом шаге решается выпуклая задача, получаем в целом выпуклую задачу, а следовательно, достигаем глобального оптимума.

Для рассматриваемого случая динамическая система одномерна и линейна, следовательно, используя условие (11), можем записать:

$$\begin{aligned} \mathbf{x}^* \rightarrow_{u=u^*} \min H(\mathbf{x}, \mathbf{u}, \boldsymbol{\lambda}, t) = \\ = \min_{u=u^*} \left\{ \lambda_0 f_0 + \frac{d\lambda_1}{dt} + \lambda_1 f \right\} = 0 \end{aligned} \quad (15)$$

Из выражения (15) следует, что на оптимальной траектории при оптимальном управлении для линейной системы, уравнение Белмана принимает вид:

$$\frac{d\lambda_1}{dt} + \lambda_1 f = -\lambda_0 f_0 \quad (16)$$

Или, с учетом выражения (5), для свободного движения (16), можем записать:

$$\frac{d\lambda_1}{dt} + \lambda_1 \frac{dx}{dt} = 0 \quad (17)$$

Что, при закреплении $x_0=0, x_1=f_0$, дает общее решение (17) в виде:

$$\lambda_1 = C e^{-x} \quad (18)$$

Поскольку λ_0 можно принять равным единице, получаем, что, для отрезка постоянства f_0 , решение уравнения Белмана имеет вид:

$$\lambda_1(t) = \frac{\delta J}{\delta x} = f_0 (1 - e^{-x(t)}) \quad (19)$$

Следовательно, оптимум связан с экспоненциальной чувствительностью функционала цели к ограничениям на каждом участке траектории.

Для случая дискретных переходов это порождает стандартную процедуру динамического программирования – двигаясь из конечной точки траектории, находим оптимальную траекторию и, при прямом движении, определяем оптимальное управление, удерживающее систему на данной траектории.

Так как декларируется попятное движение, то, на каждом отрезке траектории, множитель Лагранжа в задаче подчиняется условию (17) в виде:

$$\frac{d\lambda_1}{dt} + \lambda_1 f = 0 \quad (20)$$

Таким образом, получаем решение задачи нахождения оптимальной траектории обучения X^* , как алгоритм динамического программирования (рис. 4).

Второй задачей является выполнение найденной траектории.

Как известно [11], данная задача решается с использованием адаптивных технологий управления информационными системами.

Основываясь на представлении информационной системы с помощью структурных моделей, описывающих потоки информации, возникающие при реализации информационных технологий, в данном случае, технологий управления информационными системами,

имеем возможность описать систему обучающего (учителя).

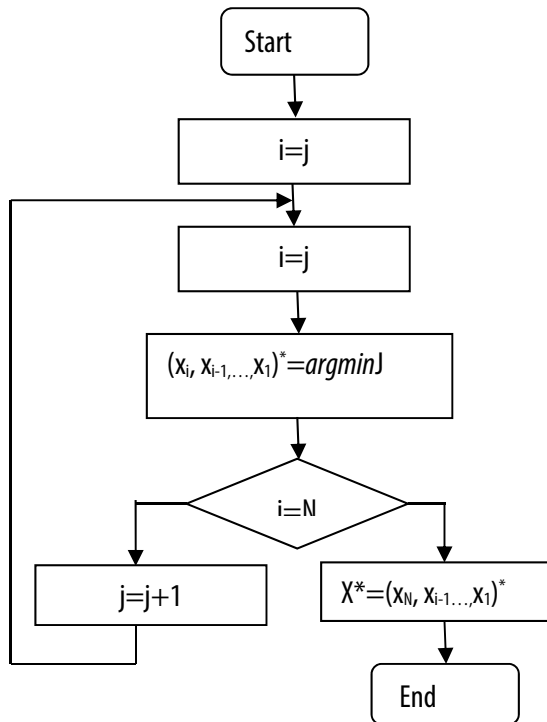


Рисунок 4 – Процедура определения оптимальной траектории обучения

Используя процедуру определения оптимальной траектории обучения, найдена требуемая последова-

тельность передачи информации по каналу учитель – ученик, формируем модель оптимального управления информационными потоками по каналу учителя (рис.5).

Модель, представленная на рис. 5, позволяет взаимно разделить: канал коррекции траектории по оценке информации об ошибке обучаемого ΔI ; канал коррекции модели обучаемого, строящейся на анализе прошлых его действий и сделанных ранее ошибок; канал прогноза действий обучаемого, выполняемого на основе текущих ошибок и модели обучаемого.

ОСНОВНЫЕ РЕЗУЛЬТАТЫ И ВЫВОДЫ

Исследована траектория обучения. В результате проведенного анализа, получены следующие заключения: задача построения оптимальной траектории обучения принадлежит к классу задач управления информационными системами; при построении траектории обучения, любое исключение элемента из объема знаний ведет к нарушению реальной траектории обучения и к потере способности обучаемого решать задачи; специфика оценки знаний, в процессе обучения, позволяет найти частное решение уравнения Беллмана для построения оптимального управления информационными потоками подсистемы обучающего.

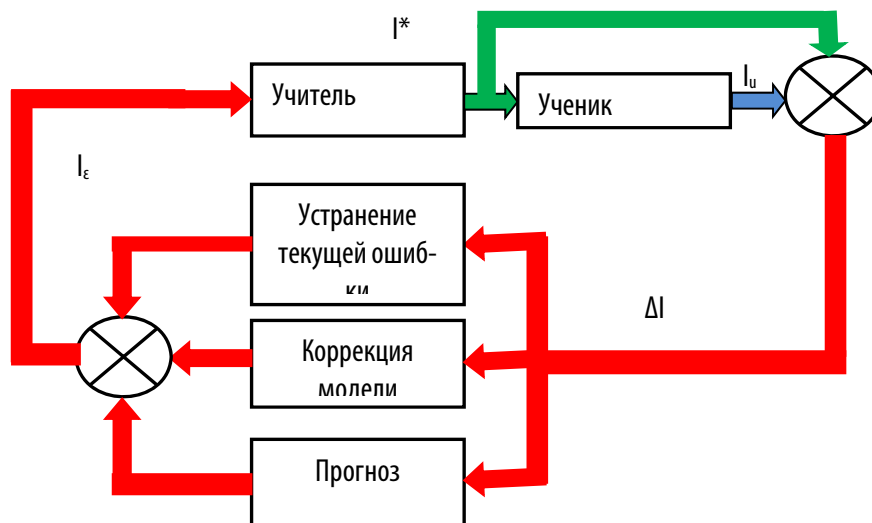


Рисунок 5 – Управление информационными потоками по каналу обучающего



ЛИТЕРАТУРА

1. Greben' I. I. Avtomaticheskie ustrojstva dlja obuchenija / I. I. Greben', A. M. Dovgjalо – K.: Naukova dumka, 1965. – 193 s.
2. Dovgjalо A. M. Tehnologija proektirovanija i razrabotki gibkih distancionnyh obuchajushhh kursov / A. M. Dovgjalо // USiM – upravljajushhie sistemy i mashiny. – 1991. – №1 – S. 95-106.
3. Savel'ev A. Ya. Obuchajushhie mashiny, sistemy i kompleksy / A. Ya. Savel'ev. – K.: Vishha shkola, 1996. – 303 s.
4. Hoa Thang Sravnitel'nyj analiz sistem distancionnogo obuchenija / Hoa Tat Thang // Obshhie problemy obrazovanija. – 2009. – №2. – S. 9-13.
5. Shandrinov V. D. Informacionnye tehnologii v obrazovanii / V. D. Shandrinov // Innovacii v obrazovanii. – 2001. – № – S. 28-33
6. Skibickij E. G. Didakticheskoe obespechenie processa distancionnogo obuchenija / E. G. Skibickij // Distancionnoe obrazovanie. – M. 2000. – №1. – S. 21-25
7. Sokolov A. E. Modelirovanie processa obuchenija s ispol'zovaniem modelej obuchaemogo i obuchajushhegosja / A. E. Sokolov // Problemi informacijnih tehnologij. – 2009. – №2. – S. 154-157
8. Sokolov A. E. Osnovnye principy sozdanija komp'juternyh obuchajushhh programm / A. E. Sokolov // Informatizacija osviti Ukraini. V mizhn. nauk.-prak. Konferencija traven' 2009, tezi dopov. Herson, 2009. – S. 46-48
9. Sokolov A. E. Formalizacija sostavljajushhh processa obuchenija / A. E. Sokolov, E. O. Mahova // Vestnik HNTU. – 2009, №1 (34) – S. 508-512
10. Petrov E. G. Sovremennye tehnologii obuchenija v vysshej shkole / E. G. Petrov, L. N. Radvannaja, N. V. Sharonova – Har'kov: Kollegium, 2007. – 172 s.
11. Krasovskij A. A. Spravochnik po teorii avtomaticheskogo upravlenija / A. A. Krasovskij – M.: Nauka fizmat-izdat, 1987. – 712 s.

*Рецензент: д.т.н., проф. В.А. Доровской
Профессор кафедры информатики и
социально-гуманитарных дисциплин
Криворожского филиала ЧВУЗ
«Европейский университет»*

УДОСКОНАЛЕННЯ МЕТОДІВ УПРАВЛІННЯ СКЛАДНИМИ КОМП'ЮТЕРИЗОВАНИМИ СИСТЕМАМИ ТЕХНІЧНИХ ЗАСОБІВ НАВЧАННЯ НА ЗАСАДАХ КОНЦЕПЦІЙ МЕТОДОЛОГІЇ СИСТЕМ ШТУЧНОГО ІНТЕЛЕКТУ

УДК 004.3(075)

ВЕСЕЛОВСЬКА Галина Вікторівна

кандидат технічних наук, доцент кафедри інформаційних технологій, доцент, кафедра інформаційних технологій факультету кібернетики та системної інженерії Херсонського національного технічного університету,

Наукові інтереси: технології підвищення ефективності комп'ютерних систем і мереж.

E-mail: galina.veselovskaya@gmail.com.

СОКОЛОВ Андрій Євгенович

кандидат технічних наук, доцент кафедри інформаційних технологій, доцент, кафедра інформаційних технологій факультету кібернетики та системної інженерії Херсонського національного технічного університету,

Наукові інтереси: комп'ютеризовані системи навчання.

E-mail: sokol8484@inbox.ru.

ВСТУП

Динамічні зрушення, що є характерними для світових соціально-економічних процесів, обумовлюють зростання вимог ринку праці до всіх організаційних рівнів, видів і форм підготовки фахівців.

Зазначене вище обумовлює посилену увагу державних урядових організацій та установ до розробки нормативної бази, організаційних форм і заходів щодо підтримки та посилення процесів підвищення рівня фахової підготовки.

Високу актуальність і практичну цінність для розв'язування задач інтенсифікації та підвищення результативності процесів фахової підготовки мають технічні засоби навчання.

На даний час було проведено численні теоретичні та прикладні дослідження, існують фундаментальні наукові напрацювання щодо підвищення ефективності застосування технічних засобів навчання в підготовці фахівців.

Разом із тим, сучасні комп'ютеризовані технічні засоби навчання, що являють собою складні системи, потребують подальшого вдосконалювання методів управління ними.

АНАЛІЗ ПРОБЛЕМИ ТА ПОСТАНОВКА ЗАДАЧІ

Сучасні технічні засоби навчання являють собою комп'ютеризовані системи, що ґрунтуються на можливості застосування розвинених локальних і мережних інформаційних технологій із якісною статичною та динамічною, реалістичною двовимірною та тривимірною, інтерактивною, мультимедійною та базованою на гіперпосиланнях візуалізацією навчального процесу.

Надзвичайно важливим є те, щоб різноманітні та потужні можливості технічних засобів навчання використовувалися в повній мірі, продуктивно та раціонально.

Виходячи з вище сказаного, на практиці одне з чільних місць об'єктивно посідає ефективне управління

як навчальним процесом, здійснюваним із застосуванням технічних засобів навчання, так і безпосередньо самими технічними засобами навчання.

Питання управління технічними об'єктами, процесами та системами є достатньо фундаментально опрацьованими в теорії та на практиці, являючи собою тісно взаємно пов'язані та взаємодіючі компоненти цілісного методологічного апарату.

Не менша увага приділялася і роботі над питаннями управління навчальним процесом у цілому та, зокрема, управління ним як середовищем застосування технічних засобів навчання.

При дослідженні аспектів фахової підготовки на базі технічних засобів навчання, перевага віддавалася їхньому розгляду під кутом зору педагогічних міркувань, навчальних підходів, методичних концепцій, організаційних засад, виховних питань, фізіологічних і психологічних вимог, ергономічних критеріїв тощо.

Протягом тривалої еволюції розвитку та практичного впровадження технічних засобів навчання, дослідженню та розробці питань ефективного управління ними, що знаходяться на перетині предметних галузей управління технічними системами, управлінсько-організаційними структурами й освітніми процесами, було присвячено цілий ряд плідних фундаментальних теоретичних і прикладних праць вітчизняних і зарубіжних учених.

Переважну більшість питань управління технічними засобами навчання як складовою ланкою навчального процесу на даний час вдалося успішно здійснити та відпрацювати на найвищому рівні науково-дослідних робіт.

У підсумку інтеграції зазначених вище досягнень, сформувалася базова методологія управління навчальним процесом на основі технічних засобів навчання.

Але не всі актуальні аспекти управління комп'ютеризованими технічними засобами навчання на даний час є опрацьованими в потрібній мірі.

Зазначене має свої об'єктивні підґрунтя, обумовлені рядом проблемних факторів, пов'язаних із практичним застосуванням комп'ютеризованих технічних засобів навчання.

Важливою проблемою є те, що сучасні комп'ютеризовані технічні засоби навчання являють собою складні багатофакторні системи, котрим властивий неспинний динамічний розвиток, стимульований як постійним удосконалюванням можливостей їхньої нормативної й апаратно-програмної бази, так і зростанням вимог до них з боку цільових категорій їхніх користувачів.

Ще одну ключову проблему становить необхідність ретельнішого дослідження та врахування яскраво висловленої специфічності комп'ютеризованих технічних засобів навчання та управління ними, пов'язаної з високою складністю вдалого поєднання та взаємодії їхніх основоположних предметних галузей – сфери навчання та галузі технічних засобів.

Важливим є приділення набагато більшої уваги підвищенню валідності сполучення педагогічних і технічних аспектів управління комп'ютеризованими технічними засобами навчання.

У даній статті опрацьовуються недостатньо досліджені питання удосконалення методів управління комп'ютеризованими технічними засобами навчання на засадах методологій складних систем і систем штучного інтелекту.

РОЗВ'ЯЗОК ЗАДАЧІ

Технічні засоби навчання (ТЗН), досліджувані як системи, мають ряд специфічних особливостей, які було проаналізовано та, в підсумку, їхній перелік зведено в таблицю 1.

Таблиця 1

№	Найменування ключових особливостей ТЗН
1	Вимога підтримки високої загальної гнучкості
2	Вимога достатньо високого рівня адаптивності до цільових категорій користувачів;
3	Невід'ємна, об'єктивно обумовлена наявність певної частки неповноти та неточності інформації про складові компоненти, взаємозв'язки компонентів, вхідні/ вихідні дані, стани системи, процеси та стани функціонування системи, цільові середовища дії та об'єкти впливу системи, про систему в цілому, про фактори її зовнішнього середовища тощо;
4	Працюють переважно зі складними предметними галузями з динамічно змінюваним контентом.

Ефективність функціонування ТЗН в указаних умовах обумовлено успішною взаємодією її ключових

складових, таких як підсистеми обробки та передачі інформації тощо.

Оскільки в основному фактори недостатньої визначеності пов'язані з поточними ситуаціями управління ТЗН у режимі реального часу та за ймовірності виникнення форс-мажорних ситуацій, важливу роль відіграє підтримка режиму динамічного управління.

Зазначений режим має реалізувати встановлену концептуальну модель динаміки системи (її моніторингу, аналізу, моделювання, обробки тощо), планування та диспетчеризації активності її компонентів.

Будемо розглядати таку динаміку робочого середовища ТЗН, коли процеси навчання, що стаються в реальному часі, передбачають можливість виникнення та впливу на зазначені процеси (на плин їхнього проходження, ефективність тощо) нештатних (непередбачених, недостатньо визначених або повністю невизначених) подій.

Розвиненість і гнучкість можливостей сучасного теоретичного та прикладного методологічного апарату систем, в яких мають місце нечітко визначені та динамічні елементи (об'єкти, стани, процеси тощо) та здійснюється управління ними, передбачає можливість високої валідності вибору конкретного класу моделей і здійснення моделювання для певних видів і реалізацій ТЗН.

З іншого боку, створення оптимальних підсистем управління ТЗН і забезпечення подальшого їхнього ефективного супроводження певним чином поважчує наявність людино-машинних ланок системи (цільових категорій користувачів, системних адміністраторів, програмістів, управлінських кадрів, обслуговуючого персоналу тощо) та, відповідно, характерного для них «людського фактору».

Суттєвий позитивний внесок у вирішення проблеми дозволяє зробити застосування композиції P° ряду взаємно пов'язаних підходів, що представлено за допомогою формули (1):

$$P^\circ = p1 \circ p2 \circ p3, \quad (1)$$

де $p1$ – підхід, який полягає в побудові ієрархій категорій підсистем управління ТЗН, їхніх характеристик

і концептуальних підходів до їхнього моделювання, представлених у форматі семантичної мережі;

$p2$ – підхід, який передбачає здійснення логічного моделювання з метою покрокового формування сукупності концепт-моделей підсистеми управління ТЗН;

$p3$ – підхід, який проваджує розробку інтегрованої концепт-моделі підсистеми управління ТЗН, реалізацію якої підтримано сукупністю впорядкованих наборів покрокових операційних дій.

Виходячи з вище сказаного, отримання конкретної реалізації підсумкової інтегрованої моделі управління ТЗН може потребувати здійснення численних операцій перебору варіантів імовірних моделей-претендентів, із повертаннями до ключових точок-прапорців їхньої ієрархії, а також набуті численних похибок через автоматичні помилки з боку людино-машинних ланок формування зазначеної моделі.

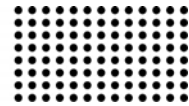
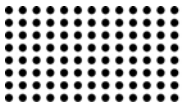
Таким чином, актуальним є впровадження до процесів отримання конкретних реалізацій підсумкових інтегрованих моделей управління ТЗН механізмів автоматизації.

У підсумку, постає задача створення способів автоматизації багатоетапних, ієрархічних, циклічних процесів формування моделей конкретних реалізацій систем управління ТЗН (із властивими для них характерними особливостями, умовами, вимогами тощо), котрі функціонують на засадах високого вмісту недостатньо визначеної інформації та динамічних змін.

Вагомим внеском до вирішення поставленої проблемної задачі є застосування технологій мережних інтелектуальних агентів, що працюють з інформаційними базами даних і знань щодо особливостей ТЗН, а також концепцій і моделей управління ТЗН (як узагальнених, типових, так і конкретизованих).

Зазначений механізм дозволяє здійснювати інтелектуалізовані процедури фільтрації (зокрема, шляхом перебирання можливих варіантів) великих обсягів вихідної інформації, потрібної для формування конкретизованих реалізацій інтегрованих моделей управління ТЗН, з метою дотримання ними заданих вимог та умов.

У даному випадку, процес функціонування механізму інтелектуальних агентів тісно пов'язаний із



формуванням відповідної топологічної моделі нейронної мережі.

На базі розглянутих вище концептуальних підходів і згаданих методологічних положень, можливо розробити чітку, однозначно інтерпретовану та, разом із тим, гнучку інтегровану схему формування моделей управління ТЗН для випадку розгляду конкретизованих ТЗН і ситуацій управління ними, підґрунтям якої є експертні мережні технології інтелектуальних агентів.

У першу чергу, розглянемо найбільш загальний підхід до обґрунтування системи управління ТЗН.

Необхідним є створення такої моделі M^* системи управління ТЗН, в якій структурно буде відображене наступне:

$$M^* = \langle k, e, u, r \rangle, \quad (2)$$

k – компоненти системи управління ТЗН (їхній склад, властивості та взаємозв'язки в статичній та динамічній їхньому розвитку);

e – елементи процесу формування системи управління ТЗН, які добираються відповідно до кортежів-базисів її ключових типізаційних характеристик, що приймають участь у формалізованому описі подання визначальних особливостей системи, виборі валідної системи, а також обумовлюють повну відповідність потребам ТЗН (вимогам, умовам, оцінювальним критеріям тощо);

u – упорядкованість елементів процесу формування системи управління ТЗН за пріоритетністю їхньої активізації згідно логічної моделі багатоетапного ітеративного формування зазначеної системи шляхом перебору в межах простору наборів характеристик-визначників можливих типів систем управління ТЗН, що надає можливість отримати систему управління ТЗН, відповідну поставленим вимогам до ходу та результатів навчального процесу на базі ТЗН;

r – ручний, частково автоматизований та повністю автоматизований режими здійснення процесу формування системи управління ТЗН за допомогою складної гнучкої автономно чи інтерактивно працюючої інтелектуалізованої мультиагентно-орієнтованої системи, що містить нечіткі компоненти-ідентифікатори.

При створенні переліку наборів характеристик-визначників можливих типів систем управління ТЗН, обумовлюється необхідність урахування основних факторів, представлених у таблиці 2.

Таблиця 2

№	Найменування спеціалізованих факторів
1	Можливі типи невизначеності, пов'язані з ТЗН
2	Концепції, механізми та засоби перебудови ТЗН та їхніх систем управління
3	Стратегії планування меж витрат часових, людських, фінансових та інших ресурсів

Таким чином, будемо надалі казати про технологію формування системи управління ТЗН як про багатоетапну ітераційну сукупність операцій отримання такої складної інтелектуалізованої системи, що дозволяє найоптимальніше відповідати особливостям (вимогам, умовам тощо) певної реалізації ТЗН.

Відповідна задача нечіткого визначення конкретної моделі системи управління ТЗН буде зводиться до створення (інтеграції) в динаміці (в реальному часі) на основі наявних інтелектуальних агентів, виділених за їхньою функціональною спеціалізацією, гнучких багатоагентних структур, які найліпше відповідають особливостям певної реалізації ТЗН.

У даному випадку, будемо розглядати окремі інтелектуальні агенти та інтегровані (структуровані, взаємно пов'язані) сукупності інтелектуальних агентів як наявні або проєктовані в реальному часі, повністю автономні чи інтерактивні, програмні або програмно-апаратні модулі, функціонування яких має здійснюватися згідно визначених функціонально-спеціалізованих інтелектуальних покрокових алгоритмів або сценаріїв з метою дотримання систем цілей, висунутих цільовими користувачами та категоріями таких користувачів до ТЗН і систем управління ТЗН.

У підсумку, утворюються гнучкі інтегровані інтелектуальні агенти середовища проєктування систем управління ТЗН, які містять функціонально повні за своїм складом прошарки взаємно пов'язаних складних (зокрема, нечітких) інтелектуальних агентів, що представлене на рисунку 1.

Здійснення окреслених підходів буде ефективним у тому разі, якщо застосовувати:

— методи декомпозиції та децентралізації з відновленням, застосовані щодо структур і параметрів;
— об'єктно-орієнтовані підходи, об'єкти, процеси, алгоритми тощо;

— механізми розпаралелювання обчислювальних алгоритмів.

Типізація спеціалізованих інтелектуальних агентів	Інтелектуальні агенти першого роду: націлені щодо вирішення завдань вибору (ідентифікації) оптимальних систем управління ТЗН згідно певних ключових класифікаційних критеріїв вибору, що належать до базового простору таких критеріїв
	Інтелектуальні агенти другого роду: спрямовані на підтримку розподілених технологій визначення типової належності та пріоритетності заданих інтелектуальних агентів першого роду в режимах перспективного прогнозування та реального часу
	Інтелектуальні агенти третього роду: призначені для підтримки оперативного функціонування (формування, ведення та застосування в складі експертних модулів) спеціалізованих баз знань, які відображають ідентифікаційні закономірності
	Інтелектуальні агенти четвертого роду: відповідають за активацію, а також за правильне та оптимальне здійснення взаємодії (зокрема, в режимі паралельної роботи об'єктів і процесів) інтелектуальних агентів інших (попередніх і наступних) категорій (у тому числі, тих інтелектуальних агентів, які належать до різних структурних утворень)
	Інтелектуальні агенти п'ятого роду: мають продукувати логічні виведення стосовно відповідності поточним вхідним вимогам на певному заданому рівні, приймати рішення та виробляти певні дії-реакції відносно статичних і динамічних цілей, задач, параметрів і станів (вихідних, поточних, результуючих, прогнозованих тощо) ТЗН, середовища застосування ТЗН, зовнішнього середовища та власне управління ТЗН за допомогою механізмів зворотних зв'язків (згідно з результатами ідентифікації, корекція вихідних показників виконується в такий спосіб, щоб відповідати вимогам входу)

Рисунок 1

Початкові та поточні вхідні сигнали системи управління ТЗН, що надходять від численних внутрішніх і зовнішніх джерел (таких, як функціональні процеси, цільові користувачі, зовнішнє середовище тощо), відбивають особливості (властивості, вимоги, умови тощо) предметної галузі зазначеної системи.

У разі неповноти стану початкового та поточного вхідного інформаційного поля системи управління ТЗН, задіюється перетворення вхідної інформації до сукупності змінних-факторів, які:

— представляють нечіткі множини, задані на значеннях вхідних даних;

— супроводжуються відповідними показниками достовірності / недостовірності, що є ступенями належності (найчастіше, сформованими за експертними оцінками).

Також здійснюється зворотне перетворення нечіткої вихідної інформації на чіткі дані, коли сукупності вихідних нечітких факторів і відповідних до них ступенів достовірності потрібно поставити у відповідність сукупність чітких значень вимог можливості реалізації й оптимальності пропонованої системи управління ТЗН при заданих початкових

значеннях властивостей ТЗН, її предметної галузі та зовнішнього середовища.

Загальна схема дії D системи управління ТЗН спирається на узгодженні функціонування її складових, для здійснення якого ключову роль відіграють певні компоненти:

$$D = \langle d_i, i = 1, \dots, 8 \rangle, \quad (3)$$

де: d_1 – обчислювальні механізми, реалізовані на основі чисельних методів;

d_2 – об'єкти з функціональною роллю виконання вибірок серед табличних даних;

d_3 – об'єкти, призначені для здійснення просторової статичної та динамічної графічної візуалізації;

d_4 – механізми отримання й обробки експертних оцінок (висновків, рекомендацій тощо), формування експертних знань;

d_5 – бази знань (евристичні факти, правила-продукції тощо);

d_6 – об'єкти, що відіграють функціональну роль здійснення операцій інтелектуального вибору та в якості яких застосовуються нейромережі;

d_7 – об'єкти, що підтримують уведення до модулів системи нових умов, обмежень, правил та т. і. згідно відповідних заключень-рекомендацій експертів та висновків інтелектуальних агентів;

d_8 – об'єкти, що формують і супроводжують конкретизовані реалізації систем управління ТЗН.

Спираючись на ієрархію цілей, завдань і вимог до вибору системи управління ТЗН, а також на критичні вхідні та поточні дані для її отримання, послідовно здійснюються етапи процесу формування моделі вказаної системи, де:

– на виході кожного з етапів отримуються механізми (концепції, підходи, способи, методи, політики витрат ресурсів тощо), котрі визначають напрямки подальшого ходу діяльності щодо поточного моделювання системи управління ТЗН;

– на виході процесу в цілому отримується підсумкова модель системи управління ТЗН, відповідна специфічним вимогам (умовам, обмеженням тощо), обумовленим ТЗН та її предметною галуззю, а також заданим показникам розмірності математичного апарату, точності наближення до моделі-еталону, складності для реалізації та практичного застосування, беззбійності активізації, надійності та продуктивності (швидкості) функціонування.

Будемо брати за основу для розгляду наступне:

– набори вхідної інформації та вихідних дій-реакцій окремих інтелектуальних агентів;

– набори вхідних і вихідних змінних, які беруть участь в обміні інформацією між інтелектуальними агентами та групами інтелектуальних агентів;

– той факт, що сукупність взаємно пов'язаних функціональних структур інтелектуальних агентів утворює загальну структуру формування інтелектуалізованої моделі системи управління ТЗН.

Тоді конкретні інтелектуальні агенти можна формалізовано подавати на базі наступних характеристик:

– набори вхідної та вихідної інформації інтелектуальних агентів, а також їхніх керуючих і підпорядкованих ланок у загальній ієрархії інтелектуальних агентів;

– належність інтелектуальних агентів до певних класів, залежно від їхнього функціонального

призначення та покладеної до основи їхньої роботи базової технології;

– належність умов запуску функціонування інтелектуальних агентів до певних видів, таких як спрацьовування в реальному часі попередньо заданих умов, визначальна зміна вхідних даних, передача управління іншими інтелектуальними агентами (тимчасова, з метою отримання певних результатів, або остаточна, після завершення їхньої роботи).

У підсумку наведеного вище розгляду, інтегрована модель для системи оптимального вибору (ідентифікації) певної системи управління ТЗН, побудованої на засадах гнучких інтелектуалізованих агентно-орієнтованих технологій, поєднує моделі окремих елементів ідентифікації, що відповідають за підтримку вирішення проблемних завдань з моніторингу й обробки стосовно інформації наступних видів (таблиця 3):

Таблиця 3

№	Види спеціалізованої інформації
1	Точна, неповно визначена та невизначена інформації щодо ТЗН та оточуючого її середовища, предметної галузі ТЗН, управління ТЗН тощо
2	Інформація щодо концепцій, способів, засобів, методів, моделей, стратегій планування та диспетчеризації, представлених у рамках систем управління ТЗН
3	Інформація, що становить собою необхідне підґрунтя утворення алгоритму формування системи управління ТЗН
4	Інформація, що окреслює достатні вимоги, обумовлені специфікою ТЗН та пов'язаних із нею об'єктів і процесів

Таким чином, пропонується впровадження спеціалізованої інтелектуалізованої системи, що має наступні властивості:

– реалізує багатоетапну модель формування системи управління ТЗН;

– побудована на засадах підтримки властивостей гнучкості й агентної орієнтованості;

– будучи повністю автономною, надає можливість роботи як під управлінням внутрішніх механізмів активізації, так і в режимах інтерактивної взаємодії з користувачами;

– підтримує такий підхід до порядку вибору можливих варіантів ідентифікованих систем управління ТЗН у базовому просторі, що безперечно

дозволяє отримати результати, відповідні критеріям оптимального функціонування ТЗН.

Найелементарнішими підсумковими діями з поетапного формування систем управління ТЗН є операції перебору, виконувани на засадах реляційного підходу та засновані на дотриманні умов здійснення критеріїв відповідності наборів можливостей певної системи управління щодо вимог та обмежень конкретної ТЗН.

Обробка відповідних невизначеностей має здійснюватися на базі технологій нечітких множин, лінгвістичних змінних і нечітких нейронних мереж із застосуванням операцій фазифікації / дефазифікації.

Підтримка процесів вибору (ідентифікації) для кожного етапу формування системи управління ТЗН здійснюється на компонентній базі, що міститься у відповідних базах знань експертних модулів і доповнюється в режимі самонавчання інформацією про кожну з реалізацій систем управління, відповідну належним вимогам.

Розглянемо далі питання про застосування агентового підходу до визначення умов.

Моделювання умов, у першу чергу, має за своє підґрунтя реляційні відносини щодо елементів ТЗН та їхніх властивостей, а також пов'язаних із ними зовнішніх і внутрішніх компонентів і процесів.

За допомогою технологій експертного оцінювання (ранжування, попарних порівнянь і т.і.) можливих варіантів, отримуються кількісні рейтингові вагові оцінки реляційних відносин для ключових ознак класифікації.

До розв'язування поставленої задачі також доцільно застосовувати топологічний підхід.

А саме, послідовності реалізацій окремих етапів концептів діяльності з вибору моделей формування систем управління ТЗН, представлені в форматах оптимальних траєкторій руху, обумовлених повнофункціональними OR-графами процесів формування зазначених систем, можуть відображатися за допомогою топологічних структур як нейромережні моделі подання вказаних графів.

Тоді визначальні процеси формування потрібної моделі системи управління ТЗН полягатимуть у корекції значень ваг елементів моделі нейромережі на підставі актуальних значень рейтингових експертних оцінок

потенційно можливих варіантів, приймаючи до уваги ваги реляційних відносин згідно вимог ТЗН на відповідному етапі формування.

Здійснюючи ітераційно вказані корегувальні дії, можна отримати траєкторію оптимального руху в просторі рішень шляхом активізації операції перетину для тих елементів моделей управління ТЗН, які мають найбільші показники відповідності.

Подібні підходи в процесі формування потрібної моделі системи управління ТЗН стосуються і процедур управління етапами її формування.

На підставі оптимальної траєкторії, отриманої за підсумками ітераційного перебору, можливо отримати компоненти потрібної моделі системи управління ТЗН.

Для здійснення сукупності операцій автоматизованого формування, система управління ТЗН має містити автономний компонент для автоматизованого формування параметрів управління з урахуванням фактору динаміки, що систематично, згідно певної часової закономірності, оброблятиме надану з боку ТЗН інформацію щодо її попереднього функціонування та формуватиме відповідні параметри функціонування системи управління ТЗН.

У свою чергу, зазначені параметри надходитимуть до інших компонентів системи управління ТЗН, здійснюючи визначальний вплив на їхнє наступне функціонування.

Слід відзначити важливу роль, яку відіграють у системі певні компоненти та здійснювані ними функції, що представлені в таблиці 4.

Таблиця 4

№	Види спеціалізованих компонентів
1	Компонент, який виконує планування (попереднє календарне, оперативне в реальному часі тощо), приймаючи на вході концепцію динамічного планування та диспетчеризації
2	Компонент, який здійснює в реальному часі оперативне корегування, для чого отримує й обробляє вхідну для нього концепцію перепланування
3	Компонент, який реалізує в реальному часі контроль, узгоджуючи його з отриманою на вході концепцією визначення часу перепланування
4	Компонент, який підтримує диспетчеризацію, приймаючи на вхід концепцію перепланування

ОСНОВНІ РЕЗУЛЬТАТИ ТА ВИСНОВКИ.

Здійснені дослідження особливостей сучасних комп'ютеризованих систем технічних засобів навчання



як складних систем. Запропоновані нові концептуальні підходи до вдосконалювання методів управління технічними засобами навчання на базі інтелектуалізованих механізмів.

Результати виконаних досліджень апробовано в практиці начального процесу кафедри інформаційних технологій Херсонського національного технічного університету, в підсумку чого було отримане посилення ефективності навчального процесу на десять відсотків.

ЛІТЕРАТУРА

1. Buinytska O. P. Informatsiini tekhnologii ta tekhnichni zasoby navchannia [Tekst] / O. P. Buinytska. – K.: Tsentri navchalnoi literatury, 2017. – 240 s.
2. Multymediini systemy yak zasoby interaktyvnoho navchannia [Tekst] : posibnyk / Za red. Zhuka Yu. O. – K.: Pedahohichna dumka, 2012. – 112 s.
3. Veselovska H. V. Analiz osoblyvostei poniatiino-terminolohichnoi bazy ta klasyfikatsiia zasobiv aparatnoi pidtrymky informatsiinykh tekhnologii komp' iuternoho navchannia [Tekst] / H. V. Veselovska, A. Ye. Sokolov // Problemy informatsiinykh tekhnologii. – 2017. – № 2 (022). – S. 117-127.
4. Anisimov A. V. Informatsiini systemy ta bazy danykh [Tekst] : navchalnyi posibnyk / Anisimov A. V., Kuliabko P. P. – Kyiv, 2017. – 110 s.
5. Antonenko V. M. Suchasni informatsiini systemy i tekhnologii: upravlinnia znanniamy [Tekst] : navchalnyi posibnyk / V. M. Antonenko, S. D. Mamchenko, Yu. V. Rohushyna. – Irpin : Natsionalnyi universytet DPS Ukrainy, 2016. – 212 s.
6. Tkachenko R. O. Neiromerzhevi zasoby shtuchnoho intelektu [Tekst] : navchalnyi posibnyk / R. O. Tkachenko. – Lviv: Vydavnytstvo Lvivska Politekhniky, 2017. – 208 s.
7. Lor'er ZH. L. Sistemy iskusstvennogo intellekta [Tekst] / ZH. L. Lor'er. – M.: Mir, 2016. – 568 s.
8. Gavrilova T. A. Bazy znaniy intelektual'nykh sistem [Tekst] : uchebnyk dlya VUZov / T. A. Gavrilova, V. F. Horoshevskiy. – SPb.: Piter, 2016. – 384 s.
9. Rassel S. Isskusstvennyy intellekt: sovremennyj podhod = Artificial Intelligence: a Modern Approach [Tekst] / S. Rassel, P. Norvig. – M.: Vil'yams, 2016. – 1408 s.
10. Lyuger Dzh. F. Isskusstvennyy intellekt: strategii i metody resheniya slozhnykh problem [Tekst] / Dzh. F. Lyuger = Artificial Intelligence: Structures and Strategies for Complex Problem Solving. – M.: Vil'yams, 2015. – 864 s.
11. Petrunin Yu. Yu. Filosofiya iskusstvennogo intellekta v koncepciyah nejronauk [Tekst]: nauchnaya monografiya / Yu. Yu. Petrunin, M. A. Ryazanov, A. V. Savel'ev. – M.: MAKS Press, 2015. – 648 s.
12. Trofimov V. B. Intellektual'nye avtomatizirovannye systemy upravleniya tekhnologicheskimi ob'ektami. [Tekst] / V. B. Trofimov. – Izdatel'stvo Infra-Inzheneriya, 2016. – 232 s.
13. Kozlov V. Sistemnyy analiz, optimizaciya i prinyatie reshenij [Tekst]: uchebnoe posobie / V. Kozlov. – M.: Prospekt, 2014. – 174 s.
14. Pegat A. Nechetkoe modelirovanie i upravlenie: Fuzzy Modeling and Control [Tekst] / A. Pegat. – M.: Binom. Laboratoriya znaniy, 2013. – 800 s.
15. Kolesnikov A. A. Sinergeticheskie metody upravleniya slozhnymi sistemami. Teoriya sistemnogo sinteza [Tekst] / A. A. Kolesnikov. – M.: Librokom, 2012. – 242 s.

Рецензент: д.т.н., проф. В.О. Доровський
Професор кафедри інформатики та
соціально-гуманітарних дисциплін
Криворізької філії ПВНЗ
«Європейський університет»

МОДЕЛЮВАННЯ КОНТЕКСТУ В РЕКОМЕНДАЦІЙНИХ СИСТЕМАХ

УДК 004.891.3

ЧАЛИЙ Сергій Федорович

д. т. н., проф., професор кафедри інформаційних управляючих систем
Харківського національного університету радіоелектроніки

Наукові інтереси: управління знання-ємними процесами, рекомендаційні системи.
e-mail: serhii.chalyi@nure.ua

ЛЕЩИНСЬКИЙ Володимир Олександрович

к. т. н., доц., доцент кафедри програмної інженерії Харківського національного університету радіоелектроніки,
e-mail: volodymyr.leshchynskiy@nure.ua

ЛЕЩИНСЬКА Ірина Олександрівна

к. т. н., доц., доцент кафедри програмної інженерії Харківського національного університету радіоелектроніки,
e-mail: iryna.leshchynska@nure.ua

ВСТУП

Рекомендаційні системи призначені для формування рекомендацій споживачам відносно вибору товарів та послуг при наявності значної кількості альтернатив [1, 2]. В якості вхідних даних такі системи використовують рейтинги або інформацію про покупки інших користувачів. З використанням машинного навчання формуються залежності між властивостями товарів та характеристиками споживачів [3, 4].

Сфера застосування таких систем пов'язано переважно з використанням е-комерції. Такі системи спрощують вибір користувачам, які не мають достатньо знань про характеристики цікавих для них об'єктів (товарів та послуг) при наявності широкого вибору [5].

Рекомендаційні системи використовують ключові інтернет-компанії для того, щоб запропонувати персоналізований вибір. Зокрема Amazon.com використовує ці системи для персоналізації свого магазину, для кожного користувача, сервіс перегляду фільмів Netflix – при формуванні рейтингу для своїх фільмів та телевізійних програм [6 - 8].

Отримані за допомогою цих систем рекомендації представляються у вигляді ранжованого списку товарів, сайтів, тощо, з урахуванням вподобань споживачів.

АНАЛІЗ ДОСЛІДЖЕНЬ ТА ПУБЛІКАЦІЙ

При побудові рекомендаційних системи виникає проблема релевантності даних [9, 10]. Дана проблема пов'язана з недоліками вхідної інформації. Зазвичай матриця рейтингів користувачів або матриця покупок є розрідженими, що утруднює формування рекомендацій.

Для вирішення цієї проблеми в контекстних рекомендаційних системах враховують контекст прийняття рішень, контекст визначення рейтингів користувачами.

Однак існуючі підходи недостатньо уваги приділяють дослідженню неповноти та протиріч у контексті [11-15]. Останні виникають внаслідок того, що кожен користувач має доступ лише до підмножини товарів, послуг, інформації – тобто враховує лише локальний контекст. При ранжуванні результатів рекомендаційною системою для підвищення точності рекомендацій необхідно максимально повно враховувати контекст. Зазначений недолік свідчить про актуальність теми даної статті.

ПОСТАНОВКА ЗАДАЧІ

Метою даної статті є побудова моделі контексту прийняття рішень в рекомендаційних системах з тим, щоб враховувати контекст при ранжуванні об'єктів стосовно користувачів або користувачів стосовно об'єктів.

Використання контекстної складової дозволяє вибрати із множини вхідних даних для побудови рекомендацій таку підмножину, що є релевантною стосовно ситуації, у якій приймає рішення користувач.

Також врахування контекстної складової щодо результатів роботи рекомендаційної системи дає можливість упорядкувати отриманий перелік об'єктів стосовно умов прийняття рішень конкретним користувачем.

Для досягнення мети дослідження вирішуються наступні задачі:

- формалізація підходів до інтеграції контексту в рекомендаційних системах;
- розробка моделі контексту.

ОСНОВНИЙ ЗМІСТ РОБОТИ

Рекомендаційні системи обробляють вподобання користувачів. Кожен з користувачів приймає рішення як на основі своїх внутрішніх вподобань, так і з урахуванням підмножини об'єктів предметної області, з якими він взаємодіє або взаємодівав.

Наприклад, при виборі фільму користувач орієнтується на ті фільми, що він передивився сам, так і на експертну оцінку друзів, колег, огляди фільмів та передач, тощо. Відповідно, користувач може прийняти різні рішення щодо одного й того ж фільму у випадку наявності детальної інформації про фільм та у випадку одностороннього висвітлення недоліків та переваг твору (нецікаві актори, сюжет, зйомки, тощо).

Іншими словами, користувач має локальний доступ до предметної області з описом фільму і, відповідно, локальний контекст прийняття рішень. З метою підвищення релевантності цих рішень при виборі того чи іншого фільму користувач поєднує локальний контекст свої колег, друзів, знайомих, рецензентів фільму.

Такий підхід до вибору фільму потребує порівняння локальних контекстів з метою видалити суперечливі фрагменти та елементи, що дублюються перед інтеграцією цих контекстів (рис. 1).

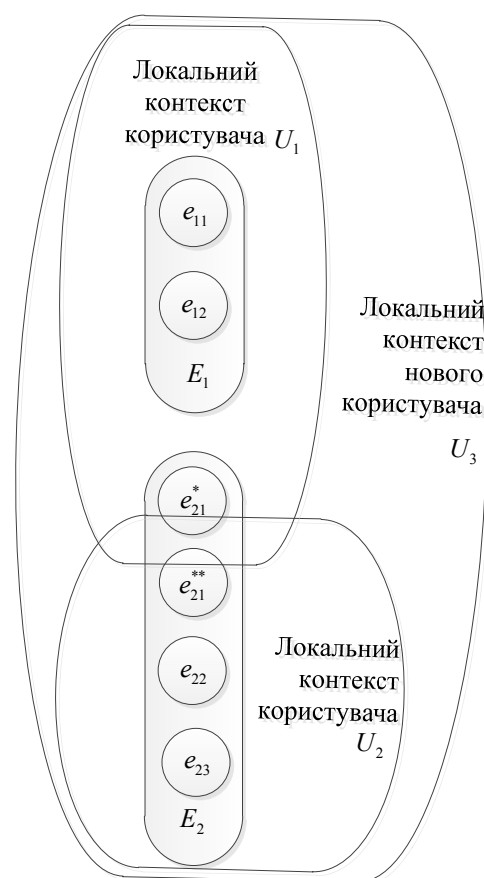


Рис. 1 – Поєднання локальних контекстів для нового користувача U_3

В наведеному на рис. 1 прикладі об'єкти характеризуються множинами атрибутів $E_1 = \{e_{11}, e_{12}\}$ та $E_2 = \{e_{21}, e_{22}, e_{23}\}$. Однак у локальному контексті C^1 користувача U_1 існують об'єкти E_1^1, E_2^1 у такому вигляді:

$$C^1 = \{E_1^1, E_2^1\} \quad (1)$$

де C^1 – контекст прийняття рішень клієнта U_1 ;

E_1^1, E_2^1 – відображення об'єктів E_1, E_2 у контексті першого споживача.

Представлення об'єктів у контексті першого споживача у даному прикладі мають такий вигляд:

$$E_1^1 = \{e_{11}, e_{12}\}, E_2^1 = \{e_{21}^*\} \quad (2)$$

де e_{11}, e_{12} – відображення першого об'єкту у контексті споживача U_1 ;

e_{21}^* – представлення споживача U_1 про атрибут об'єкту E_2 .

Іншими словами, у контексті першого споживача об'єкт E_2 представлений не повністю. Також представлення про атрибут e_{21}^* може не відповідати контекстному представленню інших споживачів.

Аналогічно, локальний контекст другого споживача $C^2 = \{E_1^2, E_2^2\}$ має такий вигляд:

$$E_1^2 = \{\emptyset\}, E_2^2 = \{e_{21}^{**}, e_{22}, e_{23}\} \quad (3)$$

де $e_{21}^{**}, e_{22}, e_{23}$ – відображення першого об'єкту у контексті споживача U_2 ;

e_{21}^{**} – представлення споживача U_1 про атрибут об'єкту E_2 .

У контексті другого споживача об'єкт E_1 не представлений. Також представлення про атрибут e_{21}^{**} відрізняється від представлення першого споживача.

При побудові контексту C^3 нового споживача U_3 в рекомендаційній системі необхідно поєднати відомі локальні контексти за умови поєднання тих атрибутів об'єктів, що відрізняються у локальних контекстах відомих споживачів. В даному випадку при невідповідності одного і того ж атрибуту одного у того ж об'єкту у різних локальних контекстах можуть виникнути дві ситуації.

По-перше, при відмінності представлення атрибутів у локальному контексті внаслідок неповноти інформації у того чи іншого споживача до об'єднаного контексту включається один із атрибутів, інформація про який є більш точною:

$$C^3 = C^1 \cup C^2 \mid \forall e_{ij}^1 \approx e_{ij}^2, (e_{ij}^1 \vee e_{ij}^2) \in C^1 \cap C^2 \quad (4)$$

де $e_{ij}^1 \vee e_{ij}^2$ – аналогічні атрибути одних і тих же об'єктів предметної області, що належать до різних локальних контекстів.

По друге, у випадку протиріч між існуючими представленнями атрибутів вони виключаються з об'єднаного контексту:

$$C^3 = C^1 \cup C^2 \mid \forall e_{ij}^1 \neq e_{ij}^2, (e_{ij}^1 \vee e_{ij}^2) \notin C^1 \cap C^2 \quad (5)$$

У відповідності до наведеного прикладу можна зробити висновок, що при використанні контексту прийняття рішень в рекомендаційній системі необхідно:

- враховувати локальність та неточність контекстів окремих споживачів;
- об'єднувати ці локальні контексти з усуненням неповноти та протиріч у представленні окремих атрибутів таких об'єктів.

Розглянутий ілюстративний приклад дозволяє сформулювати принципи локальності та зв'язності як основу для інтеграції та подальшого врахування контексту при ранжуванні результатів роботи рекомендаційної системи.

Принцип локальності полягає в тому, що кожний користувач використовує персональне відображення контексту прийняття рішень у задачах вибору з великої кількості альтернатив. Дане відображення характеризується такими властивостями:

- містить у собі підмножину об'єктів предметної області;
- враховує підмножину атрибутів кожного з об'єктів предметної області;
- інформація про атрибути може бути неповною, неточною та містити протиріччя.

Усунення неповноти та суперечливості інформації про об'єкти предметної області та властивості цих об'єктів з локальних контекстів споживачів виконується у відповідності до принципу зв'язності.

Принцип зв'язності визначає контекст прийняття рішень в рекомендаційній системі як упорядковану сукупність локальних контекстів користувачів.

На рис.2 шляхом просторового представлення деталізовано різницю між локальними контекстами користувачів U_1 та U_2 . Очевидно, що при наведеній просторовій конфігурації споживачу U_1 доступні такі локальні контексти, які задаються підмножинами властивостей об'єктів E_1 та E_2 :

Аналогічно для споживача U_2 доступні такі локальні контексти:

$$\begin{aligned} C^1 &= \{c_i^1 \mid i = \overline{1,7}\}, \\ c_1^1 &= \{\emptyset\}, c_2^1 = \{e_{11}\}, c_3^1 = \{e_{11}, e_{12}\}, \\ c_4^1 &= \{e_{11}, e_{12}, e_{21}\}, c_5^1 = \{e_{12}, e_{21}\}, \\ c_6^1 &= \{e_{21}\}, c_7^1 = \{e_{12}\}, \end{aligned} \quad (6)$$

$$\begin{aligned} C^2 &= \{c_i^2 \mid i = \overline{1,7}\}, \\ c_1^2 &= \{\emptyset\}, c_2^2 = \{e_{21}\}, c_3^2 = \{e_{31}, e_{32}\}, \\ c_4^2 &= \{e_{31}, e_{32}, e_{31}\}, c_5^2 = \{e_{32}, e_{31}\}, \\ c_6^2 &= \{e_{31}\}, c_7^2 = \{e_{32}\}, \end{aligned} \quad (7)$$

Пуста множина елементів в одній із реалізацій контексту виразах (6) та (7) означає, що з точки зору споживачів всі властивості відсутні у обох об'єктах. Аналогічно, наявність лише властивості e_{11} в c_2^1 означає, що всі інші властивості у даній реалізації контексту відсутні.

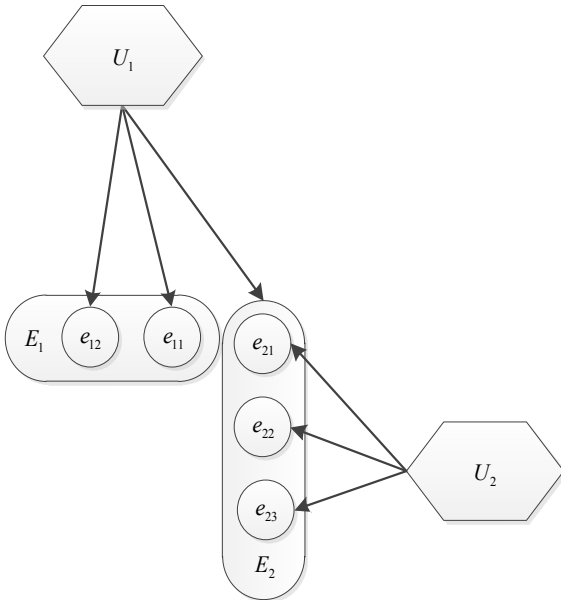


Рис. 2 – Локальність контекстів користувачів U_1 та U_2

Якщо упорядкувати послідовність елементів у множинах C^1 та C^2 , то ми отримаємо можливість інтегрувати ці контексти шляхом послідовного об'єднання реалізацій c_i^1 та c_i^2 :

При такому поєднанні отримаємо деревоподібну структуру, з покроковою деталізацією об'єднаного контексту на кожному наступному рівні дерева. Перші два рівня для наведеного на рис. 2 прикладу мають такий вигляд:

$$C^3 = C^1 \bigcup_{e_{ij}} C^2 = \left\langle \begin{array}{l} \{e_{21}\}, \\ \{\{e_{21}, e_{11}\}, \{e_{21}, e_{22}\}\} \\ \dots \end{array} \right\rangle, \quad (8)$$

де e_{ij} – j властивість i – об'єкту.

У відповідності до розглянутої на прикладі рис. 2. комбінації властивостей визначимо модуль контексту прийняття рішень для рекомендаційної системи на множині властивостей об'єктів, які використовує споживач.

При інтеграції контексту у відповідності до принципу локальності поєднуються можливі послідовності властивостей цікавих для споживача об'єктів (товарів, фільмів, музичних записів):

$$C = \langle \{c_1^l, \dots, c_1^l, \dots, c_1^l\}, \dots, \{c_k^l, \dots, c_k^l, \dots, c_k^l\}, \dots \rangle, \quad (9)$$

де c_k^l – реалізація локального контексту прийняття рішень для k комбінації властивостей об'єктів, що використовує l – користувач.

Для того, щоб поєднати локальні контексти на основі принципу зв'язності, задамо відношення еквівалентності, схожості та сумісності між реалізаціями локальних контекстів c_k^l .

Відношення еквівалентності визначається на властивостях цікавих для користувача об'єктів наступним чином:

$$Eq: \forall e_{ij}^* \in c_k^l \exists e_{ij}^{**} \in c_k^m : e_{ij}^* = e_{ij}^{**} \Rightarrow c_k^l \equiv c_k^m, \quad (10)$$

де c_k^l, c_k^m – k – реалізація локального контексту для l та m користувачів відповідно.

Відношення схожості визначається для ситуацій, коли в одній реалізації локального контексту властивість задана більш детально, ніж в іншій. Дане відношення задається наступним чином:

$$Ng: \forall e_{ij}^* \in c_k^l \exists e_{ij}^{**} \in c_k^m : e_{ij}^* \subseteq e_{ij}^{**} \Rightarrow c_k^l \subseteq c_k^m \mid e_{ij}^{**} \in c_k^l, c_k^m. \quad (11)$$

Відношення сумісності визначається для ситуацій, у яких існують протиріччя між представленням властивостей об'єктів у локальних контекстах різних користувачів. Приклад таких відмінностей наведено на рис. 1. Несумісність представлена властивостями e_{21}^* та e_{21}^{**} .

Відношення сумісності визначається наступним чином:

$$Cp: \forall e_{ij}^* \in c_k^l \exists e_{ij}^{**} \in c_k^m : e_{ij}^* \neq e_{ij}^{**} \Rightarrow c_k^l \setminus e_{ij}^* \equiv c_k^m \setminus e_{ij}^{**} \quad (12)$$

Модель контексту прийняття рішень, що відповідає наведеним принципам, містить у собі послідовність реалізацій локальних контекстів, а також відношення еквівалентності, схожості та сумісності між цими реалізаціями.

ВИСНОВКИ

Розглянуто проблему формування контексту як сукупності локальних контекстів споживачів в рекомендаційних системах. Дана проблема пов'язана з тим, що вибір споживачів виконується в умовах обмежень поточного локального контексту. Для підвищення точності результатів рекомендаційна система повинна

побудувати узагальнений контекст шляхом об'єднання реалізацій локальних контекстів користувачів. При побудові узагальненого контексту необхідно врахувати неповноту та протиріччя в даних.

У відповідності до представленої проблеми сформульовано принципи локальності та зв'язності, які були використані при побудові об'єднаної моделі контексту прийняття рішень в рекомендаційній системі.

Принцип локальності передбачає відмінність представлення одних і тих же властивостей об'єктів різними споживачами.

Принцип зв'язності передбачає необхідність упорядкування сукупності локальних контекстів споживачів.

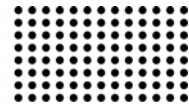
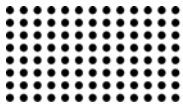
Запропоновано модель контексту прийняття рішень рекомендаційної системи, що відповідає сформульованим принципам та містить у собі послідовності реалізацій локального контексту користувачів, а також відношення еквівалентності, схожості та сумісності між цими реалізаціями.

Виділені відношення є основою для правил об'єднання локальних контекстів окремих користувачів у загальний контекст прийняття рішень в рекомендаційній системі.

В практичному плані запропонована модель дає можливість підвищити точність ранжування об'єктів та споживачів в рекомендаційних системах шляхом врахування поточного контексту прийняття рішень.

ЛІТЕРАТУРА:

1. Ricci F., Rokach L., Shapira B., Kantor P. Recommender systems handbook / F. Ricci, L. Rokach, B. Shapira, P. Kantor - Springer, New York, 2011. - 842 p.
2. Aggarwal C. C. Recommender Systems: The Textbook / C. C. Aggarwal. - Springer, New York, 2017. - 498 p.
3. Herlocker J.L. Evaluating collaborative filtering recommender systems / J.L. Herlocker, J.A. Konstan, L.G. Terveen, J.T. Riedl // ACM Transactions on Information Systems (TOIS). - 2004. - vol. 22, no.1. - P. 5-53.
4. Pazzani, M.J. Identifying Interesting Web Sites / M.J. Pazzani, J. Muramatsu, D. Billsus // Proceedings of the Thirteenth National Conference on Artificial Intelligence and the Eighth Innovative Applications of Artificial Intelligence Conference, AAAI Press, MIT Press, Menlo Park. - 1996. - P. 54-61.
5. Martín S.S. Factors determining firms' perceived performance of mobile commerce / S.S. Martín, B. López-Catalán, M.A. Ramón-Jerónimo // Ind. Manag. Data System. - 2012. - no. 112. - P. 946-963.
6. Linden G. Amazon.com recommendations: Item-to-item collaborative filtering / G. Linden, B. Smith, J. York // Internet Computing, IEEE. - 2003. - vol. 7, no1. - P. 76-80.
7. Lees-Miller J. Does Wikipedia Information Help Netflix Predictions? // J. Lees-Miller, F. Anderson, B. Hoehn, R. Greiner / Seventh International Conference on Machine Learning and Applications (ICMLA), IEEE Computer Society. - 2008. - P. 337-343. ISBN 978-0-7695-3495-4.
8. Bennet J. Lanning S. The Netflix Prize [Proceedings of KDD cup and workshop]. Available at: <http://www.netflixprize.com>. (accessed March 22, 2018).
9. Shaw, Gavin, Xu, Yue. Using Association Rules to Solveth Cold-Start Problem in Recommender Systems. [QUT Digital Repository]. URI: <http://eprints.qut.edu.au/40176>. (дата обращения 24.05.2018).



10. Sobhanam Hridya. Addressing cold start problem in recommender systems using association rules and clustering technique/ Hridya Sobhanam, A.K. Mariappan, //International Conference on Computer Communication and Informatics (ICCCI- 2013). - Coimbatore, India, 2013. - P. 402-411.
11. Adomavicius G. Towards the Next Generation of Recommender Systems/ G. Adomavicius, A. Tuzhilin// A Survey of the State-of-the-Art and Possible Extensions. IEEE Transactions on Knowledge and Data Engineering. – 2005. - no. 17. - P. 634–749.
12. Baltrunas, L. Context- Aware Places of Interest Recommendations for Mobile Users/ L. Baltrunas, B. Ludwig, S. Peer, F. Ricci //Proceedings of the 14th International Conference on Human-Computer Interaction. Berlin: Springer, 2011. - P. 531–540.
13. Baltrunas, L. Context Relevance Assessment for Recommender Systems/ L. Baltrunas, F. Ricci, B.Ludwig// Proceedings of the 2011 International Conference on Intelligent User Interfaces. - New York: Association for Computing Machinery, 2011. - P. 287–290.
14. Ono C. Context-aware preference model based on a study of difference between real and supposed situation data/ C. Ono, Y. Takishima, Y. Motomura, H. Asoh // User Modeling, Adaptation, and Personalization, 17th International Conference, UMAP 2009. -Trento, Italy, June 22-26, 2009. - P. 102–113.
15. Ekstrand M.D. Collaborative filtering recommender systems / M.D. Ekstrand, J.T. Riedl, J.A. Konstan// Foundations and Trends in Human-Computer Interaction. – 2011. - no. 4(2). - P. 81–173.

*Рецензент: д.т.н., проф. Рудакова Г.В.
Херсонський національний технічний університет*

РОЗРОБКА ПРОГРАМНИХ КОМПОНЕНТІВ ІНФОРМАЦІЙНОЇ СИСТЕМИ ЕКСТРАКЦІЇ ФАКТОГРАФІЧНИХ ДАНИХ З ВЕБ-РЕСУРСІВ

УДК 004.9:510.635

ДОРОШЕНКО Анна Юріївна

аспірант кафедри програмної інженерії ХНУРЕ,

Наукові інтереси: інтелектуальні системи, комп'ютерна лінгвістика, автоматизована обробка текстової інформації.

e-mail: yova.tanya@gmail.com.

ШАРОНОВА Наталія Валеріївна

професор, доктор технічних наук, завідувач кафедри інтелектуальних комп'ютерних систем
Національного технічного університету «ХПІ»,

Наукові інтереси: інтелектуальні системи, комп'ютерна лінгвістика, автоматизована
обробка текстової інформації, математичне моделювання.

e-mail: sharonovanv@gmail.com.

ЄНА Богдан Олександрович

студент кафедри програмної інженерії та інформаційних технологій управління Національного технічного університету «ХПІ»,

Наукові інтереси: програмна інженерія, видобування даних, інтелектуальні системи.

e-mail: enafortest@gmail.com.

ЯНГОЛЕНКО Ольга Василівна

кандидат технічних наук, асистент кафедри програмної інженерії
та інформаційних технологій управління Національного технічного університету «ХПІ»,

Наукові інтереси: інформаційні технології, веб-моніторинг, багатоагентні системи.

e-mail: olga.yan26@gmail.com.

ВСТУП

На сучасному етапі особливого значення набуває завдання збору інформації для підтримки процесів прийняття рішень, що передбачає надання фактографічної інформації. Фактографічна продукція стала першим комерційним продуктом, який почали пропонувати на інформаційному ринку. За останні роки значно виріс об'єм фактографічних запитів та їх розмаїття. Факт розглядається як знання у формі твердження, достовірність якого строго встановлена. На практиці, в сфері інформаційних технологій, фактографічну інформацію зазвичай трактують дещо інакше – як конкретні відомості або дані незалежно від того, чи є вони фактичними або прогнозованими. Головне, що ці відомості

повідомляють про якусь предметну область, а не про документи, присвячені цій області.

Велика частина вилучення знань здійснюється в спеціальних програмних системах, розроблених з нуля. Зазвичай ці рішення включають збір, аналіз, перенесення і зберігання даних. Процес проводиться розробниками з певним досвідом програмування. Розробники повинні мати справу з двома різними проблемами: технічною складністю аналізу даного документа і розумінням семантики інформації, що міститься в цьому документі. Таким чином виникла потреба в мові, яка дозволяє відображати дані при збереженні певного макета. HTML стала такою мовою Інтернету і, отже, самим прийнятим рішенням для багатьох розробників.



Проте, HTML не надає за умовчанням ніякого механізму, який полегшує автоматичний аналіз існуючих документів. Це обмеження не дозволяє відрізнити контент від макета і семантики даних.

Ми можемо ідентифікувати трьох основних учасників в будь-якій проблемі вилучення знань. Перший – джерело даних, яке містить відповідну інформацію. Другий – база даних, призначена для зберігання даних. Третій – експерт, який може визначити, як перетворити дані з джерела в базу даних. Незалежно від рівня автоматизації, роль експерта потрібна для того, щоб додати семантику щодо даних до початку вилучення. Крім того, експерт відповідає за визначення правильності та коректності вилучення даних.

При роботі з будь-яким проектом вилучення веб-знань зазвичай ігноруються такі питання, як:

1. Кількість джерел даних. Більшість підходів враховують тільки джерела даних із аналогічним дизайном (наприклад, на сторінках продуктів Amazon, або текстів на сторінках Wikipedia). Система повинна враховувати, що джерела даних не мають узагальненого дизайну, а також дані містяться в декількох джерелах інформації.

2. Багатомовність. Зазвичай існуючі підходи розглядають тільки джерела даних, написані однією мовою. Багатомовний набір джерел даних або навіть джерел даних, що використовують одночасно кілька мов не враховується в більшості підходів.

3. Різний формат даних. HTML є найбільш поширеним форматом даних, однак можуть бути присутніми інші формати, такі як XML, DOC або PDF. Це ускладнює автоматичне вилучення даних. У деяких випадках файли архівуються в різні формати (наприклад, ZIP або RAR). Також можуть виникати складнощі із застарілими форматами даних. Крім того, кодування даних також є проблемою. Не зважаючи на те, що UTF-8 є широко використовуваним стандартом, в деяких випадках сервери використовують інше кодування.

4. Оновлення та багаторазове використання. У системному проектуванні необхідно враховувати, що дані, отримані з відповідних джерел даних, розвиваються в часі. Це означає, що можна додавати більше документів, але також процесу вилучення даних може бути змінений через структурні зміни документу. При цьому більшість операцій, які виконуються в процесі вилучен-

ня, можна повторно використовувати в доменах і джерелах даних.

Таким чином, метою дослідження є розробка програмних компонентів для екстракції фактографічних даних з веб-ресурсів певного типу. Це потребує аналізу алгоритмів екстракції, сучасних програмних рішень, які реалізовані для задач видобування даних, та вибору методології і технологій вирішення проблеми.

ОГЛЯД ПРОБЛЕМИ ЕКСТРАКЦІЇ ДАНИХ

Зазвичай обробка даних, які містяться на будь-якому веб-ресурсі, передбачає певний ступінь взаємодії з людиною (заповнення форми пошуку, взаємодія зі сценарієм і т.п.). При вирішенні завдання видобування знань більшість підходів пропонує комплекс задач, пов'язаних з пошуком та визначенням необхідних даних, аналізом та зберіганням. У випадку з декількома джерелами даних складність проблеми суттєво зростає.

Кілька стандартів, таких як RDF, RDFS і OWL, були розроблені для забезпечення загального синтаксису для визначення моделей даних. Ці рішення дозволяють розробляти онтології, які підтримують запити. Ці технології зазвичай не розуміються розробниками, які спочатку ігнорують процес семантичної анотації при розробці HTML-сторінок. Щоб вирішити цю проблему, відомий підхід Schema [1], визначає словник понять, таких як люди, місця, події та продукти, які дозволяють анотувати дані, що містяться в документі HTML. Це дозволяє додати семантику в документи, які містяться в Інтернеті.

Дизайн веб-сторінок також може приховувати дані з існуючих пошукових систем. Використання динамічного контенту, CAPTCHA, приватних веб-сторінок, сценаріїв або незв'язаного контенту серед інших призводить до створення Deep Web [2]. Простим прикладом є використання веб-сторінок, які виконують пошукові запити по базі даних. Інформація, що міститься в базі даних, не може бути проіндексована пошуковою системою, оскільки для цього потрібно, щоб програмний двигун взаємодіяв з формою пошуку, задавав параметри пошуку і розумів семантику даних. Комерційні пошукові системи, такі як Google, розробляють свої інструменти з чітким акцентом на індексування так званої поверхневої мережі [3].

Екстракція знань – це створення знань з структурованих або неструктурованих джерел даних. Прямим рішенням є розробка парсеру, який обробляє документи і отримує дані, які використовуються для заповнення відповідної моделі даних. Такий підхід може бути достатнім для невеликих обсягів даних з використанням відомих структур даних. Найпростішим рішенням, прийнятим багатьма проектами, є використання XQuery [4] або регулярних виразів для вилучення точного шляху до цільового елемента. Цей підхід не дуже стійкий до структурних змін шаблону документа. Іншим популярним підходом є використання розширених перетворень мови таблиць стилів (XSLT) [5], який забезпечує уніфікований синтаксис для запису правил перетворення між сумісними з XML мовами. У базовій формі HTML в основному сумісний з XML, тому цей підхід може бути застосований до HTML. Цей підхід більш стійкий, ніж XQuery до структурних змін, але його зазвичай дуже складно налагоджувати. Ще одна проста методика, дуже практична в невеликих проектах, – це спрощена версія синтаксису HTML [6].

Для більш складних рішень використовуються спеціальні доменні мови. Рішення, подібні до тих, які представлені в [7, 8], використовують мову видобування декларативної інформації для визначення потрібних для видобування даних. Аналогічним чином [9] використовує набори правил для визначення моделі видобування знань. У цих рішеннях якість екстракції особливо залежить від навичок експертів. Друге сімейство рішень досліджує використання методів машинного навчання для поліпшення видобування інформації. Огляд існуючих методів інтелектуального аналізу веб-контенту з упором на автоматизовані і машинні методи навчання розглянуті у [10].

Автоматизований ітеративний процес для створення формального опису серії документів з використанням єдиного шаблону, який може автоматично виявляти повторювані структури і створювати структури сторінки, розглянуто в [11, 12]. Підхід IEPAD [13] використовує подібну ідею, групує теги елемента HTML в різні категорії, що знижує складність задачі. Результатом IEPAD є дерево, яке наближається до єдиного, однорідного, шаблонного документа. Автор [14] розширює цей підхід, порівнюючи подібність дерева, отримане шляхом побудови декількох дерев елементів HTML.

Одним з популярних фреймворків, що уніфікує методи машинного навчання і засновані на словниках, є GATE [15]. Цей інструментарій надає повну структуру для анотації, створення словників для іменованих об'єктів і різних методів обробки природної мови та машинного навчання, що дуже корисно для створення контрольованих підходів в Data Mining. Також останнім часом підходи DeepDive [16] привернули велику увагу дослідницького співтовариства. DeepDive використовує набір визначених правил для встановлення відношень між об'єктами. Остаточне створення бази даних – це ітеративний цикл, в якому оператор може керувати процесом машинного навчання, ідентифікуючи помилки, допущені системою. Подібним чином Google Knowledge Vault будує відношення з використанням триплетів RDF. Цей підхід показує деяку схожість з DeepDive з явним акцентом на масштабованість даних.

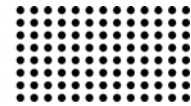
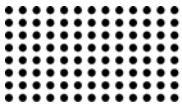
Таким чином, не зважаючи на різноманіття існуючих підходів до видобування знань питання розробки уніфікованих методів, здатних до масштабування та перенесення, працюючих в режимі реального часу та з можливістю адаптації до веб-ресурсів, які постійно змінюються, все ще залишаються відкритими.

ПОСТАНОВКА ЗАВДАННЯ

Сканування веб-сторінок є першим кроком в процесі збору даних. Елементи, що підлягають обробці, можуть відрізнитися в залежності від джерела даних. У спрощеному сценарії можна припустити, що даний продукт націлений на сканування веб-сторінок. Як сканувати ці веб-сторінки залежить від структури джерела даних. У деяких випадках сторінки легко доступні через одну URL-адресу або можуть бути отримані після виконання форми пошуку. Можна виділити наступні сценарії:

- Ідентифікація по загальнодоступному ідентифікатору. У цьому випадку кожен елемент, що підлягає обробці, визначається URL-адресою, яка містить унікальний ідентифікатор. Якщо генерація ідентифікаторів відома, можна статично генерувати список можливих URL-адрес для запиту.

- Ідентифікація по невідомому ідентифікатору. Як і в попередньому сценарії. Однак, як генеруються ідентифікатори, невідомо. В цьому випадку ідентифікатори повинні бути спочатку витягнуті з самого веб-сайту, а



потім використані для створення кінцевої URL-адреси, що підлягає обробці.

– Динамічні URL. Багато платформ розподіляють контент за динамічними URL-адресами. Це робить неможливим статично генерувати список адрес для вивчення. Цей сценарій передбачає початкову навігацію, яка запитує веб-платформу URL-адреси, а потім генерує унікальні URL-адреси, які можуть ідентифікувати ці елементи, щоб гарантувати їх унікальність в архівній системі.

Як було показано раніше, складно розробити загальне рішення для сканування, яке може бути використано в усіх сценаріях. Це пов'язано з тим, що на багатьох платформах знаходиться велика кількість коду Javascript у поєднанні з обміном повідомленнями AJAX.

Можна узагальнити деякі аспекти, які необхідно враховувати під час розробки алгоритмів обробки веб-сторінок.

– Багато веб-порталів відображають зміст, отриманий після запиту бази даних. Це означає, що не весь існуючий зміст відображається відразу, а лише невелика частина. Щоб переглянути весь існуючий контент (або посилання, які роблять його доступним), натискаємо кнопку, поки не буде відображений результат запитів. У цій операції один потік може досліджувати кожну сторінку зі списком посилань, в той час як інші можуть переміщатися по існуючим посиланням.

– Результати, які відображаються в деяких системах, є просто результатом запиту бази даних. У деяких випадках результуючий набір розбивається на сторінки, які необхідно переключати, в інших випадках в результуючому наборі є межа, яка унеможливує повне відображення існуючих результатів. У цих випадках може бути неможливо виконати сканування всіх існуючих результатів, оскільки система не розкриває цю інформацію.

– У багатьох випадках використання Javascript-рішень сильно ускладнює сканування цих систем. Найпростіше рішення – емулювати поведінку користувачів за допомогою двигунів Javascript. Проте, це накладає штрафні санкції на стороні сканування через надмірне використання ресурсів деяких браузерів.

– Щоб уникнути атак типу «відмова в обслуговуванні» (DDoS) багато платформ відстежують запити.

Перевищення певної кількості запитів може привести до тимчасового припинення послуги для користувача. Кожна платформа відрізняється, і тільки метод пробної помилки може розкривати, які заходи використовують ці платформи.

– Багато систем використовують файли cookie для зберігання ідентифікаторів сеансу, які дозволяють серверу ідентифікувати параметри запиту, які використовуються користувачем. Однак ці файли cookie мають дату закінчення терміну дії.

– Сервери можуть виходити з ладу. Багаторазове сканування цих серверів буде просто повертати код HTTP-помилки. Однак багато систем повертають веб-сторінку, що інформує про недоступність служби.

– Код типу контенту, що повертається HTTP-заголовком, особливо важливий при обході веб-сторінок, які не закодовані з використанням стандартних ASCII або UTF8 систем кодування.

Враховуючи ці особливості, завданням даної роботи є аналіз та розробка моделей та програмних компонентів для екстракції даних, які надають інформацію щодо певного факту (події або об'єкту) з веб-ресурсів певного типу. У якості прикладу будемо розглядати процес збору даних з веб-сторінок, які містять опис пропозицій мобільних телефонів. Така інформація розташована на сторінках торгівельних платформ, має однакоvu семантичну складову, але представлена різними структурами. Для успішного виконання поставленої задачі програмне рішення повинно отримувати вхідні дані для екстрактора у вигляді HTML сторінок. Додаток повинен проаналізувати та обробити ці дані для найбільш точного визначення важливості різних блоків. Далі необхідно провести екстракцію даних на основі методу обробки DOM-дерева та повернути результат, який відповідає висунутим вимогам щодо формальної моделі представлення фактографічної інформації.

ВИРІШЕННЯ ПОСТАВЛЕНОГО ЗАВДАННЯ.

ОСОБЛИВОСТІ ПРОЕКТУВАННЯ

Мета даного проекту полягає в тому, щоб автоматизувати процес вилучення точних і конкретних даних, що містяться на різноманітних веб-сторінках. Приклад загальної взаємодії системи зображено на рисунку 1.

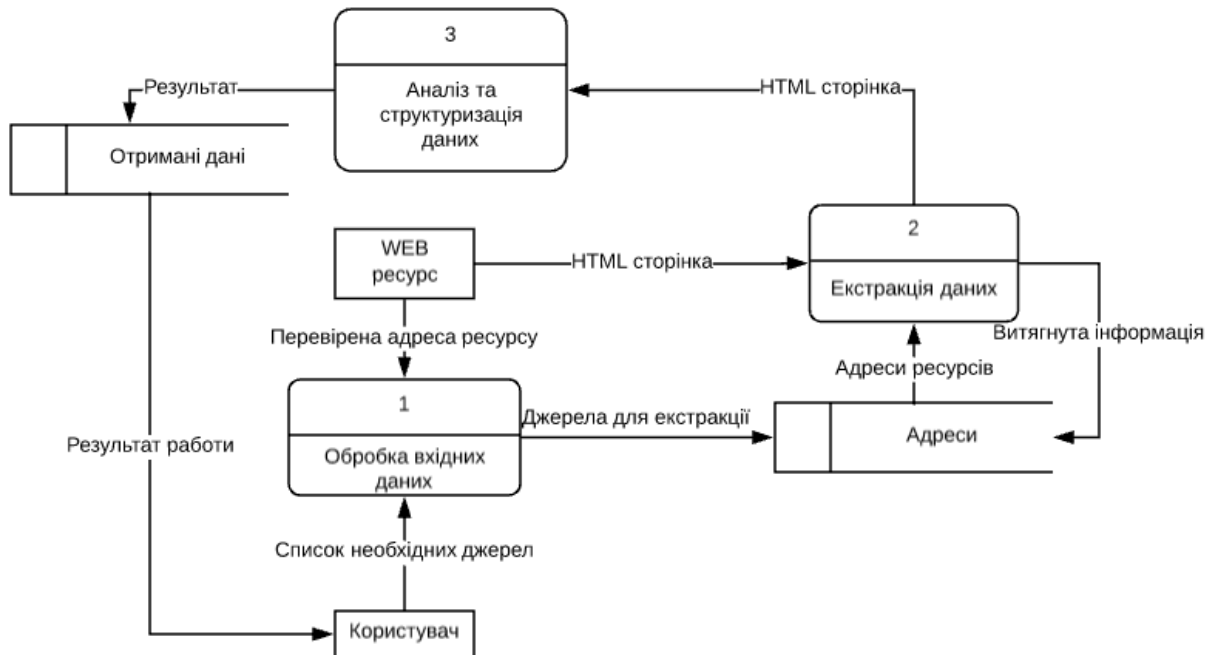


Рисунок 1 – Діаграма DFD в системі екстракції даних

Даний компонент є складним програмно-технічним комплексом. Для повноцінного його функціонування програмне забезпечення розробляється з урахуванням можливості обробки великої кількості запитів від користувачів продукту, а також його надійності, відкритості

для подальших доробок, безпеки і правильно виробленого режиму його подальшого супроводу.

На рисунках 2 і 3 зображені діаграма компонентів і діаграма прецедентів відповідно.

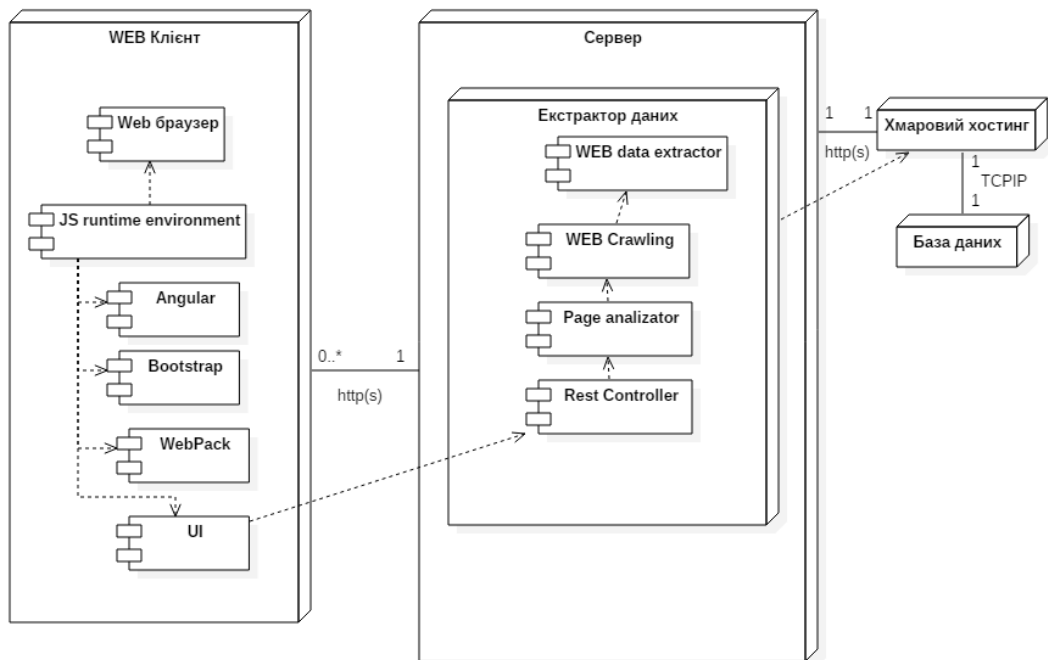


Рисунок 3.2 – Діаграма компонентів

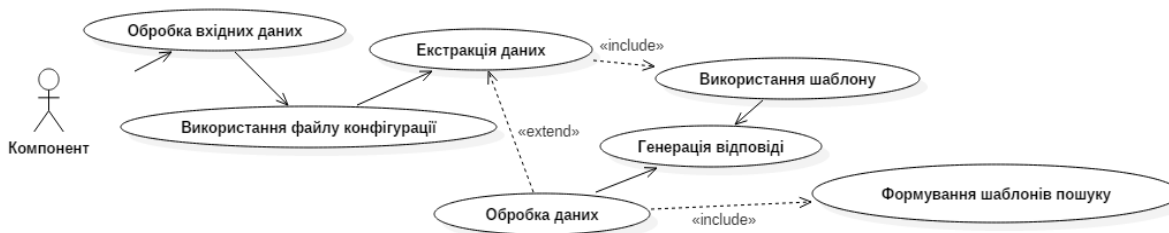


Рисунок 3 – Діаграма прецедентів

Визначимо наступні вимоги та обмеження щодо проектування програмних компонентів. Система буде отримувати адреси веб-сторінок на вході, та повертати користувачу результат роботи у тому форматі, який він заздалегідь вибере. Зв'язок між інтерфейсом та сервером буде використовувати JSON, як формат серіалізації. Компоненти системи не повинні залежати від платформи. Даний атрибут якості реалізується через використання хмарних сховищ. Завдяки цьому, не важливо на якому саме сервері буде зберігатися компонент. Якщо з'явиться необхідність перенести його з одного серверу на другий, витрати повинні бути мінімальними. Звертаючи увагу на те, що система може зростати до дуже великих розмірів, складність у конфігуруванні нових ресурсів не повинна зростати. Завдяки конфігурації екстрактора за допомогою YAML файлів, розроблюємий компонент може легко адаптуватися до екстракції з нових ресурсів без втрати великої кількості часу і сил.

Основним критерієм до вибору платформи послугувала можливість кросплатформності у використанні технології пошуку графічних елементів. Оскільки дана методологія дозволяє абстрагуватися від властивостей і технологій використаних при створенні графічного інтерфейсу. Тож серед усіх відомих мов

програмування вибір падає саме на Java. Віртуальна машина Java забезпечує кросплатформність, що поширюється на усі існуючі на сьогодні найбільш популярні платформи – Windows, Linux, Mac. Це дозволяє повноцінно користуватися перевагами графічного підходу до автоматизації GUI. Окрім кросплатформності сильними сторонами мови Java є також висока надійність роботи, розвиненість мови.

РЕЗУЛЬТАТИ

Реалізацію запропонованої архітектури наведено у вигляді діаграми класів на рисунку 4. Дана діаграма презентує основні класи, які розроблено в даному проекті, такі як: парсер, екстрактор та конфігуратор. Для досягнення необхідного результату у компонент подається URL необхідного джерела, яка оброблюється у `UrlUtils`. Якщо джерело є коректним, то далі компонент завантажує всю HTML сторінку джерела. За дану дію відповідає клас `Parser`. Після отримання компонентом веб-сторінки, він починає екстракцію даних. За екстракцію даних відповідають класи `Extractor` та `Configuration`. Екстрактор використовує екземпляр класу `Configuration` для своїх налаштувань.

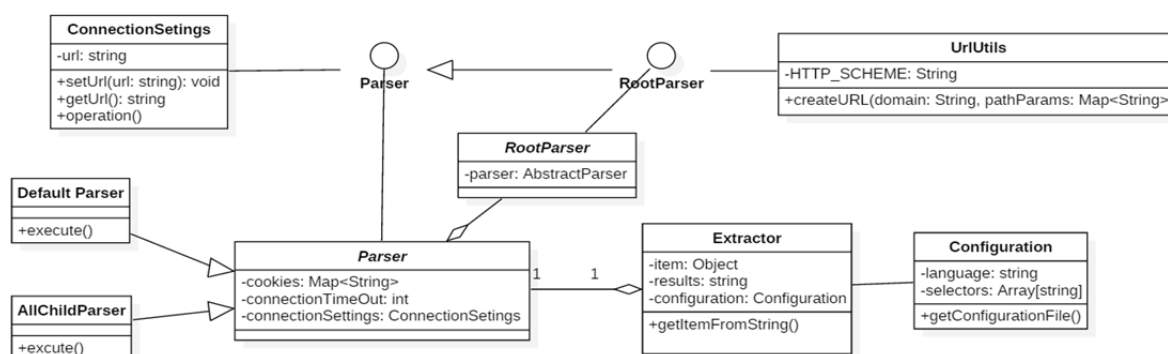


Рисунок 4 – Діаграма класів

Реалізація програмного додатку потребує тестування та валідації запропонованих рішень. Тестування – перевірка відповідності реальної поведінки програми очікуваній, що здійснюється на скінченному наборі тестів, який був обраний певним чином. У більш широкому сенсі, тестування — це одна з технік контролю якості, що включає в себе активності з планування робіт, проектування тестів, виконання тестування і аналізу отриманих результатів.

Верифікація – це процес оцінки системи або її компонентів з метою визначення чи задовольняють результати поточного етапу розробки умовам, що були сформовані на початку цього етапу. Тобто чи виконуються наші цілі, терміни, завдання по розробці проекту, визначені на початку поточної фази. Валідація – це визначення відповідності програмного забезпечення, що розроблюється, очікуванням і потребам користувача, вимогам до системи.

На першому етапі тестування необхідно провести модульне тестування усіх компонентів системи, які можуть бути протестовані окремо від інших у штучному середовищі тестування. Unit-тестування проведено за допомогою засобу автоматизації тестування Junit. Такий вибір зумовлений простотою інтеграції Junit з Java. Junit надає великий об'єм валідаційних методів, завдяки яким можна легко та ефективно тестувати як модулі обробки даних та взаємодії з базами даних, так і модулі вводу та виводу інформації.

Крім цього, враховуючи особливості реалізації трьохрівневої архітектури, необхідно провести тестування взаємодії клієнтської та серверної частин. Для цього пропонується використання сервісу Postman. Основне призначення програми Postman - створення колекцій з запитом до API. Postman дозволяє проектувати дизайн API і створювати на його основі Mock-сервер. Реалізацію сервера і клієнта можна запустити одночасно. Тестувальники можуть писати тести і виробляти автоматизоване тестування прямо з Postman. "Postman Sandbox" – це середовище виконання JavaScript, доступного при написанні "Pre-request

Script" і "Tests" скриптів. "Pre-request Script" використовується для проведення необхідних операцій перед запитом, наприклад, можна зробити запит до іншої системи і використовувати результат його виконання в основному запиті. "Tests" використовується для написання тестів, перевірки результатів, і при необхідності їх збереження в змінні.

Проведене тестування запитів за допомогою тестів, реалізованих у Postman Sandbox, показало стійку роботу розроблених компонентів.

Тестування навантаженням – підвид тестування продуктивності, збір показників і визначення продуктивності і часу відгуку програмно-технічної системи або пристрою у відповідь на зовнішній запит з метою встановлення відповідності вимогам, що пред'являються до даної системи. Для дослідження часу відгуку системи на високих або пікових навантаженнях проводиться стрес-тестування, при якому створене на систему навантаження перевищує нормальні сценарії її використання.

Apache JMeter — інструмент для проведення навантажувального тестування, що розробляється Apache Software Foundation, підпроєкту Jakarta. Хоча спочатку JMeter розроблявся як засіб тестування веб-застосунків, натеper він здатний проводити навантажувальні тести для JDBC-з'єднань, FTP, LDAP, SOAP, JMS, POP3, IMAP, HTTP і TCP. У програмі реалізовані механізми авторизації віртуальних користувачів, підтримуються користувачькі сеанси. Організовано логування результатів тесту і різноманітна візуалізація результатів у вигляді діаграм, таблиць тощо (рисунок 5).

У результаті проведених тестів отримані такі дані:

- середній час відгуку: 1135 мс;
- середня швидкість обробки: 1498 запита/хв;
- медіана часу відгуку: 455.

Таким чином, можна зробити висновок, що відповідно до отриманих даних розроблені компоненти відповідають висунутим вимогам.

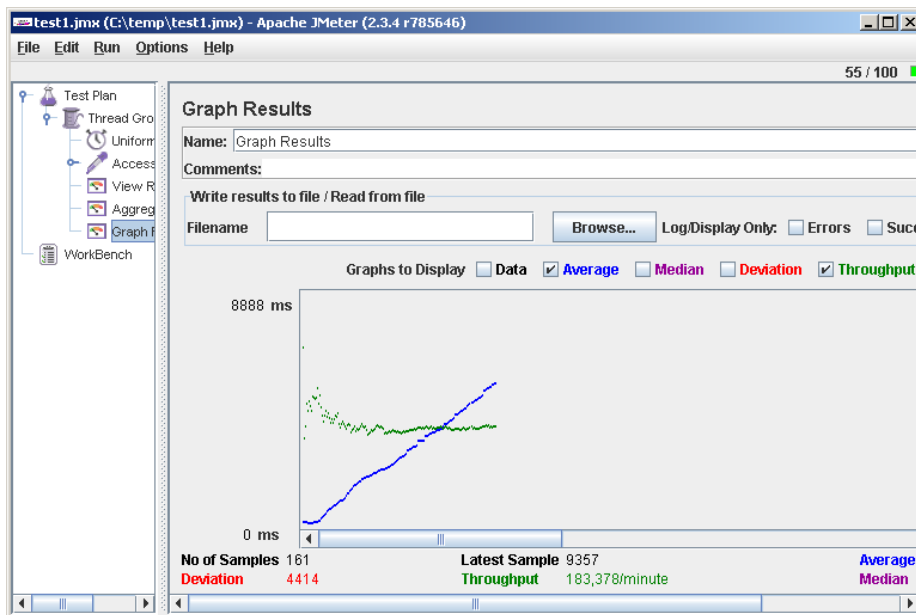


Рисунок 5 – Приклад роботи Apache Jmeter

Проведений аналіз компонента, що розроблюється з навантаженням в 100 користувачів одночасно зображений на рисунку 6.

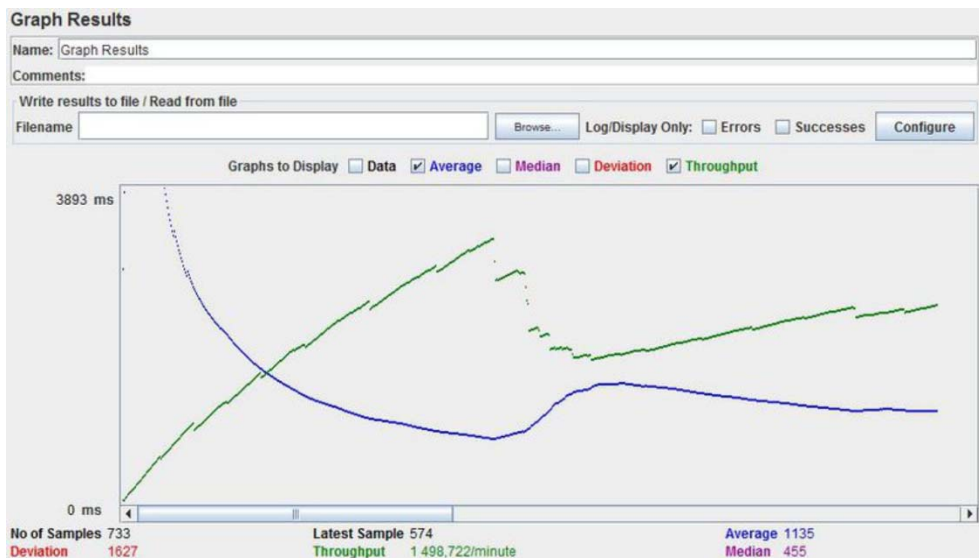


Рисунок 6 – Результати Apache Jmeter для компонента екстракції даних

ВИСНОВКИ

В результаті дослідження проведено аналіз проблеми та особливості практичної реалізації вирішення задачі екстракції фактографічних даних. Проаналізовані підходи та інформаційні технології вирішення задач парсингу на базі існуючих інформаційних систем.

Створена специфікація вимог до програмного забезпечення. Це дозволить у подальшій роботі над проектом чітко розуміти вимоги та обмеження до реалізації. Розроблено еталонну архітектуру та запропоновано варіант розгортання програмної системи.

Розроблено програмні компоненти серверної частини програмної системи, що дозволяє проводити

екстракцію даних з торговельних площадок на основі використання гнучкого конфігурування та предикатної моделі видобування даних. Розроблено та імплементовано програмні компоненти для збору даних на прикладі збору характеристик моделей мобільних телефонів. Проведено тестування розроблених компонентів та доведено їх працездатність для збору даних з трьох різних торговельних площадок. Слід зазначити, що збір даних налаштовано тільки для англійської мови, але

проведене тестування компонентів дозволяє стверджувати, що отримані результати можуть бути базисом створення програмної системи для видобування фактографічної інформації. Запропоновано архітектурне рішення може бути розвинене за рахунок імплементування моделі екстракції необхідної інформації. Додаткові джерела даних можуть бути додані в систему за допомогою YAML конфігурації.

ЛІТЕРАТУРА

1. R. V. Guha, Dan Brickley, and Steve Macbeth. Schema.org: Evolution of structured data on the web. *Commun. ACM*, 59(2):44–51, January 2008.
2. Sriram Raghavan and Hector Garcia-Molina. Crawling the hidden web. In *Proceedings of the 27th International Conference on Very Large Data Bases, VLDB '01*, pages 129–138, San Francisco, CA, USA, 2001. Morgan Kaufmann Publishers Inc.
3. Memex (Domain-Specific Search) // URL: www.darpa.mil/program/memex, 02.11.2017.
4. W3C XML Query (XQuery) // URL: <https://www.w3.org/XML/Query/>, 04.11.2017.
5. XSL Transformations (XSLT) Version 3.0 // URL: <https://www.w3.org/TR/xslt>, 11.2017.
6. Apache Nutch™ // URL: <http://nutch.apache.org/>, 18.11.2017.
7. Warren Shen, AnHai Doan, Jeffrey F. Naughton, and Raghu Ramakrishnan. Declarative information extraction using datalog with embedded extraction predicates. In *Proceedings of the 33rd International Conference on Very Large Data Bases, VLDB '07*, pages 1033–1044. VLDB Endowment, 2007.
8. Scrapy | A Fast and Powerful Scraping and Web Crawling Framework // URL: <http://scrapy.org/>, 25.11.2017.
9. Ndapandula Nakashole, Martin Theobald, and Gerhard Weikum. Scalable knowledge harvesting with high precision and high recall. In *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining, WSDM '11*, pages 227–236, New York, NY, USA, 2011. ACM.
10. Xin Luna Dong, Evgeniy Gabrilovich, Jeremy Heitz, Wilko Horn, Kevin Murphy, Shaohua Sun, and Wei Zhang. From data fusion to knowledge fusion. *Proc. VLDB Endow.*, 7(10):881–892, June 2014.
11. Oren Etzioni, Michael Cafarella, Doug Downey, Stanley Kok, Ana-Maria Popescu, Tal Shaked, Stephen Soderland, Daniel S. Weld, and Alexander Yates. Web-scale information extraction in knowitall: (preliminary results). In *Proceedings of the 13th International Conference on World Wide Web, WWW '04*, pages 100–110, New York, NY, USA, 2004. ACM.
12. Andrew Carlson, Justin Betteridge, Bryan Kisiel, Burr Settles, Estevam R. Hruschka Jr., and Tom M. Mitchell. Toward an architecture for never-ending language learning. In *AAAI*. AAAI Press, 2010.
13. Bing Liu and Kevin Chen-Chuan-Chang. Editorial: special issue on web content mining. *Acm Sigkdd explorations newsletter*, 6(2):1–4, 2004.
14. B Anantha Barathi. Structured information extraction system from web pages. *MiddleEast Journal of Scientific Research*, 19(6):817–820, 2014.
15. Arvind Arasu and Hector Garcia-Molina. Extracting structured data from web pages. In *Proceedings of the 2003 ACM SIGMOD international conference on Management of data*, pages 337–348. ACM, 2003.
16. Chia-Hui Chang and Shao-Chen Lui. Iepad: information extraction based on pattern discovery. In *Proceedings of the 10th international conference on World Wide Web*, pages 681–688. ACM, 2001.
17. Postman | API Development Environment // URL: www.getpostman.com, 23.03.2018.

*Рецензент: д.т.н., проф. Марасанов В. В.
Херсонський національний технічний університет*



АНАЛІЗ РОЛІ ТА МІСЦЯ, СУТНОСТІ ТА ЗМІСТУ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ НА СУЧАСНОМУ ЕТАПІ РОЗВИТКУ НАУКИ І ТЕХНІКИ

УДК 004

БРОДСЬКИЙ Юрій Борисович

кандидат технічних наук, доцент, завідувач кафедри комп'ютерних технологій і моделювання систем,
Житомирський національний агроєкологічний університет.

Наукові інтереси: математичне моделювання, інформаційні технології.

e-mail: yubrodskiy26@gmail.com

ГРИЦУК Руслан Валентинович

доктор технічних наук, старший науковий співробітник, начальник відділу інформаційної
та кібернетичної безпеки наукового центру, Житомирський військовий інститут ім. С.П. Корольова.

Наукові інтереси: інформаційна безпека; кібербезпека; інформаційні технології.

Постановка проблеми в загальному вигляді та її зв'язок із важливими практичними завданнями. Глобалізаційні процеси початку XXI століття обумовили формування суспільства нового типу, яке прийнято називати інформаційним [1, 2]. Рушійною силою глобалізаційних процесів при цьому в основному виступає галузь інформаційної індустрії, яка розвивається на базі інформаційних технологій (ІТ).

Так сучасні ІТ відкрили нові можливості для використання інформаційних ресурсів різного цільового призначення в інтересах людини й суспільства зокрема, та держави в цілому. Тому ІТ, продуктом яких є інформаційні ресурси, є дієвим каталізатором процесів, що відбуваються в усіх без винятку сферах діяльності суспільства та держави. При цьому особливо важлива роль ІТ відводиться в безпекових сферах, наприклад, інформаційній, економічній, технологічній, воєнній, техногенній, екологічній та ін.

В умовах катастрофічного погіршення екологічної безпеки в світі та в Україні зокрема, нині на перший план виходять ті ІТ, які використовуються або можуть бути потенційно використані для геофізичного моніторингу небезпечних геофізичних явищ природного та інколи, як виняток, й техногенного походження. Отже,

зважаючи на зазначене на сучасному етапі розвитку науки і техніки виникає нагальна потреба перегляду ролі та місця ІТ з подальшим уточненням їх сутності та змісту в інтересах створення дієвих моделей систем геофізичного моніторингу.

Аналіз останніх досліджень і публікацій [3–24] та ін. за темою дослідження показав, що вивченню ролі та місця ІТ приділяється значна увага академічної спільноти. При цьому їх сутність та зміст розкриваються й на рівні міжнародних стандартів та нормативно-правових документів, наприклад [25] та [26] відповідно.

Характерною спільною рисою зазначених досліджень є те, що й досі відсутнє єдине бачення на роль та місце ІТ. Причиною тому, як випливає з аналізу [3–24], є дефініційна неузгодженість. Кожен автор, трактує поняття ІТ в аспекті власної системи світогляду, прикладної галузі застосування ІТ та ряду інших зовнішніх та внутрішніх факторів.

З метою вирішення виявлених протиріч в 2015 р. було введено міжнародний стандарт серії ISO/IEC 38500 [25], який виступив дієвим підґрунтям для єдиного розуміння сутності та змісту ІТ. Поряд з тим його використання на практиці потребує певної адаптації до тієї

сфери в рамках якої створюються ІТ, а також його вимоги потребують узгодження з чинними нормативними та нормативно-правовими актами тієї держави, в якій вони розвиваються. Зокрема для України одним з таких документів виступає Національна програма інформатизації [26]. Таким чином, невирішеною частиною загальної проблеми побудови прогресивних ІТ в інтересах забезпечення безпеки держави в її визначальних сферах є системний аналіз їх ролі та місця, сутності та змісту на сучасному етапі розвитку науки і техніки, що є **метою статті**.

Викладення основного матеріалу дослідження. Ґрунтовний системний аналіз, методологію якого пропонується використати для досягнення поставленої мети, передбачає: по-перше, встановлення та аналіз першопричин виникнення та змінювання власне самої категорії “інформаційна технологія” в авторському її розумінні – для встановлення її сутності та змісту; по-друге, визначення ролі та місця ІТ на сучасному етапі.

Визначення сутності та змісту інформаційної технології. Виходячи з аналізу [3–24] можна стверджувати, що на формування сутності та змісту ІТ на сучасному етапі її становлення впливає тріада, базис якої становить суспільство, наука і техніка (рис. 1).



Рис. 1. Базова тріада, що становить сутність та зміст ІТ на сучасному етапі

Зроблене вище припущення ґрунтується на тому, що з одного боку – інформаційна криза суспільства, як зазначає Д. Мартін, яка обумовлена протиріччям між накопиченим у світі суттєвим інформаційним потенціалом і неможливістю його використати внаслідок природних та штучних (соціально-економічних, політичних, технічних) обмежень можливостей пошуку, одержання, сприйняття, зберігання, оброблення, фільтрації та передачі інформації. З іншого боку – вплив науки, і в першу чергу – кібернетики Н. Вінера [10, 11], теорії інформації і зв’язку К. Шеннона [9], теорії систем Л. фон Берталанфі, розвиток обчислювальної техніки – перші універсальні електронні обчислювальні машини (ЕОМ) ENIAC, EDVAC, що побудовані на принципах Д. фон Неймана (1946 - 51 pp.), а також створення розвинутої

мережі телекомунікацій – ARPANET (1969 р.) та Internet (1983 р.). Таким чином, приведена вище історична ретроспектива дозволила встановити, що головними факторами суспільних перетворень на кожному з етапів ставали інформаційні процеси, виробництво обчислювальної техніки та використання інформації. Як наслідок, відбуваються зміни в концепції розвитку світової економіки, де ключову роль починає відігравати сфера інформаційних послуг, а матеріально-технологічна індустрія поступається місцем індустрії інформаційній. Тому, мабуть природно, що вперше категорія “інформаційна технологія” уживалася в соціально-економічних літературних джерелах, а саме у 1958 р. в статті професорів Стенфордського університету Г. Лівітта та Т. Уіслера “Management in the 1980’s”, де



зокрема зазначено, що *"The new technology does not yet have a single established name. We shall call it information technology."* [3].

Для аналізу понятійно-категорійного апарату скористаємось методом декомпозиції і проведемо аналіз дефініції самої категорії. Оскільки дефініент складається з двох компонентів, дослідимо їх етимологію і семанти-

ку. Дослідимо спочатку сутність та зміст категорії "технологія", а потім її уточнюючу ознаку – "інформаційна".

Термін "технологія" (з грец. *τεχνη* – майстерність, *техніка*; *λογος* – слово, наука) запровадив у 1772 р. німецький вчений, професор Геттінгенського університету Й. Ф. Бекман [4]. Систематизувавши інші більш поширені погляди на тлумачення даної категорії, подамо їх у вигляді табл. 1.

Таблиця 1

Сутність та зміст категорії "технологія"	Джерело
"Огляд винаходів, їхнього розвитку та успіхів у мистецтвах і ремеслах може називатися історією технічних мистецтв; технологія, яка пояснює в цілому, методично і точно всі види праці з їхніми наслідками й причинами, являє собою набагато більше".	Й. Ф. Бекман [4]
"Технологія – сукупність прийомів і способів одержання, обробки або переробки (зміни стану, властивостей, форми) сировини, матеріалів, напівфабрикатів чи виробів у різних галузях промисловості, в будівництві тощо; наукова дисципліна, що розробляє і вдосконалює ці прийоми і способи".	Українська радянська енциклопедія [5].
1. Сукупність знань, відомостей про послідовність окремих виробничих операцій у процесі виробництва чогонебудь. 2. Навчальний предмет, що викладає ці знання, відомості. 3. Сукупність способів обробки або переробки матеріалів, виготовлення виробів, проведення різних виробничих операцій тощо".	Тлумачний словник сучасної української мови [6].
"Під технологією я розумію використання наукового знання для визначення способів виготовлення речей у відтвореній манері. Технологія є ресурсним потенціалом розвитку суспільства, який представляє різні варіанти соціальних змін".	М. Кастельс [7].
"У широкому сенсі під технологією розуміють науку про закони виробництва матеріальних благ, вкладаючи в неї три основні частини: ідеологію, тобто принципи виробництва; знаряддя праці, тобто верстати, машини, агрегати; кадри, які володіють професійними навичками. Ці складові називають інформаційною, інструментальною і соціальною. Для конкретного виробництва технологію розуміють у вузькому сенсі як сукупність прийомів і методів, що визначають послідовність дій для реалізації виробничого процесу".	Б. Советов [8].

Аналіз приведених вище (див. табл. 1) та ін. тлумачень категорії "технологія" дозволяє зробити висновок про їх схожість і несуперечливість одне одному. Узагальнивши їх сутність відповідно до змісту, виділимо основні характерні риси та властивості даної категорії: по-перше, технологія – це процес досягнення поставленої мети, одержання продукції; по-друге, технологія – це сукупність знань про способи та інструменти виробництва, наукова дисципліна, що їх розробляє та вдосконалює; по-третє, технологія володіє такими основними властивостями та ознаками, як оптимальність, ефективність, алгоритмічність тощо.

Аналіз дефініції "інформаційна" показує, що її основою становить категорія "інформація". Для додержання коректності при дослідженні категорії "інформація" слід зауважити, що розвиток науки і техніки за останні сімдесят років характеризується розповсюдженням саме інформаційного підходу. Ядром такого підходу власне і виступає "інформація". Не зважаючи на зазначене, дана

категорія і на сьогодні є тим феноменом, який потребує всебічного аналізу та вивчення.

Як відомо з [9], К. Шеннон в рамках створеної ним теорії інформації опираючись на праці Г. Найквіста та Р. Хартлі основоположною категорією вважав саме інформацію. Узагальнивши праці відомих вчених, таких як Р. Клаузіуса, Л. Больцмана, К. Шеннона, Е. Вебера, Г. Фехнера та ін. видатних вчених, у тому числі й сучасних, які оперували категорією "інформація" в табл. 2 для узагальнення приведемо лише найвідоміші тлумачення. При цьому також врахуємо і діючу національну нормативно-правову базу [18, 19].

Узагальнюючи зазначене вище (див. табл. 2), та ін. відомі дослідження у зазначеній царині тлумачення категорії "інформація", можна зробити висновки: по-перше, інформація є загальнонауковою філософською категорією і на сьогодні є об'єктивною реальністю разом із такими фундаментальними поняттями як матерія та енергія; по-друге, інформація нині є мірою відображення різноманіття реального світу; по-третє, інфо-

рмация є способом опису взаємодії отримувача з оточуючим середовищем; по-четверте, інформація є ха-

ктеристикою співвідношення між повідомленням і його користувачем.

Таблиця 2

Сутність та зміст категорії "інформація"	Джерело
Значення інформації і методика її вимірювання та передачі складає цілий предмет вивчення для інженера, фізіолога, психолога і соціолога. Інформація є інформація, а не матерія чи енергія [10].	Н. Вінер [10,11]
Інформація – це позначення змісту, який черпається нами із зовнішнього світу в процесі нашого пристосування до нього і приведення у відповідальність з ним нашого мислення [11].	У. Ешбі [12]
Сутність феномена інформації зводиться до різноманіття матеріального світу, а кількість інформації відображає міру різноманіття.	В. Глушков [13]
Інформація в загальному її розумінні представляє собою міру неоднорідності розподілу матерії та енергії у просторі і часу, міру змін, якими супроводжуються всі процеси, що протікають у світі.	А. Урсул [14]
Інформація, з позицій теорії відображення, може бути подана як відбите різноманіття, а саме різноманіття, яке один об'єкт містить про інший об'єкт.	К. Колін [15]
Інформація представляє собою загальну фундаментальну властивість реальності, яке проявляється в тому, що окремі фрагменти реальності по різному проявляють себе у просторі та часу, тобто володіють властивістю відмінності. Сукупність цих відмінностей і є інформація.	Е. Гойхман, Ю. Лосев [16]
Під інформацією розуміють відомості про будь-яку подію або предмет, що поступають до отримувача зовні в результаті його взаємодії з оточуючим середовищем.	В. Тростніков [17]
Інформація є характеристика не повідомлення, а співвідношення між повідомленням та його користувачем. Інформація є не матеріальна сутність, а спосіб опису взаємодії.	Закон України "Про інформацію" [18]
Інформація – це документовані або публічно оголошені відомості про події та явища, що відбуваються в суспільстві, державі та навколишньому природному середовищі.	ДСТУ 2226-93 [19]
Інформація – відомості про суб'єкти, об'єкти, явища та процеси.	

Отже, авторський підхід до виникнення інформації у сучасному її розумінні на прикладі системи геофізичного моніторингу можна подати у вигляді схеми (рис. 2).

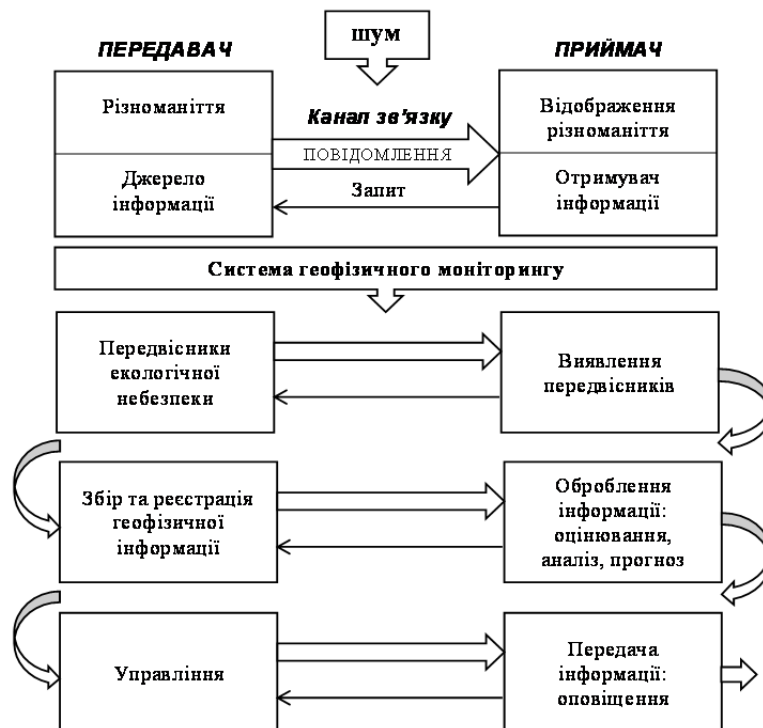


Рис. 2. Технологія зародження інформації в системі геофізичного моніторингу

На рис. 2 сутність зародження інформації в системі геофізичного моніторингу подано як процес взаємодії двох абонентів, що здійснюється через канал зв'язку: передавача і приймача, де індикатором інформації виступає зміна стану приймача, як реакція на вплив різноманіття через канал зв'язку.

Таким чином, комплексуючи складові категорії "інформаційна технологія", ґрунтовний аналіз яких приве-

дено вище, виділимо їх спільні характерні риси. При цьому, зауважимо, що звичайне тривіальне поєднання елементів "технологія" та "інформація", а також формальні аналогії матеріально-виробничої технології з інформаційною, дійсно відображають одну спільну основу – це процес, що спрямований на досягнення поставленої мети. Такою метою, як правило, є перероблення відповідного ресурсу в заданий продукт (рис. 3).

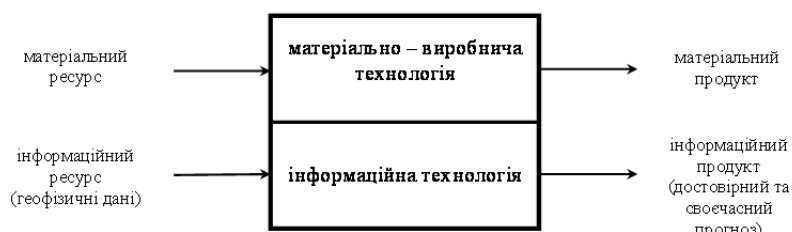


Рис. 3. Аналогія між матеріально-виробничою технологією та інформаційною технологією

Як видно з рис. 3 проста аналогія, як науковий метод дослідження, при встановленні сутності та змісту категорії "інформаційна технологія" є неприйнятним. Тому опираючись на методи системології [2] та комплексуючи різні відомі підходи до визначення досліджуваної категорії в табл. 3 приведемо тлумачення її сутності та змісту на сучасному етапі.

Таблиця 3

Сутність та зміст категорії "інформаційна технологія"	Джерело
ІТ складається із кількох взаємозв'язаних частин. Перша включає методи оброблення великих обсягів інформації за допомогою швидкісних комп'ютерів. Друга зосереджується навколо використання статистичних та математичних методів вирішення проблем прийняття рішень і представлена методами математичного програмування та дослідження операцій. Третя – комп'ютерне моделювання штучного інтелекту.	Г. Лівітт, Т. Уїслер [3]
ІТ – процеси, де основною перетворюваною продукцією є інформація.	В. Глушков [20]
ІТ – це поданий в проектній формі концентрований вираз наукових знань та практичного досвіду, що дозволяє раціонально організувати той чи інший інформаційний процес, який достатньо часто повторюється. При цьому досягається економія затрат праці, енергії або матеріальних ресурсів необхідних для реалізації даного процесу.	К. Колін [21]
"ІТ – сукупність процесів, що використовує засоби та методи накопичення, обробки і передачі первинної інформації для отримання інформаційного продукту (інформації нової якості про стан об'єкту, процесу або явища). ІТ повинна давати можливість оцінити вплив рішень, які приймаються, на підвищення ефективності процесів, які вона обслуговує. Основною метою ІТ є повне і своєчасне задоволення інформаційних потреб користувачів".	О. Томашевський [22]
ІТ – сукупність впроваджуваних в системи організаційного управління принципово нових засобів і методів обробки даних, що поєднуються в цілісні технологічні системи та забезпечують цілеспрямоване створення, передачу, збереження та відображення інформаційного продукту (даних, ідей, знань) з найменшими затратами у відповідності з закономірностями соціального середовища, в якому розвивається нова інформаційна технологія.	В. Гриценко, Б. Паньшин [23, 24]
ІТ – ресурси, що використовуються для збору, обробки, зберігання та розповсюдження інформації.	ISO/IEC 38500: 2015 [25]
"ІТ – цілеспрямована організована сукупність інформаційних процесів з використанням засобів обчислювальної техніки, що забезпечують високу швидкість оброблення даних, швидкий пошук інформації, розосередження даних, доступ до джерел інформації незалежно від місця їх розташування".	Закон України Про Національну програму інформатизації [26]
ІТ – технологічний процес, предметом перероблення й результатом якого є інформація.	ДСТУ 2226-93 [19]

Таким чином, як показано в табл. 3 поняття ІТ досить багатогранне, тому за зовнішнім різноманіттям підходів до його визначення проглядаються такі основні смислові домінанти. Перша – це *мета*, результат

технологічного процесу – одержання інформації для своєчасного задоволення інформаційних потреб користувача, забезпечення можливості прийняття оптимальних управлінських рішень та, у загальному випадку

ку, – підвищення ефективності системи-замовника інформаційного продукту. Друга домінанта – *це процес організації досягнення мети*. З одного боку – сукупність впроваджених засобів і методів збору, зберігання, оброблення та передачі інформації, даних (дані – предмет технологічного процесу), з іншого – “концентрований вираз наукових знань та практичного досвіду” [21]. Третя домінанта визначає *принцип оптимальності* як гармонійного поєднання оптимізації технологічного процесу і мінімізації витрат на створення нової інформаційної технології.

Отже, в результаті проведеного аналізу та з урахуванням наведених визначень поняття “інформаційна технологія”, на сучасному етапі розвитку науки і техніки пропонується таке її тлумачення:

Інформаційна технологія – це цілеспрямований оптимальний технологічний процес збирання, зберігання, оброблення, відображення, передачі та розповсюдження інформації за допомогою сучасних програмно-технічних засобів на основі наукових методів та практичного досвіду.

Визначення ролі та місця ІТ (на прикладі сучасних систем геофізичного моніторингу). Сучасну систему моніторингу (за концепцією Ю. Ізраеля) можна розглядати як інформаційно-управляючу систему, що забезпечує розв’язування таких основних задач: *спостереження за станом довкілля та факторами впливу, оцінювання та аналіз фактичного стану довкілля, прогнозування небезпечних явищ і катастроф та управління – своєчасної підготовки наукової інформації та обґрунтування рекомендацій щодо прийняття ефективних рішень.* Центральне місце в цій системі відведено саме *інформаційним технологіям*, як процесу збирання, зберігання, оброблення, відображення, передачі та розповсюдження геофізичної інформації.

Відомі світові проекти систем моніторингу небезпечних геофізичних явищ (Глобальна система моніторингу навколишнього середовища – GEMS, Глобальна система систем спостереження Землі – GEOSS, Система попередження про катастрофи та стихійні лиха – Sentinel Asia, “Міжнародна Хартія “Space and Major

Disasters” та ін.) представляють собою великі багатокмponentні комплекси, які поєднують в собі космічні, аерокосмічні та наземні засоби моніторингу. Проведений аналіз відкритих джерел інформації дозволив встановити особливості таких систем, їх переваги та недоліки. Узагальнивши результати аналізу, можна зауважити, що існуючі системи геофізичного моніторингу спрямовані в першу чергу на *спостереження та оцінювання фактичного стану довкілля, констатації факту і повідомлення про виникнення конкретного природного явища.* Задачі *прогнозування* зародження небезпечних геофізичних процесів, підготовки достовірної інформації та своєчасного попередження про можливість виникнення катастрофи практично не виконуються. Найбільш вагомою проблемою є короткостроковий та оперативний прогноз з точним визначенням дати, часу та місця катастрофічного явища.

Таким чином, в умовах погіршення екологічної безпеки зростає актуальність моніторингу небезпечних природних та антропогенних явищ і виникає гостра необхідність в забезпеченні саме задач достовірного *прогнозування та ефективного управління*, що суттєво підвищує роль ІТ для аналізу та синтезу моделей систем геофізичного моніторингу.

Висновки та перспективи подальших досліджень. У статті показано, що на сучасному етапі розвитку науки і техніки роль та місце, сутність та зміст ІТ суттєво залежать від галузі її практичного застосування. Поряд з тим ІТ нині розглядається як складова самостійної фундаментальної науки, що має свій власний науково-категорійний апарат. У статті доведено, що ІТ, як наукова галузь, спирається на фундаментальні положення теорії кібернетики, теорії інформації, теорії систем та прикладної математики та інших супутніх їм теорій. Саме тому, як випливає з одержаних результатів, ІТ є також і смисловою системною категорією, що відображає такі домінанти як мета, процес і оптимальність. Одержані результати виступають підґрунтям для розроблення нових моделей систем геофізичного моніторингу, основу яких становлять сучасні ІТ.

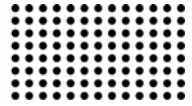
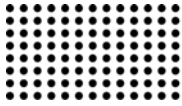
ЛІТЕРАТУРА:

1. Lipkan V.A. Pravovi zasady rozvytku informatsiinoho suspilstva v Ukraini : monohrafiia / V.A. Lipkan, I. M. Sopilko, V.O. Kirian ; za zah. red. V.A. Lipkana; Hlob. orh. soiuzn. liderstva, Akad. bezpeky vidkryt. susp-va, Akad. nauk vyshch. osvity Ukrainy. - Kyiv : FOP Lipkan O. S., 2015. - 664 s.



2. Brodskiy Yu.B. Informatyka ta systemolohiia: navch. posibnyk / Yu.B. Brodskiy, K.V. Molodetska. – Zhytomyr: ZhNAEU, 2014. – 244 s.
3. Leavitt. Harold J. and Thomas L. Whistler. Management in the 1980's. Harvard Business Review, November – December, 1958. – P. 41-48.
4. Salomon J. What is Technology? The Issue of its origins and definitions // History and technology. 1984, Vol. 1. – p. 113-156.
5. Ukrainska radianska entsyklopediia: u 12-ty t. /Hol red. M.P. Bazhan; red. kol.: O.K. Antonov ta in. – 2-he vyd. – K.: Holovna redaktsiia URE, 1974-1985.
6. Tlumachnyi slovnyk suchasnoi ukrainskoi movy. /Ukladachi L.P. Kovryha, T.V. Kovalova, V.D. Ponomarenko./ Za red. doktora filolohichnykh nauk, prof. V. S. Kalashnyka. - Kharkiv: Belkar-knyha, 2005. – 800 s.
7. M. Kastels. Informatsionnaya epoha: ekonomika, obschestvo i kultura: Per. s angl. Pod nauch. red. O.I. Shkaratana. M.: GU-VShE, 2000.
8. Sovetov B. Ya. Informatsionnaya tehnologiya: Uchebnik / B. Ya. Sovetov. – M.: Vysshaya shkola, 1994. – 368 s.
9. Shennon K. Matematicheskaya teoriya svyati. V kn.: Raboty po teorii informatsii i kibernetike. M.: IL, 1963.
10. Viner N. Kibernetika ili upravlenie i svyaz v zhyvotnom i mashine. – M.: Sovetskoe radio, 1968.
11. Viner N. Kibernetika i obschestvo. – M.: IL, 1958.
12. Eshbi U. R. Vvedenie v kibernetiku. – M.: IL, 1959.
13. Glushkov V. M. O kibernetike kak nauke // Kibernetika, myshlenie, zhizn. – M.: Myisl, 1964. – S. 53-62.
14. Ursul A. D. Informatsiya. Metodologicheskie aspekty. – M.: Nauka, 1971. – 296 s.
15. Kolin K. K. Filosofiya informatsii: struktura realnosti i fenomen informatsii // Metafizika. – 2013. - № 4 (10). – S. 61-84.
16. Goyhman E. Sh., Losev Yu. I. Peredacha informatsii v ASU. Izd. 2-e, dop. i pererab. M.: Svyaz, 1976. – 280 s.
17. Trostnikov V. N. Chelovek i informatsiya. – M.: Nauka, 1970.
18. Pro informatsiiu: Zakon Ukrainy vid 02.10.1992, №48-VR // Vidomosti Verkhovnoi Rady Ukrainy (VVR). – 1992. – №48. – st. 651.
19. Derzhavnyi standart Ukrainy. Avtomatyzovani systemy. Terminy ta vyznachennia. DSTU 2226-93.
20. Glushkov V. M. Osnovy bezbumazhnoy informatsii. – M.: Nauka, 1987. – 552 s.
21. Kolin K. K. Informatsionnaya tehnologiya kak nauchnaya distsiplina / K. K. Kolin // Informatsionnaya tehnologiya. – 2001. – № 2. – S. 2-10.
22. Tomashevskiy O. M., Tsehelyk H. H., Viter M. B., Dudukh V. I. Informatsiini tekhnologii ta modeliuvannia biznes-protsesiv. Navch. Pos. – K.: «vydavnytstvo «Tsentр uchbovoi literatury», 2012. – 296 s.
23. Gritsenko V. I. Informatsionnaya tehnologiya: voprosy razvitiya i primeneniya / V. I. Gritsenko, B. N. Panshin. – Kiev, 1988. – 265 s.
24. Gritsenko V. I. Ishodnaya kontseptsiya i opredelenie ponyatiya informatsionnoy tehnologii / V. I. Gritsenko, B. N. Panshin // Informatsionnoe obschestvo. – 1990. – # 2. – S. 67-76.
25. Information technology – Governance of IT for the organization: ISO/IEC 38500: 2015 (en). – (Міжнародний стандарт). – Режим доступу: <https://www.iso.org/obp/ui/ru/#iso:std:iso-iec:38500:ed-2:v1:en>.
26. Pro Natsionalnu prohramu informatyzatsii: Zakon Ukrainy vid 04.02.1998, №74/98-VR // Vidomosti Verkhovnoi Rady Ukrainy (VVR). – 1998. – №27-28. – st. 181.

*Рецензент: д.т.н., проф. Ходаков В.Є.
Херсонський національний технічний університет*



АВТОМАТИЧНЕ ПРОЕКТУВАННЯ СХЕМ РОЗКРОЮ ЛИСТОВИХ МАТЕРІАЛІВ НА ПЛОСКІ ГЕОМЕТРИЧНІ ОБ'ЄКТИ

УДК 685.34.01

ЧУПРИНКА Віктор Іванович

доктор технічних наук, професор, Київський національний університет технологій та дизайну, кафедра інформаційних технологій,

Наукові інтереси: задачі, що пов'язані з проектуванням раціональних схем розкрою матеріалів на плоскі геометричні об'єкти зі складною конфігурацією зовнішнього контуру; автоматизоване проектування виробів легкої промисловості.

e-mail:Chuprinka_V_I@ukr.net

ЗЕЛІНСЬКИЙ Григорій Юрійович

Київський національний університет технологій та дизайну, кафедра інформаційних технологій, аспірант,

Наукові інтереси: задачі, що пов'язані з проектуванням раціональних схем розкрою матеріалів на плоскі геометричні об'єкти зі складною конфігурацією зовнішнього контуру

e-mail:gzelinskiy@gmail.com

ЧУПРИНКА Наталія Вікторівна

кандидат технічних наук, Київський національний університет технологій та дизайну, кафедра інформаційних технологій,

Наукові інтереси: задачі, що пов'язані з проектуванням раціональних схем розкрою матеріалів на плоскі геометричні об'єкти зі складною конфігурацією зовнішнього контуру; автоматизоване проектування шкіргалантерейних виробів

e-mail:natasha-chup@ukr.net

ВСТУП

Автоматичне проектування раціональних розкрійних схем матеріалів дозволить ефективно використовувати матеріали при розкрої, зменшити кількість відходів, що забруднюють навколишнє середовище, знизити собівартість виробів. Тому завдання автоматичного проектування раціональних схем щільного розміщення плоских геометричних об'єктів з складною конфігурацією зовнішніх контурів на прямокутній області Ω заданих розмірів є актуальним завданням.

Мета роботи: розробити метод автоматичного проектування раціональних схем щільного розміщення плоских геометричних об'єктів з складною конфігурацією зовнішніх контурів в прямокутній області заданих розмірів.

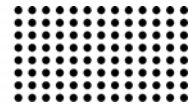
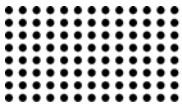
В більшості випадків процес підготовки раціональних схем розкрою листових матеріалів на деталі взуття виконуються вручну. Але розвиток обчислювальної техніки та методів обчислювальної математики дозво-

ляють виконувати ці етапи в автоматичному режимі. В роботах [1-3] представлені математичні моделі компактного розміщення опуклих плоских геометричних об'єктів. Але в більшості випадків деталі взуття є не опуклі плоскі геометричні об'єкти. В роботах [4-9] застосовуються генетичні алгоритми. Але ці алгоритми не завжди дають задовільний результат за обмежений час. Тому завданням цієї роботи є розробка методу автоматичного проектування раціональних схем розкрою листових матеріалів для плоских геометричних об'єктів з складною конфігурацією зовнішніх контурів.

Для розв'язання цієї задачі необхідно математична постановка задачі, в якій сформулювати технологічні вимоги та обмеження.

ПОСТАНОВКА ЗАДАЧІ

Дано прямокутні області Ω необмеженої кількості з габаритами розмірами D_{mat} та H_{mat} . Необхідно щільно



розмістити N_j плоских геометричних об'єктів S_j , де $j=1,2..q$, таким чином, щоб сумарна площа використаних прямокутних областей Ω була б мінімальною та задовольнялась потреба у викроєних плоских геометричних об'єктів S_j , де $j=1,2..q$. При цьому задовольнялись наступні обмеження:

– кількість різних плоских геометричних об'єктів у схемі розкрою для одної прямокутної області Ω не повинна перевищувати шести;

– плоскі геометричні об'єкти у схемі розкрою повинні не перетинатись та розміщатись на відстані один від одного не менше сталої величини Δ ;

– плоскі геометричні об'єкти у схемі розкрою повинні не виходити за межі прямокутної області Ω на величину Δ ;

– вимоги до орієнтації плоских геометричних об'єктів відносно матеріалу (плоских геометричних об'єктів на матеріалі розміщуються в основному положенні та повернуті відносно основного положення на 180 градусів; не має обмежень на орієнтацію деталей на матеріалі.);

– кількість плоских геометричних об'єктів кожного виду, що розміщені в прямокутних областях Ω , повинні задовольняти потреби в цих об'єктах.

РОЗВ'ЯЗОК ЗАДАЧІ

В такій постановці задача автоматичного проектування не має розв'язків, так як ця задача має нескінченну кількість локальних екстремумів. Тому розглянемо більш просту модель задачі, яка забезпечить ефективний пошук раціональних схем розміщення плоских геометричних об'єктів із врахуванням комплектного виходу. Для цього розіб'ємо цю задачу на три задачі: генерування множини допустимих секцій; генерування множини допустимих схем розкрою із комбінації згенерованих секцій; вибір із множини допустимих схем розкрою тих, які забезпечать мінімальну

Нижче детально розглянуто теоретичні основи, на яких розроблено ММ поставленої задачі

Аналитичне описання матеріалу прямокутної форми. Матеріал прямокутної форми однозначно можна представити системою наступних нерівностей (рис. 1):

$$\Omega: \begin{cases} 0 \leq X \leq D_{l_{mat}} \\ 0 \leq Y \leq H_{mat} \end{cases}$$

Аналитичне представлення інформації про контур деталі S . Контурні деталей взуття мають складну конфігурацію і в більшості випадків їх не вдається описати аналітично, тому їх апроксимують більш простими кривими.

сумарну площу використаних прямокутних областей Ω для побудови схем розміщення плоских геометричних об'єктів S_j ($j=1,2..q$) та задовольняють потребу в цих об'єктах.

Секція Q_{j_m} для плоского геометричного об'єкту S_j – це прямокутна область довжиною $D_{l_{mat}}$ та висотою H_{j_m} ($0 < H_{j_m} \leq H_{mat}$), в якій системно розміщуються плоскі геометричні об'єкти S_j (рис. 1.).

Тепер можна дати математичні постановки цієї задачі. Для цього спочатку визначимо поняття подвійної решітки. Множину векторів виду [10-11]:

$$r_1 = na + mb \text{ та } r_2 = na + mb + q,$$

де $n, m = 0, \pm 1, \pm 2, \pm 3, \dots \pm t, \dots$,

де a, b - лінійно-незалежні вектори, назвемо подвійною решіткою з базисом a, b і вектором зсуву решітки q та позначимо через $W = W(a, b, q)$.

Задача «Секція». Для плоского геометричного об'єкту S_j , де $j=1,2..q$, знайти прямокутну подвійну решітку $W^{pj}: na^{pj} + mb^{pj} + kq^j$ (рис.1.а) з найщільнішою укладкою деталей S_j . На базі цієї решітки згенерувати всю допустиму множину секцій Q_{j_m} , $m=1,2..m_k$ із висотою $H_{j_m} \leq H_{mat}$ та довжиною $D_{l_{max}}$, щільність яких $P_{j_m} \geq P$, де P - наперед задана щільність.

В математичній моделі цієї задачі можна виділити наступні структурні компоненти.

- Аналітичне описання матеріалу прямокутної форми;
- Аналітичне описання геометрії плоских геометричних об'єктів S_j , де $j=1,2..q$, що розміщуються в прямокутній області Ω .
- Аналітичне представлення положення плоских геометричних об'єктів в прямокутній області.
- Математичне описання системи розміщення плоских геометричних об'єктів в прямокутній області.
- Отримання аналітичного виразу для функції цілі.

Найбільше розповсюдження одержала *кусково - лінійна* апроксимація, тобто апроксимація кривими першого порядку (прямими). При цій апроксимації контур деталі представляється багатокутником. Кількість вершин апроксимуючого багатокутника повинна забезпечувати потрібну точність апроксимації. Тоді деталь S можна представити координатами точок вершин апроксимуючого багатокутника, тобто масивом $\{X_i, Y_i\}$, $i=1..n$, де X_i, Y_i - координати i -ї вершини та n -кількість вершин апроксимуючого багатокутника.

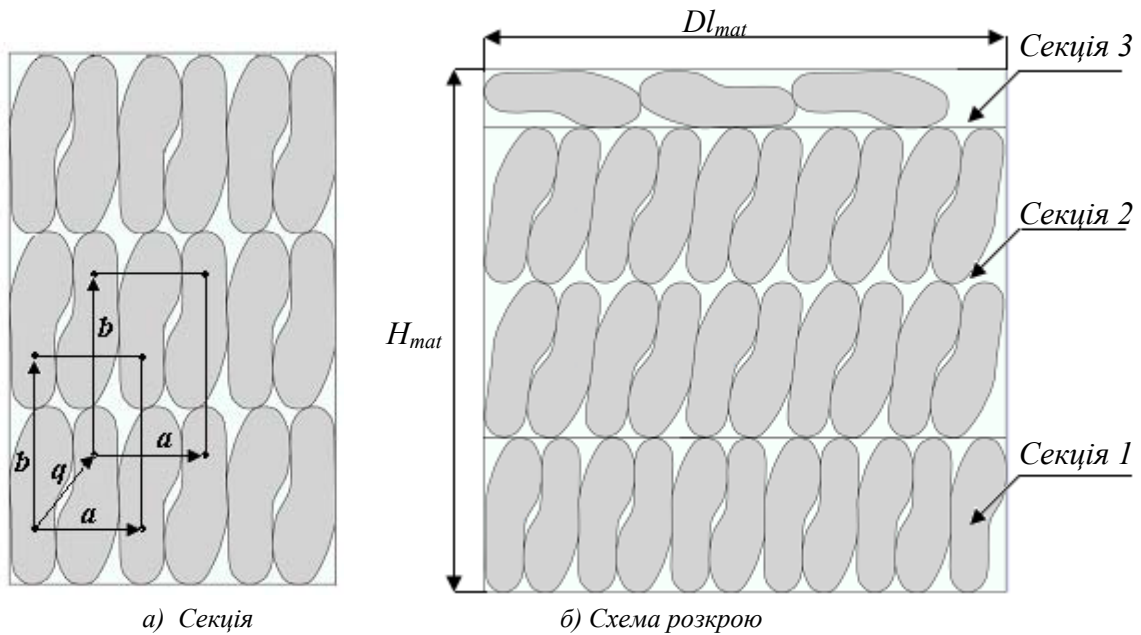


Рис. 1. Секція та схема розкрою

Таке представлення надає можливість дати аналітичне описання контурів розміщуваних фігур у вигляді систем рівнянь відрізків, з яких складаються ці контури. В параметричній формі запису ця система має вигляд [9]

$$\begin{cases} X(t_i) = X_i + (X_{i+1} - X_i)t_i \\ Y(t_i) = Y_i + (Y_{i+1} - Y_i)t_i \end{cases} \quad i = \overline{1, n} \quad (1)$$

де $(X_i; Y_i)$ точки на контурі деталі, вибрані при апроксимації, або вершини багатокутника, t_i - параметр, $t_i \in [0; 1)$.

Досвід свідчить, що для практичних задач достатньо забезпечити точність апроксимації контуру 0.5мм. Іншими словами, максимальне відхилення контуру деталі від ланки апроксимуючої лінії не повинно перевищувати 0.5мм. Виберемо на площині координатну систему XOY . В подальшому будемо визначати координати вершин апроксимуючого багатокутника для плоского геометричного об'єкту відносно фіксованої

точки в середині цього об'єкту. Цю точку назовемо полюсом плоского геометричного об'єкту.

Аналітичне представлення положення плоских геометричних об'єктів в прямокутній області. Для однозначного відображення положення плоского геометричного об'єкту S в прямокутній області Ω необхідно знати координати полюсу $P_{ij}(Xp_{ij}, Yp_{ij})$ плоского геометричного об'єкту S^q (точки, що знаходиться на плоскому геометричному об'єкті S^q та відносно якої задаються координати вершин апроксимуючого багатокутника $A_k^q(XS_k^q, YS_k^q)$, $k=1,2..n_q$ для цього плоского геометричного об'єкту) в системі координат XOY , що пов'язана із прямокутною областю Ω , та кут повороту θ_{ij} плоского геометричного об'єкту відносно його вихідного положення (рис. 2).

Тоді координати будь-якої вершини $A_k^q(XS_k^q, YS_k^q)$, $k=1,2..n_q$ апроксимуючого багатокутника для плоского геометричного об'єкту S^q в системі

координат XOY , що пов'язана з прямокутною областю Ω (рис. 2), визначатимуться наступним чином[7]:

$$\begin{cases} Xsr_k^q = Xs_k^q \cos \theta_{ij} - Ys_k^q \sin \theta_{ij} + Xp_{ij} \\ Ysr_k^q = Xs_k^q \sin \theta_{ij} + Ys_k^q \cos \theta_{ij} + Yp_{ij} \end{cases} \quad (2)$$

$i = 1, 2 \dots Kd, \quad j = 1, 2 \dots Kr$
 $q = 1, 2 \dots L$

де Kd – кількість плоских геометричних об'єктів в одному ряду секції, Kr – кількість рядів у секції, L – кількість різних за конфігурацією плоских геометричних об'єктів, для яких необхідно спроектувати схеми розкרוлю.

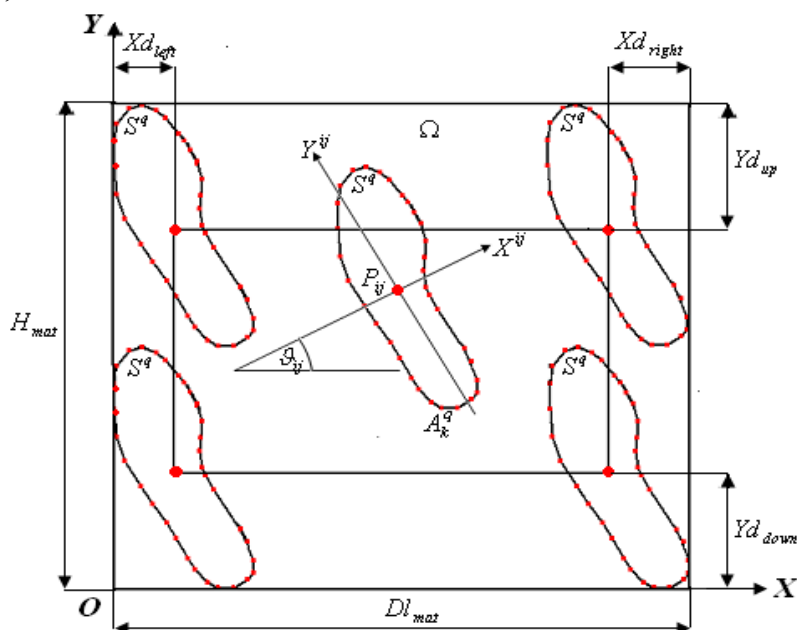


Рис. 2. Визначення положення плоского геометричного об'єкту в прямокутній області Ω

Тобто за допомогою виразів (2) можна однозначно аналітично описати зовнішній контур плоского геометричного об'єкту S^q із заданою точністю ϵ у щільній ґратці на площині.

Математичне описання системи розміщення плоских геометричних об'єктів в прямокутній області. За систему розміщення плоских геометричних об'єктів візьмемо прямокутну подвійну решітку $W: na + mb + kq$ $n, m = 0, \pm 1, \pm 2, \pm 3, \dots \pm t \dots, k = 0, 1$. Для однозначного визначення параметрів подвійної решітки W необхідно визначити вектори a, b, q .

Визначимо ці вектори через наступні параметри (рис. 3):

Sh_1 – ширина одного плоского геометричного об'єкту;

Sh_2 – ширина двох щільно суміщених плоских геометричних об'єктів;

Sh_3 – ширина трьох щільно суміщених плоских геометричних об'єктів;

H_1 – висота одного плоского геометричного об'єкту;
 H_2 – висота двох щільно суміщених плоских геометричних об'єктів;

Xp_{11}, Yp_{11} – координати полюсу P_{11} плоского геометричного об'єкту.

Тоді вектори a, b матимуть наступні координати:

$$a(\Delta_1, 0), b(0, \Delta_2).$$

Визначимо Δ_1 та Δ_2 .

Так як $P_{11}P_{13} = P_{13}P_{15} = P_{15}P_{17} = P_{21}P_{23} = P_{23}P_{25} = P_{25}P_{27} = P_{31}P_{33} = P_{33}P_{35} = P_{35}P_{37}$, то очевидно, що $\Delta_1 = Sh_3 - AB - GH = Sh_3 - (AB + GH)$. Так як $GH = BC$, а $AB + BC = Sh_1$, то $\Delta_1 = Sh_3 - Sh_1$. Очевидно, що $\Delta_2 = H_2 - d_1 - d_2 = H_2 - (d_1 + d_2)$ (рис. 3.6). Так як $d_1 + d_2 = H_1$ (рис. 3), то $\Delta_2 = H_2 - H_1$. Тоді $a\{\Delta_1, 0\} = \{Sh_3 - Sh_1, 0\}$, $b\{0, \Delta_2\} = \{0, H_2 - H_1\}$.

З рис. 3.6 очевидно, що $q\{Xp_{12} - Xp_{11}, Yp_{12} - Yp_{11}\}$. Так як плоский геометричний об'єкт з полюсом P_{12} повернутий відносно плоского геометричного об'єкту з полюсом P_{11} (рис. 2), то $Yp_{12} = H_1 - Yp_{11}$. Визначимо Xp_{12} . З рис. 3 видно, що $Xp_{12} = Sh_2 - EF$. Так як $EF = AB = Xp_{11}$, то $Xp_{12} = Sh_2 - Xp_{11}$.

$z = Sh_2 - Xp_{11}$. Тоді вектор q визначається наступним чином:

$$q\{Xp_{12}-Xp_{11}, Yp_{12}-Yp_{11}\} = \{Sh_2-2Xp_{11}, H_1-Yp_{11}\}.$$

Визначивши вектори $a\{a_x, 0\}$, $b\{0, b_y\}$, $q\{q_x, q_y\}$ та Xp_{11} , Yp_{11} можна виразити координати полюса будь-якого із розміщених в секції плоского геометрич-

ного об'єкту через ці величини наступним чином (рис. 3.6):

$$Xp_{ij} = Xp_{11} + [i/2] \cdot a_x + (j+1-2 \cdot [(j+1)/2]) \cdot q_x \quad (3)$$

$$Yp_{ij} = Yp_{11} + (i+1) \cdot b_y + (j+1-2 \cdot [(j+1)/2]) \cdot q_y \quad (4)$$

$$i=1,2..Kd, j=1,2..Kr$$

де $Kd = [(Dl_{mat} - Dl_1)/a_x]$, $Kr_{max} = [(H_{mat} - H_1)/b_y]$ та $Kr = 1,2..Kr_{max}$, Kr_{max} - максимальна можлива кількість рядів у секціях для заданого плоского геометричного об'єкту.

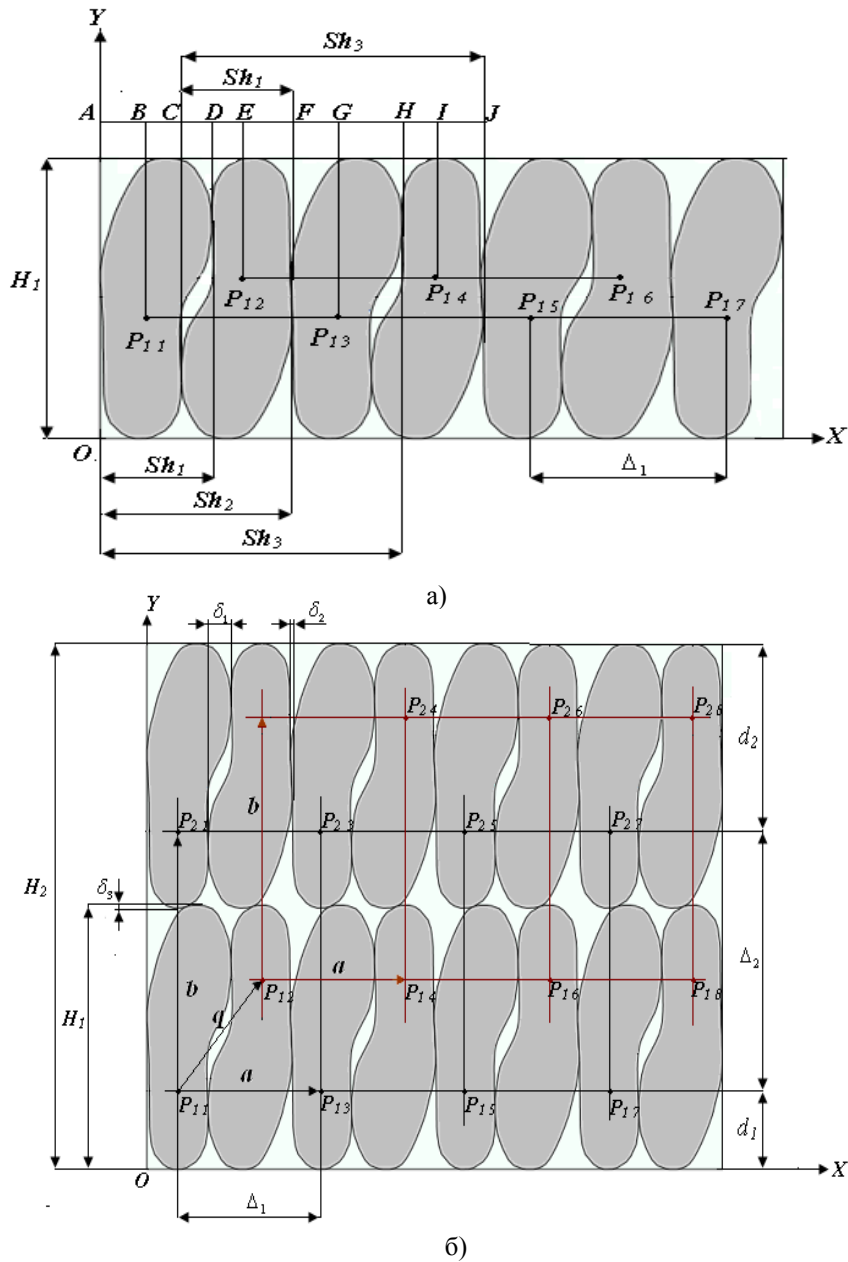


Рис. 3. Параметри, що визначають схему розкря

Для визначення Sh_2 введемо поняття лівої L_{left} та правої L_{right} границь зовнішнього контуру плоского геометричного об'єкту S . Під лівою(правою) границею $L_{left}(L_{right})$ зовнішнього контуру плоского геометричного об'єкту S будемо розуміти ділянку зовнішнього контуру цього плоского геометричного об'єкту, яка знаходиться ліворуч(праворуч) від прямої, що з'єднує точки з максимальним та мінімальним значенням координати Y .

Так як зовнішній контур плоского геометричного об'єкту ми апроксимуємо у вигляді багатокутника, то ліва та права границі зовнішнього контуру плоского геометричного об'єкту будуть представлятись у вигляді ламаних ліній, тобто $L_{left} \{X_{left_i}, Y_{left_i}\} i = 1, 2, \dots, n_{left}$ та $L_{right} \{X_{right_j}, Y_{right_j}\} j = 1, 2, \dots, n_{right}$ (рис. 4.a).

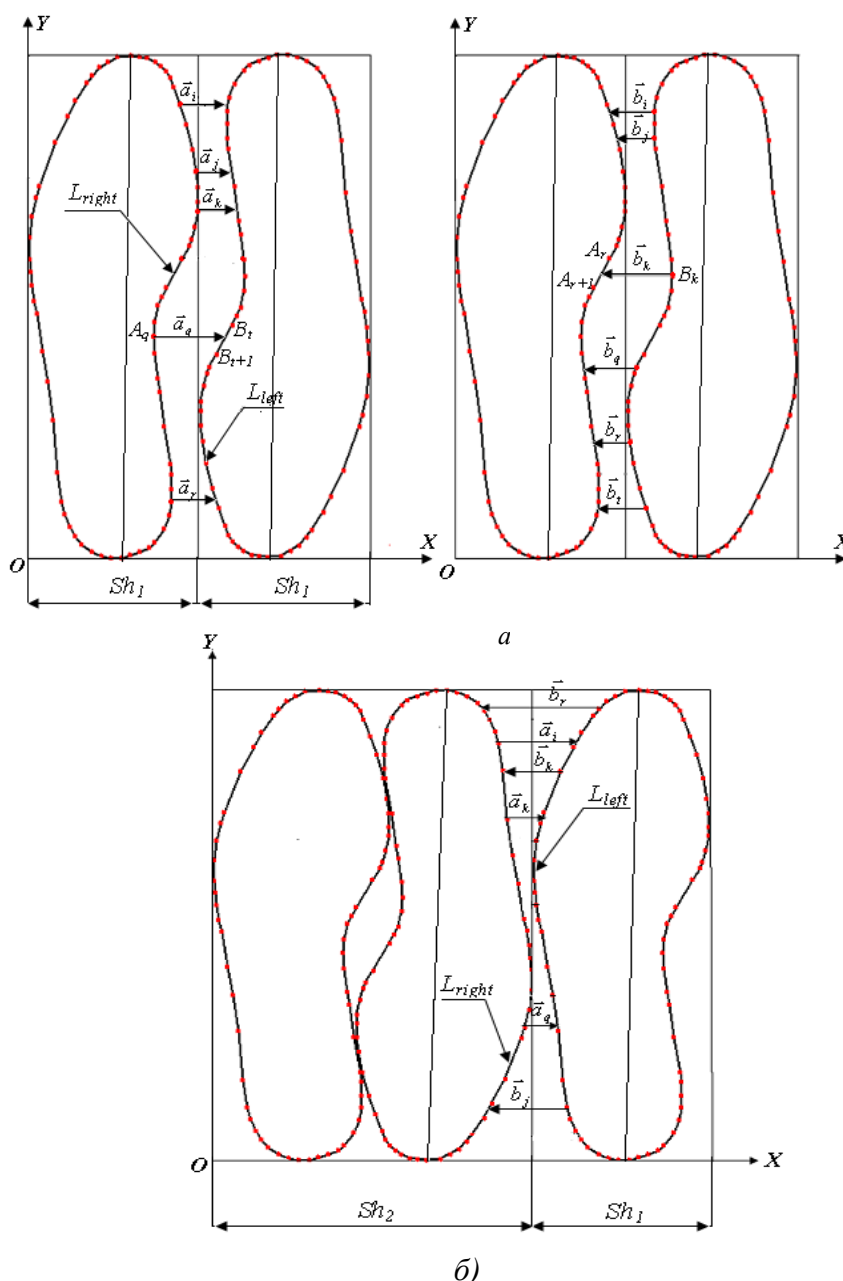


Рис. 4. Знаходження параметрів щільного суцїщення Sh_2 та Sh_3

Для знаходження Sh_2 (рис. 5.а) (Sh_3 (рис.5.6)):

- щільно сумістимо в ряду прямокутники, які описані навколо першого (перших двох) та другого(третього) плоских геометричних об'єктів;

- з кожної вершини A_i правої границі зовнішнього контуру першого (другого) плоского геометричного об'єкта проведемо вектор \vec{a}_i паралельний вісі OX таким чином, що кінець його дотикається до лівої границі другого(третього) плоского геометричного об'єкта;

- знайдемо $\delta_a = \min_{i=1,2..n_{right}} \{|\vec{a}_i|\}$;

- серед векторів \vec{a}_i знайдемо ті вектори, які задовольняють умові $|\vec{a}_i| = \delta_a$. Для них визначимо вектор \vec{a}_k з максимальним значенням координати Y , тобто $Y_{\vec{a}_k} = Y \max_{left}$. Визначимо Q_a як $Q_a = H_1 - Y \max_{left}$;

- з кожної вершини B_j лівої границі зовнішнього контуру другого (третього) плоского геометричного об'єкта проведемо вектор \vec{b}_j паралельний вісі OX таким чином, що кінець його дотикається до правої границі першого(другого) плоского геометричного об'єкта;

- знайдемо $\delta_b = \min_{j=1,2..n_{left}} \{|\vec{b}_j|\}$;

- серед векторів \vec{b}_j знайдемо ті вектори, які задовольняють умові $|\vec{b}_j| = \delta_b$. Для них визначимо

вектор \vec{b}_k з максимальним значенням координати Y , тобто $Y_{\vec{b}_k} = Y \max_{right1}$ ($Y_{\vec{b}_k} = Y \max_{right2}$). Ви-

значимо Q_{1b} (Q_{2b}) як $Q_{1b} = H_1 - Y \max_{right}$ ($Q_{2b} = H_1 - Y \max_{right2}$);

- знайдемо $\delta_1 = \min\{\delta_a, \delta_b\}$

($\delta_2 = \min\{\delta_a, \delta_b\}$)(рис.4);

- знайдемо $Q_1 = \begin{cases} Q_{1a}, & \text{якщо } \delta_a = \delta_1 \\ Q_{1b}, & \text{якщо } \delta_b = \delta_1 \end{cases}$

($Q_2 = \begin{cases} Q_{2a}, & \text{якщо } \delta_a = \delta_2 \\ Q_{2b}, & \text{якщо } \delta_b = \delta_2 \end{cases}$);

- знайдемо $Q = \min\{Q_1, Q_2\}$;

- визначимо Sh_2 (Sh_3) як $Sh_2 = 2Sh_1 - \delta_1$

• ($Sh_3 = Sh_2 + Sh_1 - \delta_2$).

Введемо поняття верхньої границі нижнього ряду та нижньої границі верхнього ряду у секції. Для цього щільно сумістимо два прямокутники, що описані навколо двох рядів щільно суміщених плоских геометричних об'єктів (рис. 5). Під верхньою (нижньою) границею нижнього(верхнього) ряду у секції будемо підрозумувати ломану лінію, яка складається із ділянок зовнішніх контурів щільно суміщених у ряду плоских геометричних об'єктів та координата Y яких задовольняє наступній умові:

$$H_1 - Q \leq Y \leq H_1 \quad (H_1 \leq Y \leq H_1 + Q).$$

Нехай верхня границя нижнього ряду секції описується функцією $F_1(x,y)=0$, а нижня границя верхнього ряду секції описується функцією $F_2(x,y)=0$. Очевидно, що $F_1(x,y)=0$ та $F_2(x,y)=0$ є періодичними функціями з періодом, який дорівнює довжині вектору решітки \mathbf{a} , тобто $T=|\mathbf{a}|$ (рис. 5). Тоді для щільного суміщення рядів у секції достатньо щільно сумістити ділянки верхньої границі нижнього ряду та нижньої границі верхнього ряду секції на наступному інтервалі $X_0 \leq X \leq X_0 + T$. значення X : Для щільного суміщення верхнього ряду з нижнім рядом в секції необхідно визначити величини зсуву верхнього ряду δ_3 , Тобто необхідно для виділених ділянок верхньої границі нижнього ряду та нижньої границі верхнього ряду секції на інтервалі $X_0 \leq X \leq X_0 + T$:

- щільно сумістити прямокутники, що описані навколо верхнього та нижнього рядів у секції(рис. 6.а);

- виділити ділянки верхньої границі нижнього ряду та нижньої границі верхнього ряду секції на інтервалі $X_0 \leq X \leq X_0 + T$ (рис. 5.6);

- з кожної вершини A_i верхньої границі нижнього ряду проведемо вектор \vec{a}_i паралельно вісі OY таким чином, що кінець його дотикається до нижньої границі верхнього ряду секції(рис. 6.б);

- знайти $\delta_a = \min_{i=1,2..n_{up}} \{|\vec{a}_i|\}$;

- з кожної вершини B_j верхньої границі нижнього ряду проведемо вектор \vec{b}_j паралельно вісі OY таким чином, що кінець його дотикається до нижньої границі верхнього ряду секції(рис. 6.б);

- знайти $\delta_b = \min_{j=1,2,\dots,n_{down}} \{\bar{b}_j\}$;
- знайти $\delta_3 = \min\{\delta_a, \delta_b\}$.

Тоді $H_2 = 2H_1 - \delta_3$ (рис. 5).

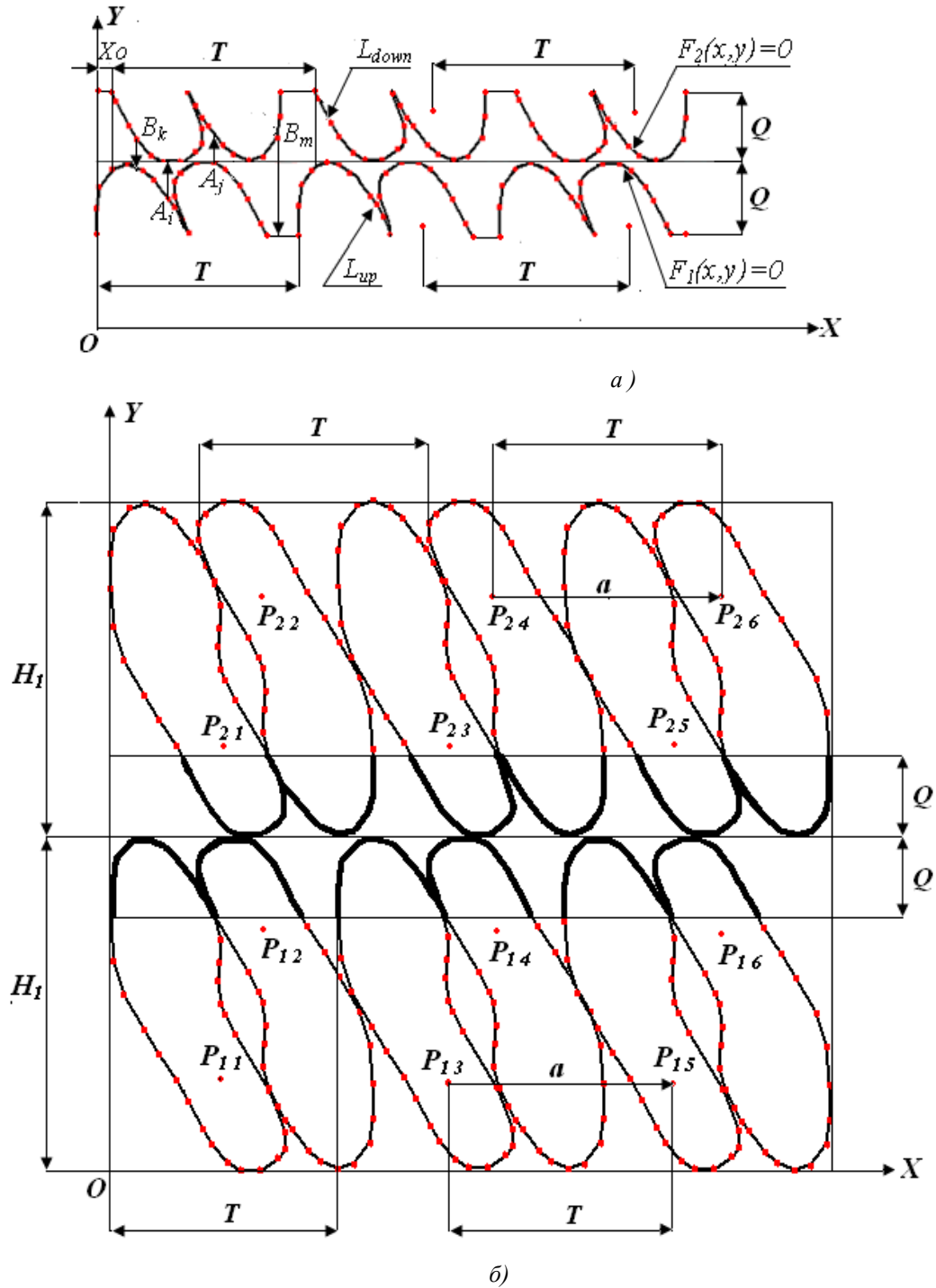


Рис. 5. Щільне суміщення рядів

Отримання аналітичного виразу для функції цілі. За функцію цілі візьмемо щільність P_{jm} секції Q_{jm} – це відношення чистої площі плоских геометричних об'єктів S_j , які розміщені у секції до площі секції, тобто

$$P_{jm} = N_{jm} |S_j| / (Dl_{mat} \cdot H_{jm}),$$

де N_{jm} – кількість плоских геометричних об'єктів S_j , що розміщені у секції Q_{jm} ; $|S_j|$ – площа плоского геометричного об'єкта S_j .

Визначимо N_{jm}, H_{jm} . Очевидно, що:

$$N_{jm} = Kd \cdot Kr; \quad H_{jm} = (Kr - 1) \cdot b_j + H_1.$$

Алгоритм генерування секцій. Для кожного із плоских геометричних об'єктів $S_i, i=1,2..K$ дет генеруємо множину допустимих секцій за наступним алгоритмом:

- визначаємо параметри $Xp_{1i}, Yp_{1i}, Sh_1, Sh_2, Sh_3, H_1, H_2$ для плоского геометричного об'єкту S_i ;
- визначаємо вектори решітки a, b, q для плоского геометричного об'єкту S_i ;
- визначаємо параметри $Xp_{1i}^A, Yp_{1i}^A, Sh_{1i}^A, Sh_{2i}^A, Sh_{3i}^A, H_{1i}^A, H_{2i}^A$ для плоского геометричного об'єкту S_i , які за-

безпечать стали відстань Δ між плоскими геометричними об'єктами S_i при їх щільному розміщенні;

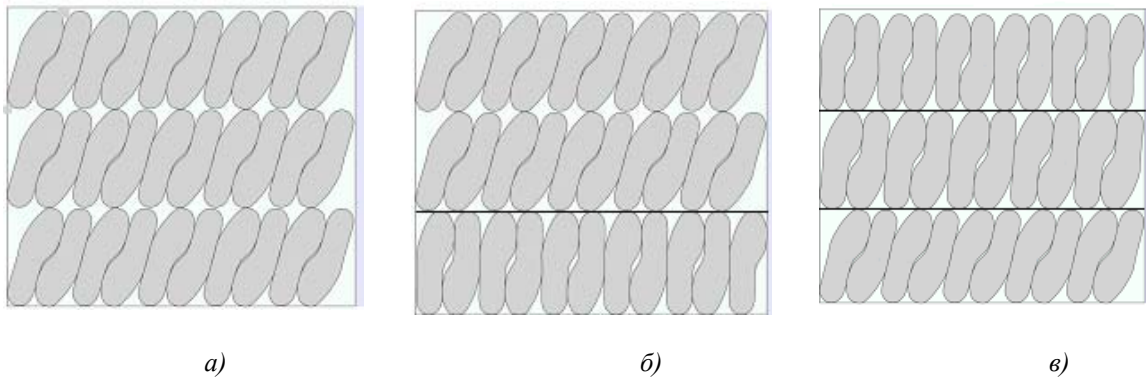
- визначаємо вектори решітки a^A, b^A, q^A для плоского геометричного об'єкту S_i , які забезпечать стали відстань Δ між плоскими геометричними об'єктами S_i при їх щільному розміщенні;

- визначаємо Kd – максимальну можливу кількість плоских геометричних об'єктів S_i в ряду секції;

- визначаємо максимальну можливу кількість рядів Kr_{max} в секції для плоского геометричного об'єкту S_i ;

- генеруємо секції для плоского геометричного об'єкту $S_j, j=1,2..k$ з одного ряду, двох рядів і так до Kr_{max} рядів.

Генерування множини допустимих схем розкрою. Схеми розкрою генеруються із комбінації однієї, двох, трьох секцій для різних плоских геометричних об'єктів із множини згенерованих секцій (рис. 6). Схему розкрою будемо вважати допустимою, якщо для неї виконується наступна умова: $P_{sx} \geq P_0$, де P_{sx} – щільність розміщення плоских геометричних об'єктів в схемі розкрою, P_0 – наперед задана допустима щільність розміщення плоских геометричних об'єктів в схемі розкрою.



а) з однієї секції; б) з двох секцій; в) з трьох секцій

Рис. 6. Схема розкрою, що складається:

Щільність розміщення плоских геометричних об'єктів в схемі розкрою визначається наступним чином:

- для схеми розкрою, що складається з однієї секції (рис. 6.а)

$$P_{sx} = P_{jm} \cdot H_{jm} / H_{mat};$$

- для схеми розкрою, що складається з двох секцій (рис. 6.б)

$$P_{sx} = (P_{jm} \cdot H_{jm} + P_{it} \cdot H_{it}) / H_{mat};$$

- для схеми розкрою, що складається з трьох секцій (рис. 6.в)

$$P_{sx} = (P_{jm} \cdot H_{jm} + P_{it} \cdot H_{it} + P_{nr} \cdot H_{nr}) / H_{mat},$$

де P_{jm}, P_{it}, P_{nr} – відповідно щільність розміщення плоских геометричних об'єктів в j, i, n – их секціях відповідно для плоских геометричних об'єктів S_m, S_t, S_r .

Алгоритм генерування схем розкрою. Алгоритм генерування схем розкрою складається із трьох основних частин:

- генерування схем розкрою для одного виду плоского геометричного об'єкту $S_j, j=1,2..k$. Це будуть секції для плоских геометричних об'єктів $S_j, j=1,2..k$ з максимально допустимою кількістю рядів;
- генерування схем розкрою із двох секцій з різними видами плоских геометричних об'єктів;
- генерування схем розкрою із трьох секцій з різними видами плоских геометричних об'єктів.

Введемо поняття план розкрою. Під планом розкрою будемо розуміти набір схем розкрою із згенерованих схем розкрою, які забезпечують комплектний вихід в плоских геометричних об'єктів та мінімізують відходи при розкрої матеріалу.

Підзадача «Комплект». Із множини допустимих схем розкрою \hat{S}_{kr} вибрати таку підмножину схем \hat{S}_{kr}^* , комбінація яких утворить план розкрою, в якому буде врахований комплектний вихід плоских геометричних об'єктів та використання матеріалу буде максимальним.

Математична постановка підзадачі «Комплект».

Аналітичне представлення вимог комплектного виходу. Аналітичне представлення вимог комплектного виходу має вигляд

$$N_j - \chi \leq \sum_{i=1}^r A_{ij} \cdot x_i \leq N_j, \quad j=1,2..q,$$

де q – кількість видів плоских геометричних об'єктів S_j , для яких будується план розкрою, N_j – потреба в плоскому геометричному об'єкті S_j , A_{ij} – кількість плоских геометричних об'єктів S_j в i -ій схемі розкрою, X_i – кількість i -их схем розкрою в плані розкрою, χ – допустиме відхилення від комплектного виходу.

Множина допустимих розв'язків задачі. Множиною допустимих розв'язків задачі „Комплект” будуть підмножини секцій із множини згенерованих секцій \hat{S}_i , для яких

$$N_j - \chi \leq \sum_{i=1}^r A_{ij} \cdot x_i \leq N_j, \quad j=1,2..q.$$

Функція цілі. Функція цілі для задачі «Комплект» має наступний аналітичний вигляд:

$$L=f(X_i, S_j, N_j, D_{mat}, H_{mat}) = \sum_{i=1}^r X_i \cdot D_{mat} \cdot H_{mat} - \sum_{j=1}^q \sum_{i=1}^r X_{ii} \cdot |S_j| \cdot A_{ij} \rightarrow \min,$$

де $S_{h_{mat}}$ та H_{mat} – габаритні розміри прямокутної області; $|S_j|$ – площа плоского геометричного об'єкту $S_j, j=1,2..q$ та $i=1,2..r$.

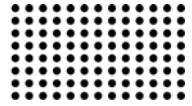
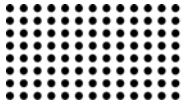
Це задача цілочислового програмування великого розміру. Так як кількість допустимих схем розкрою досягає понад 10 тисяч, то ефективних точних методів розв'язання цієї задачі на цей час не існує. Для розв'язку цієї задачі спеціально розроблений евристичний метод. Цей метод закладається в наступному:

1. Відсортуюмо згенеровані схеми розкрою за спаданням щільності розміщення деталей у них.
2. Перебираючи схеми розкрою з допустимою ефективністю розраховуємо для кожної розкрійної схеми її кількість в плані розкрою.
3. Після додавання кожної розкрійної схеми до плану розкрою також розраховуємо і залишок деталей.
4. Процес перебору розкрійної схеми продовжуємо до того часу, поки не буде задоволена потреба у деталях або не будуть перебрані всі допустимі схеми розкрою.
5. Визначаємо щільність розміщення плоских геометричних об'єктів P_{plan} у отриманому плані розкрою та запам'ятовуємо схеми розкрою, що ввійшли у план розкрою та кількість кожної із цих схем.
6. Серед відсортованих секцій міняємо першу і другу схеми розкрою місцями і весь попередній процес (пункти 2-5) побудови плану розкрою повторюємо і т.д..

ВИСНОВКИ

Запропонований метод автоматичного проектування раціональних схем розкрою листових матеріалів для плоских геометричних об'єктів довільної конфігурації зовнішнього контуру дозволив розробити програмне забезпечення для автоматичного проектування схем розкрою.

Цей програмний продукт можна в подальшому після доопрацювання використати для автоматичного



проектування раціональних схем розкрою рулонних матеріалів

Впровадження програмного продукту у виробництво дозволить підвищити продуктивність праці техноло-

га при підготовці схем розкрою та ефективність використання матеріалів та зробить можливим впровадження в розкрійне виробництво автоматизовані розкрійні комплекси.

ЛІТЕРАТУРА

1. Guo P.N. Floor-planning using a tree representation/ P.N. Guo, T. Takahashi, C.K. Cheng, T. Yoshimura // IEEE Trans. on Computer Aided Design of Integrated Circuits and Systems. – 2001. – 281 p.
2. Sakanushi K. The quarter-state-sequence floorplan representation / K. Sakanushi, Y/ Kajitani, D.P. Mehta// IEEE Trans. on Computer Aided Design of Integrated Circuits and Systems. – 2003. – 376 p.
3. 3.Orlov A.N. Kureychik V.V. Mekhanizm kodirovaniya-dekodirovaniya pri reshenii zadachi pryamougol'nogo raskroya-upakovki materiala // Sbornik nauchnykh statey Vserossiyskoy molo-dezhnoy shkoly seminaru «Aktual'nyye problemy informatsionnykh tekhnologiy, elektroniki i radiotekhniki - 2015». – Taganrog: Izd-vo NOTS ZIS KT YUFU, 2015. – S. 420-428.
4. Kolisko O.Z. Modifikatsiya genetichnogo algoritmu dlya generatsii' sektsiy rozkryinnykh skhem / O. Z. Kolisko // Visnik KNUTD. – 2009. – №1. – S. 14-17.
5. Gladkov L.A., Kureychik V.V., Kureychik V.M., Sorokoletov P.V. Bioinspirirovannyye metody v optimizatsii. – M.: Fizmatlit, 2009. – 384 c.
6. Gladkov L.A. Kureychik V.V., Kureychik V.M. Geneticheskiye algoritmy. – M.: Fizmatlit, 2010. – 368 c.
7. Sherwani N.A. Algorithms for VLSI Physical Design Automation. Third Edition, Kluwer Academic Publisher, USA, 2013., 338 p.
8. Kureychik V.V., Kureychik V.I. Bioinspirirovannyi poisk pri proyektirovanii i upravlenii // Izvestiya YUFU. Tekhnicheskiye nauki. – 2012. – № 11 (136). – S. 178-183.
9. Zhang D.F. An improved heuristic recursive strategy based on genetic algorithm for the strip rectangular packing problem/ D.F. Zhang, C.D.Chen, Y.I. Lin // Acta Automatica Sinica. – 2007. – №33(9). – P. 911-916
10. Chuprynka V.I. Method of construction of lattice tilings of two plane geometric object with different configurations of outer contours/ V.I. Chuprynka, G.Y. Zelinsky, N.V. Chuprynka // The scientific heritage – 2017, – №8, Vol. 1, P. 100-109
11. Chuprinka V.Í. Avtomatizovane proyektuvannya ratsional'nykh skhem rozkroyu rulonnikh materialiv na detalí vzuttya / V.Í. Chuprinka, V.S. Murzhenko, A.V. Pinchuk, // Visnik Skhidnoukraïns'kogo natsional'nogo univ'ersitetu ímení Volodimira Dalya. – 2013. – №15 (204). Chastina Í. – S. 243–249
12. Il'in V.A. Analiticheskaya geometriya/ V.A. Il'in, E.G. Poznyak. – M.:Izdatel'stvo "Nauka", Glavnaya redaktsiya fiziko-matematicheskoy literatury, 1975, – 243 s.

*Рецензент: д.т.н., проф. Рудакова Г.В.
Херсонський національний технічний університет*



ПРОГНОЗУВАННЯ ДОХІДНОСТІ БАНКІВСЬКИХ ПРОДУКТІВ З ВИКОРИСТАННЯМ СКОРИНГОВОГО ПІДХОДУ

УДК 519.226; 336.717.06

БІДЮК Петро Іванович

д.т.н., професор, професор Інституту прикладного системного аналізу НТУУ «КПІ», м. Київ.

Наукові інтереси: аналіз нестационарних часових рядів, імовірнісне прогнозування, ідентифікація, математичне моделювання і прогнозування фінансових ризиків.

e-mail: pbidyuke@gmail.com

ДЕМКІВСЬКИЙ Євген Олександрович

к.т.н., доцент, доцент кафедри інформаційних систем факультету комп'ютерних наук та кібернетики Київського національного університету імені Тараса Шевченка

e-mail: demkivsky@gmail.com

Наукові інтереси: моделювання та прогнозування динамічних систем, побудова оптимальних систем підтримки прийняття рішень на основі математичних моделей та сучасних методів управління.

ПУДЛО Ілля Віталійович

магістр Інституту прикладного системного аналізу НТУУ «КПІ», м. Київ.

Наукові інтереси: ідентифікація, математичне моделювання і прогнозування фінансових ризиків.

e-mail: IlliaPudlo@gmail.com

ДЕМКІВСЬКА Тетяна Іванівна

к.т.н., доцент, доцент кафедри Інформаційних технологій проектування Київського національного університету технологій та дизайну.

Наукові інтереси: моделювання та прогнозування процесів на основі комп'ютерного моделювання часових рядів, побудова оптимальних систем підтримки прийняття рішень на основі математичних моделей та сучасних методів управління

e-mail: demkivskiy@gmail.com

Вступ

В ефективному функціонуванні банківських установ в умовах ринкової економіки та в умовах кризових явищ важливе місце повинна зайняти стратегія управління ризиком. Стратегія управління ризиком повинна поєднувати внутрішньобанківську діяльність із механізмами з боку Національного банку України та інших державних структур [1].

Кредитування є найбільш прибутковим і одночасно ризикованим видом банківської діяльності. Висока частка проблемного кредитного портфелю може спричинити банкрутство банку, а через його положення в економіці, до цілого ряду банкрутств, пов'язаних з ним суб'єктів господарювання. Тому управління кредитним

ризиком є необхідною частиною стратегії і тактики розвитку будь-якого банку.

Із збільшенням об'ємів кредитування актуалізуються і задачі управління кредитним ризиком банку. В зв'язку з цим розробка методів оцінки і механізму регулювання кредитних ризиків забезпечує зміцнення фінансового положення банку.

Кредитний ризик є одним із найбільш вивчених видів ризику, для оцінки характеристик якого розроблено багато методів (серед існуючих методів оцінки кредитного ризику є скоринг, методика Монте-Карло, методика НБУ, методика Базельського комітету тощо).

Базовим та одним із найбільш поширеним елементом стратегії управління ризиком є скоринг.

Скоринг є математичною або статистичною моделлю, за допомогою якої на основі кредитної історії «минулих» клієнтів банк намагається визначити, наскільки велика вірогідність, що конкретний потенційний позичальник поверне кредит в строк [2, 3, 4].

Його суть полягає в тому, що кожному параметру, що характеризує позичальника, надається реальна оцінка в балах. Таким чином, у спрощеному вигляді, скорингову модель можна представити як зважену суму визначених характеристик позичальника: вік, сімейний стан, місце роботи, дохід та багато інших факторів. Така методика є знеособленою і може застосовуватися як для фізичних, так і для юридичних осіб.

Скоринг є класифікаційною задачею, де виходячи з наявної інформації необхідно одержати функцію, що найточніше розділятиме вибірку клієнтів на «поганих» і «хороших». Скорингові моделі є первинним індикатором кредитоспроможності потенційного позичальника. На їх основі експерт ухвалює остаточне рішення про видачу кредиту [5, 6, 7].

Можна виділити два основні типи кредитного скорингу, незалежно від того яка саме цільова змінна прогнозується – це:

– аплікаційний скоринг – оцінка клієнта без використання інформації про його попередню поведінку в компанії (базуючись на соціо-демографічних характеристиках клієнта). Найчастіше модель будується на прогнозуванні ймовірності дефолту (недотримання зобов'язань) клієнта в перші n місяці життя кредиту. Основне завдання відділити на портфельному рівні групи клієнтів з високою ймовірністю дефолту та запобігти видачі кредиту таким клієнтам.

– поведінковий скоринг – оцінка клієнта з використанням інформації про його попередню поведінку в компанії. Найчастіше використовуються для застосування тих чи інших дій до клієнта вже в процесі обслуговування ним боргу. Діапазон прогнозування набагато ширший ніж в аплікаційному скоринзі: ймовірність дефолту, прогнозування повернення після дефолту, ймовірність відтоку клієнта з банку (закриття договору), ймовірність переплати по кредиту, ймовірність додзвону тощо.

Статистичне прогнозування в кредитному скорингу базується не на багаторазовому повторенні експерименту для одного клієнта (що технічно є неможливим), а в групуванні клієнтів в подібні групи по демографічних, бізнесових, поведінкових характеристиках та в знаходженні взаємозв'язку між характеристиками, що визначають ці групи, та цільовою змінною.

Прогнозоване значення цільової змінної розбивається для зручності на сегменти, – рейтингові класи, і стратегії всіх подальші дій з клієнтом та його кредитами приймаються на рівні рейтингу [8, 9].

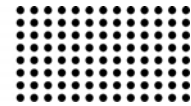
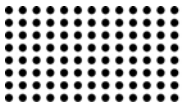
Основний статистичний метод, який в останнім часом асоціюється в банках з кредитним скорингом – це логістична регресія [10, 11]. Поряд з іншими більш витонченими та складнішими моделями вона має дві суттєві переваги:

- простота інтерпретації;
- стабільність.

Вагові коефіцієнти отримані в результаті розрахунків легко переводяться в скорингові бали та карти, які в свою чергу інтерпретуються просто: «краща група клієнтів – більший бал». Ризик-менеджмент більшості європейських банків не сприймає моделі на кшталт «чорний ящик», в яких логіка виведення рішення по конкретному запису(клієнту) настільки складна, що не може бути доволі просто, а то і взагалі відображена аналітично, тобто формулами на листку паперу. Прикладами таких моделей є моделі нейронних мереж, SVM, метод k -найближчих сусідів, Random forests тощо. Поряд з іншими статистичними методами, які можуть давати несуттєвий приріст до предикативної сили моделі, логістична регресія є однією з найбільш стабільних, тобто такою що зберігає свою силу протягом довгого періоду часу і не потребує частішої перебудови. Крім того наявність класичної скоркарти дозволяє якісно «моніторити» модель та давати відповіді на питання, які характеристики моделі (фактично її складові частинки) дають збої в предикативності та зсуви в популяції.

ПОСТАНОВКА ЗАДАЧІ

Починаючи з 2006 року банки в Україні почали приділяти значно більшу увагу кредитним ризикам. З того часу були розвинуті знання у цій сфері та побудовано безліч моделей для мінімізації та контролю кредитного ризику. Попри це існує ще один ризик, який може нане-



сти банку не менші збитки від кредитного. Він пов'язаний із клієнтами які користуються продуктами із грейсовим (пільговим) періодом або достроково закривають кредит.

Даний вид ризику полягає у тому, що банк, залучивши ресурси під певний відсоток (за певну ціну), не отримує прибутку від подальшої видачі цих ресурсів клієнту і цим самим несе збиток. Запропонована модель дозволить провести сегментацію клієнтів і запобігти втратам, що можуть призвести до значного зниження кінцевого фінансового результату. Таким чином, залежно від кредитної політики, яку проводить банківська установа, можна здійснювати більш якісний контроль дохідності від даного типу продукту.

Задача даної роботи полягає у розробці моделі для револьверних карток із грейсовим (пільговим) періодом, яка дозволить контролювати дохідність даного продукту. Револьверний кредит (англ. revolving credit) – автоматично поновлюваний (від лат. revolve –

обертатись) кредит, який широко використовується у світовій практиці на ринку позичкового капіталу [12].

Грейсовий період (англ. grace period) – пільговий період сплати відсотків за кредитом. За використання пільгового періоду відсотки за користування грошовими засобами не нараховуються взагалі або їх величина є значно нижчою за базову ставку.

ПОБУДОВА МОДЕЛІ

Аплікаційна карта розрахована на прогноз дефолту по кредиту протягом певного фіксованого часу після його видачі (найчастіше 12 місяців). Дана модель оцінює популяцію, що заходить в банк (як видані так і відмовлені кредити) та визначає ймовірність дефолту кожного із таких кредитів, на основі чого зокрема і приймається рішення про видачу/невидачу кредиту. Відповідно і вибірка, історія дефолтів по якій вже відомо, формується в такий же спосіб.

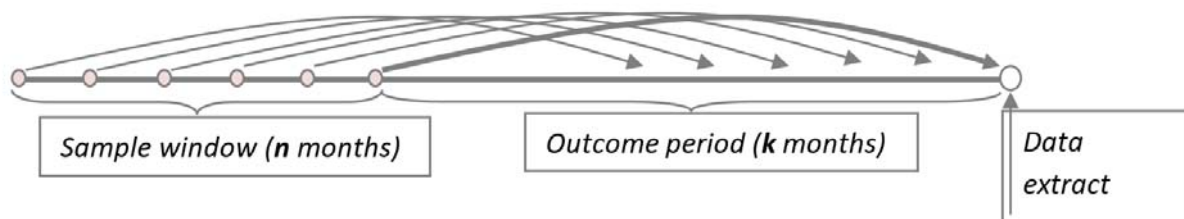


Рис.1 Формування вибірки

Беруться всі кредити видані за певний період в минулому, найближчі до поточної дати, так, щоб в кредитів був час стати «добрими» чи «поганими». Цей фіксований час (k) називається Outcome періодом. Сама вибірка виданих кредитів (за n місяців) називається Sample Window.

Найчастіше розмір Sample Window сягає від 6 до 12 місяців (в нашому випадку – 6 місяців). Вибір розміру такого вікна повинен враховувати сезонності при видачах кредитів та досягнення мінімального обсягу вибірки для побудови.

У випадку якщо найбільш остання історія не є найкращим представником для найближчого майбутнього (наприклад історія кризового періоду для побудови моделі в антикризовий період) можна брати «старіше» Sample window. Основним мотивом в такому випадку є

максимальне наближення популяції, на якій будується модель до популяції, на якій вона буде застосовуватись. Таким чином для побудови було обрано період 01.07.2012 – 31.12.2012. Саме цей період можна прийняти як до кризовий в банківській системі України.

Outcome період k найчастіше вибирається як 6-12 місяців (в нашому випадку 6 місяців).

Як для кредитів виданих k місяців тому так і для кредитів виданих $k+1$, $k+2$, ... місяці тому дається однаково час k для того щоб кредит попав у дефолт. Іншими словами, якщо кредит не стає дефолтом протягом перших k місяців – він визначається як «хороший».

Цільова змінна Bad визначається як 1 або 0 якщо кредит став, або відповідно не став дефолтним протягом періоду k . Дефолтом вважається досягнення протягом періоду k хоча б один раз певного рівня прострочки.

В нашому випадку модель відрізняється від «класичної». Її задача полягає в побудові скорингової карти з неперервною прогнозованою змінною. Результуючою змінною є частка неповернених відсотків по револьверній карті з грейсовим періодом протягом деякого фіксованого «outcome» періоду (6 місяців). Кожен запис по кредиту дублюється 100 разів. Таким чином, якщо рівень повернення відсотків по кредиту складає 73% то 73 записи будуть з «0», а 27 з «1».

НАВЧАЛЬНА ТА ТЕСТОВА ВИБІРКИ

З метою уникнення так званого «перенавчання» моделі (Overfitting), коли побудована модель показує високі показники предикативності на вибірці, на якій вона будувалася, проте на новій популяції ці показники екстремально падають, робиться розбиття базової вибірки (Sample Window) на:

– навчальну вибірку (70-80%) – використовується для побудови моделі;

– тестову вибірку (20-30%) – використовується для перевірки предикативної здатності моделі.

Записи було розбито на дві групи випадковим чином (генерація випадкового числа від 0 до 1 і розділення вибірки по значенню 0.3); можна проводити розбивання більш витонченим методом таким як стратифікація.

Стратифікація полягає у розбитті вибірки на однорідні сегменти по вибраних змінних (стратах). Такими змінними найчастіше виступають: сама результуюча змінна, вік, тип продукту, сума кредиту тощо. В середині кожної групи випадковим чином вибирається 70% вибірки з ознакою «train sample», решта 30% отримує ознаку «test sample». В такий спосіб після розбиття

на тестову та навчальну вибірку зберігаються розподіли кожної із страт.

ПРЕДИКАТИВНА СИЛА

Предикативна сила характеристики визначається з допомогою показника IV (informative value):

$$IV = \sum_{i=1}^k (\%good_i - \%bad_i) \circ \ln\left(\frac{\%good_i}{\%bad_i}\right)$$

де k – кількість атрибутів відповідної характеристики, $\%good_i$ та $\%bad_i$ – відповідно відсотки добрих та поганих клієнтів, які попали в атрибут « i ».

В залежності від значення IV можна зробити висновок про предикативну здатність характеристики:

– менша ніж 0,02 – непередикативна (тобто розподіляється однаково по різних сегментах результуючої змінної);

– від 0,02 до 0,1 – слабка предикативна здатність;

– від 0,1 до 0,3 – середня предикативна здатність;

– 0,3 – висока предикативна здатність.

Варто звернути увагу на те, що при екстремально високих значеннях IV (більше 1 для аплікаційних характеристик та більше 3-4 для поведінкових) включення такої характеристики в модель може призвести до overfitting'у (англ. перенавчання) скорингової моделі.

Часто при побудові аплікаційних карт у випадку слабкої предикативної здатності аплікаційних змінних варто будувати так звані крос-змінні, які дозволяють знаходити нестандартні зв'язки між змінними та посилювати нелінійність моделі.

Для прикладу, при побудові змінна «Стать» та «Освіта» показали слабку предикативну здатність. Проте при побудові крос-змінної вдалося провести значно кращу сегментацію і включити її у модель:

Таблиця 1.

Крос-змінна «Стать» та «Освіта»

Vars	Categ	Good	Bad	WOE	Badrate, %	IV
APPSEX_APPBIRTHDATE	APPSEX = 0 & APPBIRTHDATE <= 25	25 737	8 712	-0,16819	25,3	0,0012
APPSEX_APPBIRTHDATE	APPSEX = 0 & APPBIRTHDATE <= 40	152 759	40 579	0,07420	21	0,0013
APPSEX_APPBIRTHDATE	APPSEX = 0 & APPBIRTHDATE <= 50	93 184	21 178	0,23021	18,5	0,0069
APPSEX_APPBIRTHDATE	APPSEX = 0 & APPBIRTHDATE > 50	87 462	17 426	0,36182	16,6	0,0149
APPSEX_APPBIRTHDATE	APPSEX = 1 & APPBIRTHDATE <= 47	218 943	75 975	-0,19300	25,8	0,0140
APPSEX_APPBIRTHDATE	APPSEX = 1 & APPBIRTHDATE <= 52	30 259	9 706	-0,11436	24,3	0,0007
APPSEX_APPBIRTHDATE	APPSEX = 1 & APPBIRTHDATE > 52	35 396	10 599	-0,04558	23	0,0001
APPSEX_APPBIRTHDATE	total	643 740	184 175	0,00000	22,2	0,0390

Крім того показник IV (informative value) для цієї змінної набагато більший ніж сума показників IV кожної із змінних. Такий ефект називається синергетичним ефектом.

СЕГМЕНТАЦІЯ ХАРАКТЕРИСТИК

Сегментація (розбиття на атрибути) характеристик, що входять в скорингову модель, відіграє важливу роль для побудови стабільної та предикативної скорингової моделі. Правильна сегментація сприяє збільшенню предикативної сили та стабільності характеристики, а отже і скорингової моделі.

Сегментація проводиться як для неперервних характеристик при розбитті їх на категоріальні так і для

категоріальних при наявності атрибутів з малою часткою в вибірці та атрибутів подібних між собою.

Показник WOE (weight of evidence) для кожного атрибуту характеристики розраховується за формулою

$$WOE_i = \ln\left(\frac{\%good_i}{\%bad_i}\right)$$

де $\%good_i$ та $\%bad_i$ – відповідно відсотки добрих та поганих клієнтів, які попали в атрибут «i».

Класична таблиця після розбиття характеристики на атрибути має наступний вигляд:

Таблиця 2.

Розбиття характеристики на атрибути

Vars	Categ	Good	Bad	%, good	%, bad	WOE	Badrate, %	IV
APPBIRTHDATE	<= 25	45 930	16 020	7%	9%	-0,19812	25,9	0,0031
APPBIRTHDATE	(25;30]	114 782	36 349	18%	20%	-0,10154	24,1	0,0019
APPBIRTHDATE	(30;45]	272 153	80 166	42%	44%	-0,02914	22,8	0,0004
APPBIRTHDATE	(45;50]	76 978	20 350	12%	11%	0,07903	20,9	0,0007
APPBIRTHDATE	> 50	133 897	31 290	21%	17%	0,20236	18,9	0,0077

Від’ємні показники WOE вказують на те, що атрибут «поганий» (чим менше – тим гірший), тобто гірший ніж середній показник в портфелі, додатні – що «хороший».

Основні принципи, яких потрібно дотримуватись при сегментації характеристик:

– проводити сегментацію в такий спосіб, щоб IV кожної характеристики було максимальним (максимізація статистики Хі-квадрат) та зберігалась логічність такої сегментації. Логічність в даному випадку передбачає;

- монотонність для неперервних характеристик;
- бізнес-логіку – відповідність очікуваним результатам (пр. ми очікували що старші клієнти кращі);
- групування в одні атрибути сегментів з близьким значенням WOE.

Наприклад, якщо сегмент «вдівець/вдова» має близьке значення WOE до сегменту «неодружений/на», то їх варто об’єднувати в один сегмент, якщо сумарний показник IV характеристики при цьому зменшується несуттєво:

– намагатись уникати створення малих сегментів (<5% популяції);

– об’єднувати дуже дрібні сегменти з сегментом, значення WOE якого найближче до нуля;

– об’єднувати сегмент порожніх значень з сегментом, значення WOE якого найближче до нуля;

– у випадку якщо такий сегмент значний (> 2-3%) і його значення WOE суттєво відрізняється від нуля – це означає що порожні значення утворюються не випадково. Якщо сегмент таких відмінних порожніх значень можна пояснити і надалі очікується утворення таких порожніх значень, – лише тоді можна виділяти сегмент порожніх значень як окремий атрибут.

В результаті сегментації для побудови моделі було обрано такі характеристики:

- APPBIRTHDATE - вік клієнта у заявці;
- APPFAMILYSTATUS - сімейний стан;
- APPPOSTCATEG - займана позиція;
- APPSEX_APPBIRTHDATE - схрещений параметр статі та віку;
- APPSEX_APPEDUCATION - схрещений параметр статі та освіти;
- DEP_NUMACT_DEP_SUM - схрещений параметр кількості активних депозитів та їх суми;

- APPWPERIODG - робочий стаж;
- CHAR10 - кількість місяців до планованої дати закриття;
- COMPRO - форма влаштування;
- CHR4 - кількість активних депозитів за 2 роки;
- FLAT - наявність житла.

КОРЕЛЯЦІЯ ТА БІЗНЕС-ЛОГІКА

Включення/не включення змінних, що корелюють між собою є доволі суперечливим питанням серед спеціалістів по скорингу та статистиці, хоча більшість все-таки схиляється до того, щоб забороняти входження в одну модель змінних, кореляція між якими висока (> 0.7).

З одного боку включення змінних що корелюють між собою в одну модель не впливає негативно на її предикативну здатність (при різних видах моделювання: логіт-регресія, класифікаційне дерево, нейронна мережа, тощо), проте поява високої мультиколінеарності між змінними в моделі

може призводити до нелогічного (з точки зору бізнес-логіки) розрахунку вагових коефіцієнтів. Приклад: для гірших атрибутів змінної (по bad rate) проставляються вищі бали скоринговою картою ніж для кращих атрибутів.

При побудові моделей логіт-регресії, рекомендовано виключати слабшу по IV з двох характеристик, кореляція між якими перевищує 70%. При дуже великих наборах змінних (більше 100) ще до розрахунку кореляції можна відкидати змінні з меншим IV з груп однотипних змінних.

РЕЗУЛЬТАТИ

У результаті подальшого присвоєння скорингового балу для кожного значення параметра та подальшої оцінки її якості було визначено, що побудована математична модель з індексом GINI 24%. Динаміка індексу GINI коливається в межах допустимого рівня протягом всього розглянутого періоду, що свідчить про стабільність обраних параметрів.

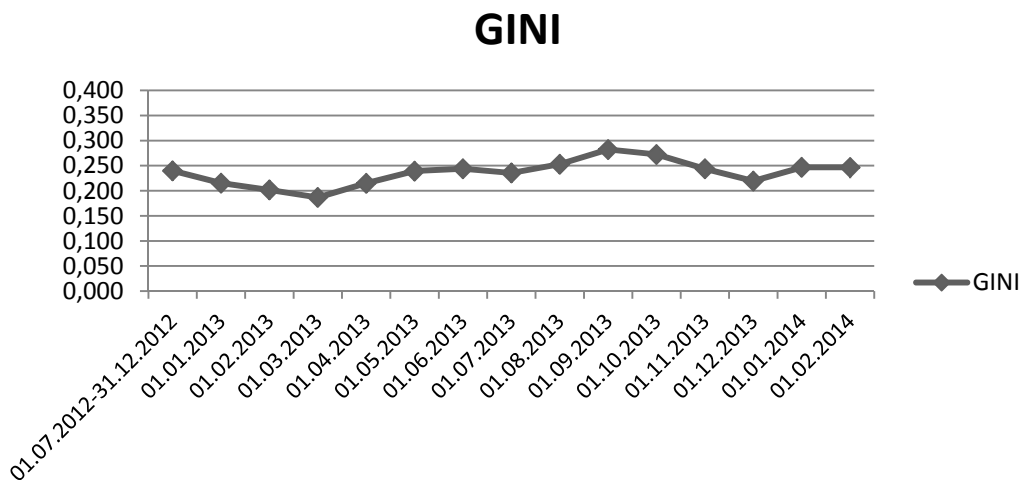


Рис. 2 Динаміка індексу GINI

Згідно зі створеною моделлю, граничні значення скорингового балу можуть коливатися від -92 до 205 (рис. 3). Концентрація популяції у скоринговому балі (розподіл з кроком 20) наведена також на рис. 3.

Дану сегментацію можна практично застосовувати починаючи із етапу розробки продукту, закінчуючи

прогнозуванням дохідності револьверної карти. Також, коригуючи пункти відтину, можна здійснювати контроль видачі тому чи іншому сегменту популяції і тим самим запобігти втратам, що можуть призвести до значного зниження кінцевого фінансового результату.

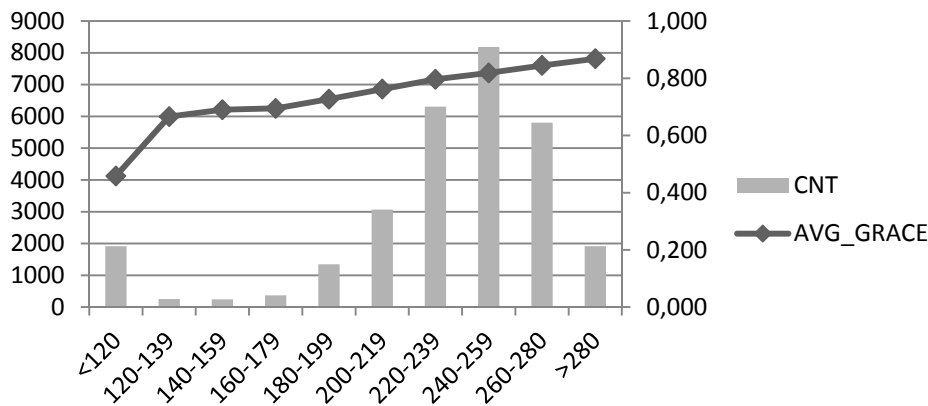


Рис. 3 Концентрація популяції у скоринговому балі

ВИСНОВКИ

В даній роботі було розглянуто проблему кредитно-го ризику, визначені причини його виникнення та підходи до оцінки. Проаналізовано супутні ризики в процесі кредитування на прикладі револьверних карток з грейсовим (пільговим) періодом. Було розглянуто основні етапи побудови скорингової моделі: формування вибірки, сегментації характеристик тощо.

Результатом роботи стала побудована аплікаційна скорингова модель яка, на основі визначених параметрів, дозволяє провести сегментацію клієнтів таким чином, щоб банк не зазнав збитків від недоотримання прибутку. Для побудованої моделі індекс GINI мав значення 24%. Динаміка індексу GINI коливається в межах допустимого рівня протягом всього розглянутого періоду, що свідчить про високу стійкість побудованої моделі до змін у даних.

ЛІТЕРАТУРА:

1. Osoblyvosti minimizatsii kredytnoho ryzyku bankivskoi ustanovy // I.V.Yeleiko, O.V.Sidak // Naukovyi visnyk NLTU. – 2011. – Вуп. 21.8.
2. Kaminskyi A. Ekspertna model kredytnoho skorynhu pozychalnyka banku // Bankivska sprava. – 2006. – №1. – S. 75-81.
3. Hussein A. Abdouab, Marc D. Dongmo Tsafack, Collins G.Ntimd, Rose D. Baker. Predicting creditworthiness in retail banking with limited scoring data. Knowledge-Based Systems Volume 103, 1 July 2016, Pages 89-103
4. Rais Ahmad Itoo, A. Selvarasu, José António Filipe. Loan Products and Credit Scoring by Commercial Banks (India). Int. J Latest Trends Fin. Eco. Sc. Vol-5 No. 1 March, 2015. pp. 851-860.
5. Bondarenko V. Skorynh-otsenka kredytoposobnosti zaemshchyka // Fynansovaia konsultatsiia. – 2005. – №1-2. – S. 13-16. 4.
6. H. Abdou, S. Alam, J. Mulkeen. Would credit scoring work for Islamic finance? A neural network approach Int. J. Islamic Middle Eastern Finance Manage., 7 (1) (2014), pp. 112-125.
7. H. Bekhet, S. Eletter Credit risk assessment model for Jordanian commercial banks: neural scoring approach Rev. Dev. Finance, 4 (1) (2014), pp. 20-28.
8. Rekomendatsii z pryvodu otsinky komertsiiynykh bankamy kredytopromozhnosti i finansovoi stabilnosti pozychalnyka / Natsionalnyi bank Ukrainy. № 23011/79 vid 02.06.94 r. – [Elektronnyi resurs]. – Rezhym dostupu: <http://www.bank.gov.ua>.
9. Kim M.-J., Kang D.-K., Kim H.B. Geometric mean based boosting algorithm with over-sampling to resolve data imbalance problem for bankruptcy prediction. Expert Syst. Appl., 42 (3) (2015), pp. 1074-1082.
10. Bidiuk P.I., Matros Ye.O. Modeli otsinky ryzykiv kredytuvannia fizychnykh osib // Kibernetika ta obchysliu-valna tekhnika. – 2007. – № 153. – С. 87–95.
11. Bellotti T. and Crook J. Forecasting and stress testing credit card default using dynamic models, International Journal of Forecasting, vol. 29, Issue 4, 2013, pp. 563–57.
12. Mishchenko V. I. Bankivski operatsii: Pidruchnyk / V. I. Mishchenko, N. H. Slavianska, O. H. Korenieva. – 2-e vyd., pererobl. i dop. – K.: Znannia, 2007. – S. 280–283.

Рецензент: д.т.н., проф. Марасанов В. В.
Херсонський національний технічний університет

ДОСЛІДЖЕННЯ МЕТОДІВ РОЗРАХУНКУ ВІДСТАНІ ДЛЯ ОКРЕМИХ СЕГМЕНТІВ МАРШРУТУ

УДК 004.23

ДЕРКАЧ Марина Володимирівна

асистент кафедри комп'ютерних наук та інженерії, Східноукраїнський національний університет ім. В. Даля

Наукові інтереси: інформаційні технології в інтелектуальних транспортних системах, бази даних.

СКАРГА-БАНДУРОВА Інна Сергіївна

д.т.н., завідувач кафедри комп'ютерних наук та інженерії, Східноукраїнський національний університет ім. В. Даля

Наукові інтереси: теорія прийняття рішень, інформаційні технології в промисловості, екології, медицині.

Сучасна тенденція розвитку міста призводить до збільшення кількості транспортних засобів у мережах дорожнього руху. Дорожній рух вимагає точної та найсучаснішої інформації про поточну ситуацію та доступні послуги. Покращення дорожніх умов є важливим для всієї інфраструктури. Проте лише це покращення не може задовольнити постійно зростаючим вимогам до безпечних, зручних, економічно ефективних та комфортних послуг дорожнього руху. До того ж саме рівень розвитку транспортної системи є однією з головних ознак економічної стабільності та процвітання міста. Єдина транспортна система має забезпечувати розвиток і функціонування всіх видів транспорту та максимально відповідати потребам міста при мінімальних витратах. За для вирішення цієї проблеми останні десятиріччя ведуться активні розробки та впровадження сучасних інтелектуальних транспортних систем, зокрема апаратно-програмних комплексів, побудованих за технологією IoT, призначених для інформування про час прибуття громадського транспорту на зупинки міста. Прогнозування часу руху транспортних засобів є найбільш затребуваним і зрозумілим завданням для кінцевого споживача, як учасника дорожнього руху.

Це перший крок для підвищення рівня доступності та поліпшення якості обслуговування в громадському транспорті міста, завдяки збереженню вільного часу населення, що марно витрачається на очікування транспорту на зупинках міста. Особливо актуально прогнозування часу прибуття транспорту в містах, які не ма-

ють чіткої відповідності з розкладом руху пасажирського транспорту до часу.

Але саме прогнозування часу прибуття транспорту на зупинки залишається одним із багатьох найскладніших завдань, що потребують вирішення. За для отримання надійного алгоритму, який може бути прийнятий для прогнозування часу прибуття міського громадського транспорту, перш за все треба знайти ефективний метод розрахунку відстані між географічними координатами місцезнаходження транспортного засобу, зупинок міста.

Протягом багатьох років було здійснено величезну роботу для виявлення ефективних методів, що дозволяють розрахувати відстань між географічними координатами, але однозначного рішення цієї проблеми досі не знайдено. Дане дослідження намагається порівняти ефективність методів розрахунку відстані для окремих сегментів маршруту для реального набору даних.

Мета роботи полягає в пошуку найкращого методу для розрахунку відстані між поточним місцезнаходженням транспортного засобу до цільової точки на різних ділянках маршруту.

Системи, що дозволяють прогнозувати прибуття міського транспорту, ефективно працюють у багатьох країнах світу таких, як Китай [1, 2], Великобританія, Нідерланди, Чехія, Румунія, Пакистан [3] та інші. Тим не менш, в Україні впровадження таких систем все ще знаходиться на стадії розвитку. Оскільки інноваційні

системи управління трафіком та інформаційної підтримки пасажирів спроможні зробити міський пасажирський транспорт більш безпечним, ефективним і зручним для усіх учасників руху, постає завдання розробити ефективний метод для прогнозу в реальному часі руху міського транспорту, а це важлива складова транспортної інфраструктури міста [4].

Одним з ключових етапів такого прогнозу є розрахунок відстані між географічними координатами, а саме для отримання траєкторії руху транспортного засобу, пошуку оптимального маршруту, оцінки пройденого шляху або відстані, що залишилось подолати до контрольної точки, тощо.

У геодезії така відстань розуміється як найкоротша відстань між двома точками на поверхні еліпсоїда, методами для отримання геодезичної ділянки можуть бути метод, заснований на формулах гаверсинусів, методи Вінсенті та Евклідової відстані. До того ж ці методи припускають, що земля являє собою сферу, в той час як метод Вінсенті припускає, що це еліпсоїд. У зв'язку з чим, необхідно врахувати такі чинники, як:

- радіус Землі (радіус еліпсоїда) не є постійною величиною, і залежить від широти даної точки;
- широта (геодезична широта) визначається кутом між площиною екватора і нормаллю до поверхні еліпсоїда в даній точці, на відміну від геоцентричної широти, яка визначається кутом між екваторіальною площиною та центром еліпсоїда.

Формули гаверсинусів використовували автори в роботі [5]. Вони вважають, що розрахункова відстань, отримана завдяки саме цьому методу, добре узгоджується з фактичною відстанню, має відхилення лише 0,001 милі. Під час практичної реалізації автором була використана евклідова відстань [6]. В роботі [7] автори наводять оцінку похибки методу Вінсенті, пов'язаному з наближенням урахуванням еліпсоїдної форми Землі при розрахунку відстаней між двома точками на її поверхні. Показують, що похибка, яка виникає, стає сумірною похибці GPS / ГЛОНАСС навігаторів при розрахунку відстаней величиною понад 200 км. Формули гаверсинусів також використовували у своїх роботах автори [8, 9, 10], евклідову відстань застосовували автори [8, 11]. Для досягнення поставленої мети та

вирішення задачі в роботі розглянуто наступні методи: метод, заснований на формулах гаверсинусів, метод Вінсенті та Евклідової відстані.

Евклідова відстань. Евклідова відстань - це зазвичай найкоротший шлях, виміряний вздовж транспортної мережі. Прямолінійна (евклідова) відстань найчастіше використовується через легкість її розрахунку, тому що відстані виражається в географічних одиницях, таких як кілометри, завдяки простій функції прямої лінії між двома місцями. Зазвичай використовується для наближення відстані, але рідко має практичне застосування. Тобто маючи дві точки на земній поверхні, задані широтою та довготою, можна розрахувати відстань між ними безпосередньо, крізь Землю. Широта і довгота є частиною сферичних координат в тривимірному просторі, а саме: радіус - це радіус Землі, кут нахилу - це широта, і азимут - це довгота. У разі перетворення сферичних координат в Декартові координати x , y , z , знайти відстань напряму між цими точками, використовуючи формулу Евклідової відстані, можна наступним чином:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}. \quad (1)$$

Метод Вінсенті. Вінсенті [12] розробив метод для розрахунку геодезичних відстаней між парою точок широти та довготи на земній поверхні, використовуючи точну еліпсоїдальну модель землі. Це набір формул, які ґрунтуються на ітераційній процедурі, але кількість ітерацій, необхідних для рішення, як правило становить три чи чотири. Метод обчислює відстань між двома координатами, початковий шляховий кут, кінцевий шляховий кут, а також путні кути в проміжних точках. Метод Вінсенті включає прямий та зворотний метод, де: прямий метод обчислює розташування точки, якщо задана відстань та азимут від іншої точки, а інверсний метод обчислює географічну відстань та азимут між двома заданими точками. Як згадувалося вище, формули є ітераційними процесами, що означає, що розраховується послідовність рівнянь, коли виводиться повторно в ту ж послідовність рівнянь. Мета полягає в тому, щоб мінімізувати вихідне значення після встановленої кількості ітерацій. Цей метод, тобто набір формул, використовуються у наступному скрипті:

$$\begin{aligned}
 &a, b = \text{major \& minor semi-axes of the ellipsoid} \\
 &f = \text{flattening } (a-b)/a \\
 &\varphi_1, \varphi_2 = \text{geodetic latitude} \\
 &L = \text{difference in longitude} \\
 &\tan U_{1/2} = (1-f) \cdot \tan \varphi_{1/2} \\
 &\cos U_{1/2} = 1 / \sqrt{1 + \tan^2 U_{1/2}}, \sin U_{1/2} = \tan U_{1/2} \cdot \cos U_{1/2} \\
 &\lambda = L \text{ (first approximation)} \\
 &\text{iterate until change in } \lambda \text{ is negligible (e.g. } 10^{-12} \approx 0.006\text{mm)} \{ \\
 &\sin \sigma = \sqrt{[(\cos U_2 \cdot \sin \lambda)^2 + (\cos U_1 \cdot \sin U_2 - \sin U_1 \cdot \cos U_2 \cdot \cos \lambda)^2]} \\
 &\cos \sigma = \sin U_1 \cdot \sin U_2 + \cos U_1 \cdot \cos U_2 \cdot \cos \lambda \\
 &\sigma = \text{atan}(\sin \sigma / \cos \sigma) \\
 &\sin a = \cos U_1 \cdot \cos U_2 \cdot \sin \lambda / \sin \sigma \\
 &\cos^2 a = 1 - \sin^2 a \\
 &\cos 2\sigma_m = \cos \sigma - 2 \cdot \sin U_1 \cdot \sin U_2 / \cos^2 a \\
 &C = f/16 \cdot \cos^2 a \cdot [4 + f \cdot (4 - 3 \cdot \cos^2 a)] \\
 &\lambda' = L + (1 - C) \cdot f \cdot \sin a \cdot \{ \sigma + C \cdot \sin \sigma \cdot [\cos 2\sigma_m + C \cdot \cos \sigma \cdot (-1 + 2 \cdot \cos^2 2\sigma_m)] \} \\
 &u^2 = \cos^2 a \cdot (a^2 - b^2) / b^2 \\
 &A = 1 + u^2/16384 \cdot \{4096 + u^2 \cdot [-768 + u^2 \cdot (320 - 175 \cdot u^2)]\} \\
 &B = u^2/1024 \cdot \{256 + u^2 \cdot [-128 + u^2 \cdot (74 - 47 \cdot u^2)]\} \\
 &\Delta\sigma = B \cdot \sin \sigma \cdot \{ \cos 2\sigma_m + B/4 \cdot [\cos \sigma \cdot (-1 + 2 \cdot \cos^2 2\sigma_m) - B/6 \cdot \cos 2\sigma_m \cdot (-3 + 4 \cdot \sin^2 \sigma) \cdot (-3 + 4 \cdot \cos^2 2\sigma_m)] \} \\
 &s = b \cdot A \cdot (\sigma - \Delta\sigma) \\
 &a_1 = \text{atan}(\cos U_2 \cdot \sin \lambda / \cos U_1 \cdot \sin U_2 - \sin U_1 \cdot \cos U_2 \cdot \cos \lambda) \\
 &a_2 = \text{atan}(\cos U_1 \cdot \sin \lambda / -\sin U_1 \cdot \cos U_2 + \cos U_1 \cdot \sin U_2 \cdot \cos \lambda)
 \end{aligned}$$

Де s - геодезична відстань уздовж поверхні еліпсоїда,
 α_1 - перший азимут,
 α_2 - кінцевий азимут (у напрямку p1 → p2).

$\Delta\sigma$ – кутова різниця, що розраховується за формулою (3):

$$\Delta\sigma = 2 \arcsin \left\{ \sqrt{\sin^2 \left(\frac{\varphi_2 - \varphi_1}{2} \right) + \cos \varphi_1 \cos \varphi_2 \sin^2 \left(\frac{\Delta\lambda}{2} \right)} \right\}, \quad (3)$$

Метод, заснований на формулах гаверсинусів, є дуже точним методом обчислення відстаней між двома точками на поверхні сфери з використанням широти та довготи двох точок. Цей метод являє собою реформування сферичного закону косинусів, але формулювання з точки зору гаверсинусів є більше корисним для малих кутів та відстаней. Фактично, відстань d між двома точками є добутком кутової різниці та радіусу:

$$d = \Delta\sigma \cdot R, \quad (2)$$

де R – радіус Землі;

де $\varphi_1, \lambda_1; \varphi_2, \lambda_2$ – широта та довгота двох точок відповідно;

$\Delta\lambda$ – різниця координат за довжиною;

$\Delta\sigma$ – кутова різниця.

Обчислення довжини маршруту проводиться для реального набору даних з використанням інформації про географічні координати зупинок міста. Розрахунок відстані здійснювався на реальних вхідних даних, за основу були взяті географічні координати тролейбусних зупинок початку та кінця чотирьох діючих маршрутів міста Сєвєродонецьк (табл. 1).

Таблиця 1

Маршрути міста Северодонецьк

Номер маршруту	Координати початкової зупинки	Координати кінцевої зупинки	Кількість додаткових контрольних точок
1	(48.933256, 38.520513)	(48.951085, 38.468542)	3
2	(48.943869, 38.536434)	(48.951085, 38.468542)	5
5	(48.930282, 38.490000)	(48.951085, 38.468542)	5
6	(48.943869, 38.536434)	(48.961398, 38.460732)	5

Виходячи з отриманих даних (табл. 2), маємо суттєве відхилення від фактичної довжини маршруту. Це пов'язано з тим, що запропоновані методи повертають значення найкоротшої відстані між двома точками, як

показано на рисунку 1а. Але такий результат також дає підставу для наступного припущення: є сенс розробити метод, який на основі існуючих надав би результат близький до дійсного.

Таблиця 2

Отримані результати розрахунку відстані між початком та кінцем маршруту міста

Назва методу Номер маршруту	Формула гаверсинусів		Формула Вінсенті, км		Евклідова відстань, км		Фактична відстань, км
	Відстань, км	Відхилення, %	Відстань, км	Відхилення, %	Відстань, км	Відхилення, %	
1	5,39	26,47	4,42	39,7	4,50	38,61	7,33
2	5,05	28,17	5,06	28,02	5,06	28,02	7,03
5	2,81	71,79	2,78	72,09	2,81	71,79	9,96
6	5,90	50,34	5,92	50,17	5,90	50,34	11,88
Загальне відхилення, %	44,19		47,49		47,19		

Для покращення результатів здійснено лінеаризацію маршрутів, повторно використано три методи для обчислення відстані та проведено порівняння з фактичною довжиною маршруту для отримання оцінки ефективності роботи методів.

Оскільки відстань між контрольними точками маршруту часто неможливо визначити як довжину прямої лінії між цими точками, кожний маршрут було розбито на дрібніші сегменти, довжини яких можна складати, отримуючи відстані між контрольними точками (4), що являє собою лінеаризацію маршрутів (рис. 1б).

$$d = \sum_{i=1}^{N-1} d_i, \quad (4)$$

де d – відстань між двома точками;

d_i – відстань між додатковими точками;

i – індекс сегменту;

N – кількість контрольних точок.

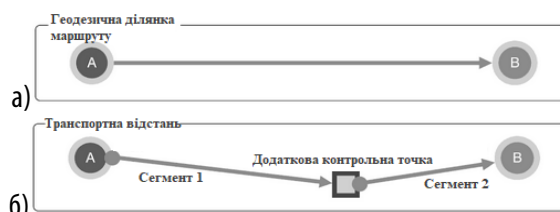


Рисунок 1 – Різниця між геодезичною ділянкою та фактичним маршрутом транспорту

Довжина сегмента, що представляє пряму лінію, при цьому може бути визначена за формулами гаверсинусів, методами Вінсенті та Евклідової відстані, відповідно. Відстані між початком маршруту, додатковими точками, тобто зупинками, та кінцем маршруту представлені у таблиці 3.

Таблиця 3

Отримані результати розрахунку відстані між початком та кінцем маршруту міста після лінеаризації

Назва методу Номер маршруту	Формула гаверсинусів		Формула Вінсенті, км		Евклідова відстань, км		Фактична відстань, км
	Відстань, км	Відхилення, %	Відстань, км	Відхилення, %	Відстань, км	Відхилення, %	
1	7,422	1,26	7,043	3,92	7,425	1,3	7,33
2	6,628	5,72	6,633	5,65	6,622	5,8	7,03
5	9,917	0,43	9,544	4,18	9,922	0,38	9,96
6	11,313	4,77	11,336	4,58	11,318	4,73	11,88
Загальне відхилення, %	3,04		4,58		3,05		

Продовжуючи дослідження на прикладі міста Северодонецьк, крім початкової та кінцевої зупинки були обрані додаткові контрольні точки для кожного маршруту, в якості яких виступили зупинки на ділянках доро-

ги, що мають поворот (рис. 2). А саме для маршруту №1 використали 3 додаткові контрольні точки, для усіх інших по 5 (табл. 1).



Рисунок 2 – Діючі маршрути міста Северодонецьк

За оцінкою отриманих результатів зроблено наступний висновок: на відміну від попереднього випадку, де відстань визначалась, як найкоротший шлях між двома географічними координатами, лінеаризація маршрутів дозволила урахувати відстань між точками за існуючим маршрутом, який включає увесь нерівний шлях, що значно зменшило загальне відхилення. Найбільш прийнятним методом розрахунку відстані для окремих сегментів маршруту виявився метод, заснований на формулах гаверсинусів. Використовуючи лінеаризацію, відхилення становить лише 3,04 %, при чому точність розрахунків напряму залежить від кількості додаткових контрольних точок.

ВИСНОВКИ

Технології, що базуються на інтелектуальних транспортних системах, широко застосовуються в управлінні дорожнім рухом, завдяки тому, що містять засоби інформування учасників дорожнього руху, системи контролю та оптимізації маршрутів, засоби і технології управління при надзвичайних ситуаціях, тощо. Покращення послуг громадського транспорту є важливим аспектом. В цьому сенсі одним з найбільш затребува-

них і зрозумілих для кінцевого споживача, як учасника дорожнього руху, завдань є прогнозування часу руху транспортних засобів. Вирішення цього завдання дозволяє контролювати рух міського транспорту, ввести своєчасні коригування диспетчерськими службами, сповістити пасажирів про час прибуття громадського транспорту на зупинки. Завдяки зібраній в реальному часі, інформації транспортні підприємства отримують можливість планувати, керувати та контролювати транспортні засоби, що суттєво вдосконалює сервіс.

Напрямок цього дослідження було направлено для отримання належного методу розрахунку відстані між географічними координатами для подальшого використання в системі прогнозування часу прибуття громадського транспорту на зупинки міста. Було розглянуто три методи, а саме метод, заснований на формулах гаверсинусів, методи Вінсенті та Евклідову відстань. Вибір методу розрахунку відстані суттєво впливає на кінцевий результат, а попередній висновок дає змогу вважати, метод, заснований на формулах гаверсинусів, доцільним для використання у подальшій роботі.

**ЛІТЕРАТУРА:**

1. Hua X. Bus arrival time prediction using mixed multi-route arrival time data at previous stop / X. Hua, W. Wang, Y. Wang, M. Ren // Transport ISSN 1648-4142 / eISSN 1648-3480. – 2015.
2. Bai C. Dynamic Bus Travel Time Prediction Models on Road with Multiple Bus Routes / C. Bai, Z.-R. Peng, Q.-C. Lu, J. Sun // Computational Intelligence and Neuroscience. – 2015. – Vol. 2015, doi:10.1155/2015/432389.
3. Asfandyar M. Real-Time Public Transport Arrival Information System / Student Research Paper Conference. – 2015. – Vol. 2, No. 13, PP. 67-71.
4. Fabbri G. An Innovative System for a Clean and Sustainable Public Transport System in Smart Cities / G.Fabbri, C.M. Medaglia, M. Ippolito, E. Saraceno, M. Antonucci, L. Fiorentino, M. Bistolfi, P. Cozzolino, M. Gallarate // Proceedings of the IEEE 25th International Symposium on Industrial Electronics (ISIE), Santa Clara, CA, USA. 8-10 June 2016. – pp. 974-979.
5. Grinshpan A. Comparative Analysis of GPS Data / A.Grinshpan, S.Campbell // Undergraduate Journal of Mathematical Modeling: One + Two, 2014. – Vol. 5, Iss. 2.
6. Topolska K. Model of hierarchical regression in decision problems of transportation systems // Archives of Transport System. Telematics, 2017. – Vol. 10, Iss. 2.
7. Кукало И.А. Линейное отображение географических координат подвижного объекта на линейную часть магистрального нефтепровода / И.А. Кукало, С.Н. Гривцов // Известия Томского политехнического университета. Инжиниринг георесурсов, 2015. – Т. 326, № 11, С. 31–43.
8. Biswas A. Microsegmenting: An approach for precise distance calculation for GPS based ITS applications/ A.Biswas, G.Pilla, B.Tamma // IEEE Recent Advances in Intelligent Computational Systems (RAICS), 2013.
9. Basyir M. Determination of Nearest Emergency Service Office using Haversine Formula Based on Android Platform / M.Basyir, M.Nasir, Suryati, Widdha Mellyssa // EMITTER International Journal of Engineering Technology, 2017. – Vol. 5, No. 2.
10. Berlin M. A. Safety Distance Calculation for Collision Avoidance in Vehicular Ad hoc Networks / M. A. Berlin, S. Muthusundari // Scholars Journal of Engineering and Technology (SJET), 2016. – 4(1). P. 63-69.
11. Gonçalves D. Analysis of the difference between the Euclidean distance and the actual road distance in Brazil / Daniel Neves Schmitz Gonçalves et al. // Transportation Research Procedia 3, 2014. – P. 876 – 885.
12. Thomas C. M. Validation of Vincenty's Formulas for the Geodesic Using a New Fourth-Order Extension of Kivioja's Formula / C.M. Thomas, W E. Featherstone // Journal of Surveying Engineering © ASCE. – 2005.

*Рецензент: д.т.н., проф. В.Ф. Коваленко
Херсонський національний технічний університет*

МЕТОДИКА ПРИЙНЯТТЯ РІШЕННЯ ПО МІСЦЮ РОЗТАШУВАННЯ ПРОДУКЦІЇ В СКЛАДСЬКИХ ПРИМІЩЕННЯХ

УДК 004.75, 65.011.56

ДІДУК Віталій Андрійович

кандидат технічних наук, Черкаський національний університет імені Богдана Хмельницького, м. Черкаси, Україна

Наукові інтереси: автоматизація виробничих та невиробничих процесів, мікропроцесорна техніка, робототехніка, професійна підготовка майбутніх інженерів.

e-mail: inokc@bk.ru

Для задоволення вимог замовника до якості, своєчасності поставок і цінами вся продукція, готова до споживання, повинна бути своєчасно, з мінімальними витратами і без шкоди для якості доставлена в необхідній кількості в обумовлені терміни до тих споживачів, які проявили на неї попит. Логістика у сфері розподілу являє собою комплекс стратегічних, організаційних, фінансових та інших заходів, тісно пов'язаних між собою в гнучку систему управління матеріальними, інформаційними, фінансовими та іншими потоками в невиробничий період

Логістика вивчає і здійснює наскрізне управління матеріальними потоками, тому вирішувати різні завдання розподільного характеру доводиться на всіх етапах:

- розподіляються замовлення між різними постачальниками при закупівлі товарів;
- розподіляються вантажі по місцях зберігання при надходженні на підприємство;
- розподіляються матеріальні запаси між різними ділянками виробництва;
- розподіляються матеріальні потоки в процесі продажу і т. д.

Розподіл матеріальних потоків вже досить давно є суттєвою стороною господарської діяльності. Проте нинішнє становище однієї з найбільш важливих функцій воно набуло лише порівняно недавно. У країнах з розвинутою ринковою економікою в 50-х - початку 60-х років системи розподілу розвивалися в значній мірі стихійно. Окремі підфункції, які в сукупності утворюють

функцію розподілу, трактувалися як самостійні функції управління. Інтегрований погляд на функцію розподілу отримав розвиток в 60-х - початку 70-х років. У цей період прийшло розуміння того, що об'єднання різних функцій, що стосуються розподілу виробленого продукту в єдину функцію управління, несе в собі великий резерв підвищення ефективності. Результатом інтегрованого підходу до реалізації різних функцій розподілу стало включення розподілу в структуру функціонального управління організацій і підприємстві.

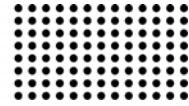
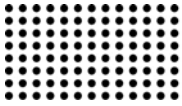
У торгівлі прикладом інтеграції різних підфункцій розподілу може служити виділення і розвиток спеціальних структур, що займаються постачанням магазинів. У країнах з розвинутою ринковою економікою такі структури розподілу в 70-і роки стали створювати і розвивати великі ланцюги роздрібної торгівлі. Вітчизняний досвід представлений організацією централізованої доставки товарів в магазини експедиційними підрозділами торгових оптових баз [3].

З ростом товарообороту та збільшенням запитів від споживачів, питання ефективної організації розподілу продукції на складах та організація логістики їх доставки набирає все більшої актуальності.

Метою роботи є розробка методики планування ефективного розміщення продукції на складських приміщеннях.

ОСНОВНІ РЕЗУЛЬТАТИ

В умовах швидкозростаючої конкуренції компанії, що займаються збереженням та транспортуванням продукції



змушені постійно вдосконалювати методи, спрямовані на управління складом, підвищувати його продуктивність, а також рівень обслуговування клієнтів. Зростання попиту на високий рівень обслуговування призводить до необхідності підвищення швидкості і точності інвентаризації, своєчасного обслуговування поставки, можливості виконання індивідуального замовлення, надання гнучкого сервісу з доданою вартістю і здатності реагувати на спеціальні запити клієнтів [2].

Для того щоб вирішити перераховані вище завдання, багато компаній застосовують інноваційний підхід – систему управління складом (англ. Warehouse Management System). Система управління складом (WMS) являє собою програмний додаток, яке спеціалізується на оптимізації, частковій автоматизації та підтримки повсякденних операцій в межах складу. При цьому WMS-системи можуть бути як частиною ERP-системи (англ. Enterprise Resource Planning), так і автономними, що включають в себе складні AIDC- технології. Система WMS здійснює збір, зберігання і передачу інформації, яка необхідна для ефективного управління матеріальним потоком всередині складу, контролює рух матеріалів в межах складу, виробляє точний і своєчасний облік всіх товарів шляхом реєстрації складських операцій і оптимізує запаси за допомогою постійного контролю інформації про їх стан [1, 2]. Також досить часто підприємства обмежуються звичайними системами обліку продукції на складі.

Відмінності між WMS і обліковими системами, за великим рахунком, стають зрозумілі вже виходячи з назви класифікації програмного продукту – система управління і система обліку, не рахуючи функціональних аспектів.

Якщо говорити про WMS-системи, то основне їхнє призначення – це управління технологічними процесами на складі. Професійні рішення забезпечують автоматизацію, оптимізацію і контроль всіх складських процесів в наскрізному режимі (приймання, розміщення, зберігання, комплектація вантажів і багато іншого, в залежності від класифікації і набору функцій програмного продукту). Також WMS дозволяють забезпечити контроль і оптимізацію роботи персоналу, техніки та догляд за садом і багато іншого.

До «облікового» функціоналу можна віднести таку реалізацію процесу, коли співробітник самостійно вирішує, як виконати операцію набору. Співробітник оцінює необхідний обсяг, типорозміри тари і вантажної техніки, вибирає собі конкретні завдання або їх групи у вигляді списку накладних, під час виконання завдань вирішує, чи буде він їх виконувати або пропустить, щоб повернутися потім.

У випадку з WMS співробітник отримує вже сформовані, конкретні вказівки від системи, а саме: взяти задані типорозміри тари, використовувати задану техніку, при спробі відмови від завдання – чітко вибирати причину, щоб система могла ініціювати запуск іншого, паралельного процесу (наприклад, контроль кількості в осередку, контрольний перерахунок вантажів з певними атрибутами, перевірка ваго-габаритних характеристик і інше) [4].

Чим більше залежність від співробітників, тим більша ймовірність похибок в роботі та менша швидкість структури в цілому.

Проте навіть WMS системи не володіють достатньо розвиненими алгоритмами прийняття рішення, аби в повній мірі автоматизувати роботу складу з максимальною ефективністю використання площ та часу обробки завдань.

Загальні процеси обробки потоків продукції умовно можна розподілити на 3 складові [3]:

1. Доведення матеріального потоку до виробництва;
2. Управління процесом проходження потоку всередині виробництва;
3. Управління процесом доведення готової продукції до споживача.

Найбільший науковий інтерес становить пункт №2 – управління процесом проходження потоку всередині виробництва, оскільки на сьогоднішній день він є найменш досліджений.

Для здійснення повної автоматизації прийняття рішення по розміщенню товарів в складських приміщеннях потрібно розглянути значну кількість факторів, що впливатимуть на нього.

Загалом, при автоматизації прийняття рішень в складських приміщеннях, варто розпочати з відповідного зонування території. Перелік основних технологіч-

них операційних ділянок (зон) для раціонального розміщення товару на складі, які повинні бути пов'язані один з одним проходами і проїздами:

- ділянка розвантаження
- ділянка приймання за кількістю і якістю
- ділянка зберігання
- ділянка комплектування замовлень
- ділянка відвантаження

Оптимальне зонування складу: ділянка розвантаження примикає до зони приймання за кількістю і якістю. Ділянка зберігання примикає до зони комплектування замовлень і знаходиться поруч з ділянкою відвантаження.

У зоні приймання за кількістю і якістю ведеться облік прибулого товару, облік тимчасового зберігання до передачі товару в зону зберігання складу.

При підготовці товару до зберігання формуються місця зберігання. У зону зберігання товари надходять із зони приймання або зони розвантаження [6]. Зона зберігання є найбільш відповідальною.

Зберігання продукції на складах необхідно виконувати таким чином, щоб при наступних технологічних операціях число переміщень складських службовців було мінімальним. З цієї метою ділять всі товарні позиції на три групи, після чого для їх зберігання виділяють «гарячі» і «холодні» складські зони. Щоб оптимально розділити всю номенклатуру, необхідно скористатися методиками ABC і XYZ-аналізу. В даному випадку основним критерієм розподілу товарних позицій на групи стане виступати число підходів / переміщень складського персоналу при виконанні технологічних операцій, зокрема при процедурі комплектації замовлень від виробництв або клієнтів.

Для проведення ABC або XYZ-аналізу необхідно зібрати статистичні дані, переважно не менше ніж за попередній рік. Інформація повинна містити: найменш за кожною товарною позицією, число замовлень і число відвантажених універсальних вантажних одиниць помісячно по кожній товарній позиції. Після складання статистичної бази весь товарний асортимент по ABC-аналізу ділять на три групи. Як правило, поділ проводиться в класичних пропорціях: група А - 20%, група В - 30%, група С - 50%. Критерієм поділу є число замовлень по кожній товарній позиції.

Щоб вирішити кінцеве завдання про розміщення товарів у «гарячу» і «холодні» зони, необхідно скласти матрицю результатів ABC, XYZ-аналізу. Зазвичай зіставлення результатів проведеного аналізу проводиться в MS Excel за допомогою функції сортування і займає мінімальну кількість часу.

Таким чином, після розстановки всіх позицій за відповідними категоріями отримують інформацію щодо розподілу продукції по зонах. Зазвичай, до «гарячої» зони зберігання будуть віднесені позиції, що знаходяться в квадратах AX, AY, BX, до «середньої» зони зберігання – позиції, що знаходяться в квадратах AZ, BY, CX, і до «холодної» зони зберігання – позиції, що знаходяться в квадратах BZ, CZ, CY [5].

Навіть за такого підходу, інформація про розташування продукції не є повною, оскільки безперервний обіг продукції неминуче призведе до її змішування і втрати часу на пошук потрібної позиції.

Угрупування товарів в зонах зберігання в залежності від оборотності, ліквідності, частоти звернень. Розміщення груп товарів з більш високими показниками ближче до виходу із зони і на нижніх ярусах зберігання, відбір з яких ведеться без використання допоміжних інструментів і техніки.

Угрупування товарів за обсягом упаковки. Розміщення товарів відбувається так, щоб упаковки з великими обсягами знаходилися на початку маршруту комплектувальника, а упаковки меншого обсягу – ближче до кінця. Таким чином, здійснюючи відбір товару на замовлення клієнтів, комплектовщик складає на піддон або в короб спочатку великі упаковки, а потім дрібні, не витрачає час на перекладання товару в процесі відбору [8].

Угрупування товарів в місцях зберігання. Угрупування товарів за частотою звернення. Розміщення груп товарів з більш високою частотою звернення на ярусах, відбір з яких ведеться без використання допоміжних інструментів і техніки.

Угрупування товарів за вагою упаковки. Розміщення груп товарів з великою вагою упаковки на нижніх ярусах стелажа, що скорочує трудовитрати персоналу складу по розміщенню і відбору товару.

Автоматичне визначення мінімально потрібного осередку зберігання для товару. Для реалізації цього методу необхідно створити довідник габаритів всіх

осередків зберігання на складі, а також фіксувати об'ємно-вагові характеристики вступників товарів (транспортних упаковок і штук) [7].

З огляду на це, можна сказати, що кожна продукція на складі може володіти певним комплексним показником K , що вказуватиме на довготривалість зберігання, вагу, габаритні розміри, частота клієнтських запитів. Таким чином, чим вищий показник даного коефіцієнта – тим ближче до зони відвантаження він має знахо-

дитись, а отже, й сама «гаряча зона» з ділянками АХ, АУ, ВХ. З'являється можливість навіть в вузьких рамках однієї зони провести градацію її розподілу та підвищити ефективність розташування в них продукції.

З огляду на розглянуту інформацію, можна побудувати спрощену модель автоматизованої комп'ютерної системи прийняття рішення розташування продукції на складі (рис. 1).

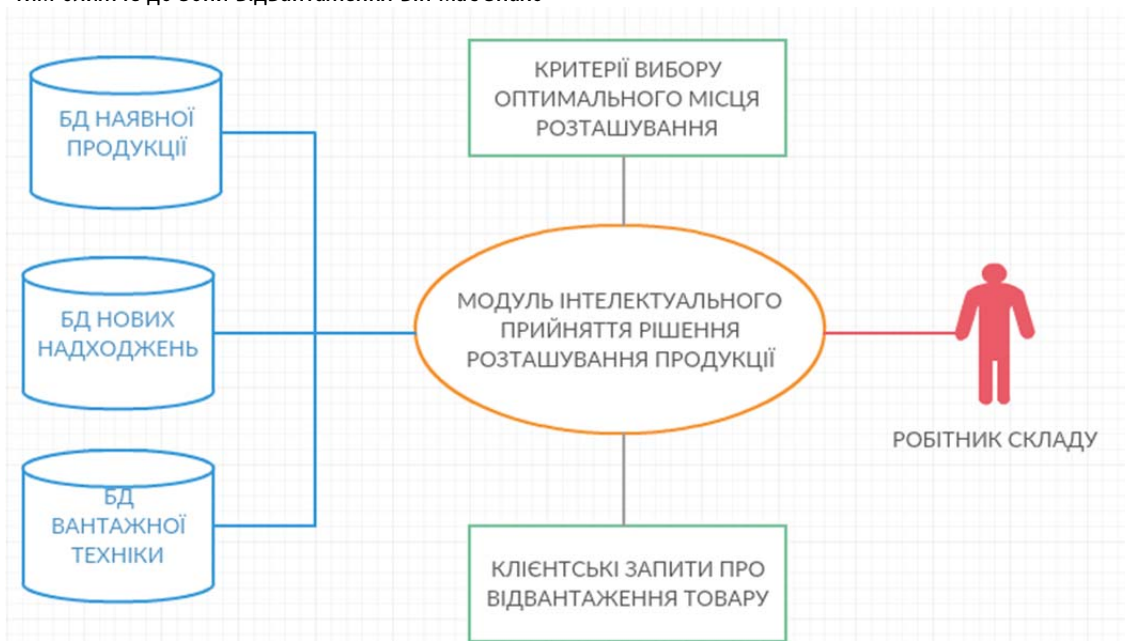


Рис. 1. Блок-схема спрощеної моделі автоматизованої системи прийняття рішення розташування продукції на складі

Дана модель дозволить модернізувати існуючі WMS системи таким чином, що система не тільки вестиме облік наявного товару та складської техніки з генерацією звітності та завдань, а й в реальному часі пропонуватиме оптимальні варіанти розташування продукції на території складу.

Даний підхід дозволить здійснити також перехід до повністю автоматичного складу за умови наявності автоматичних завантажувачів та розвинутої системи ідентифікації зберезуваних об'єктів. Оскільки товари зберігаються в типовій тарній упаковці, застосовувати елементи машинного зору є недоцільним.

При наявності автоматичних складських завантажувачів для обслуговування складських приміщень достатньо буде лише однієї людини, яка контролюватиме ведення облікової інформації та встановлюватиме для

кожного товарного надходження додаткові вагові коефіцієнти чи інші мітки.

Таким чином, дану методику можна застосувати в моделі процесу надходження товару на складі, що розроблена в роботі [2].

Після надходження товару на склад його не доведеться розміщувати в будь-якому доступному місці. Суть роботи працівника зведеться переважним чином до прийомки товару та внесення супровідної інформації, у разі її відсутності, до системи. Для оптимізації роботи складу та переведення його в автоматичний режим система повинна буде додатково розрахувати критерії оптимізації розміщення товарів на складі. До таких критеріїв можна віднести час зберігання товару, маса, цінність та інше.

Ваговий коефіцієнт визначатиметься в системі як відношення часу на його транспортування в середині

складських приміщень до прогнозованого часу його зберігання:

$$k = \frac{t_{\text{трансп}}}{t_{\text{зберігання}}} \quad (1)$$

де, k – ваговий коефіцієнт товару, $t_{\text{трансп}}$ – час, затрачений на транспортування товару по території складу, $t_{\text{зберігання}}$ – орієнтовний час зберігання товару, протягом якого з товаром не буде здійснюватися ніяких операцій [2].

Під час попередньої оцінки, застосування вагового коефіцієнта, оскільки час, затрачений на транспортування товару по території складу є набагато меншим за орієнтовний час зберігання товару, $t_{\text{трансп}}$ можна знехтувати і прирівняти

до 1. Тоді оцінка показує, що при наближенні значення коефіцієнта до одиниці, його потрібно розташовувати найближче до входу, чим менше від одиниці буде обраховане значення коефіцієнту, тим далі можна розташувати товари в приміщенні складу.

Для прикладу, якщо здійснювати повну автоматизацію складу та введення в його склад автоматичних завантажувачів, або мобільних маніпуляторів, весь склад потрібно розділити на сектори по категоріях товарів. В кожному секторі повинні міститись стійки, для змоги розміщення вантажів в декілька ярусів. За такого підходу важчі товари мають розташовуватись в нижніх ярусах, легші – у верхніх. Цінні товари мають зберігатися в окремих під-секторах, для усунення ймовірності випадкового пошкодження під час роботи з іншими товарами.

Таким чином, раніше запропонована модель діаграми процесу надходження товару на склад, що запропонована в [2], включно з блоком розрахунку вагового коефіцієнту набуде вигляду, приведеного на рис. 2.

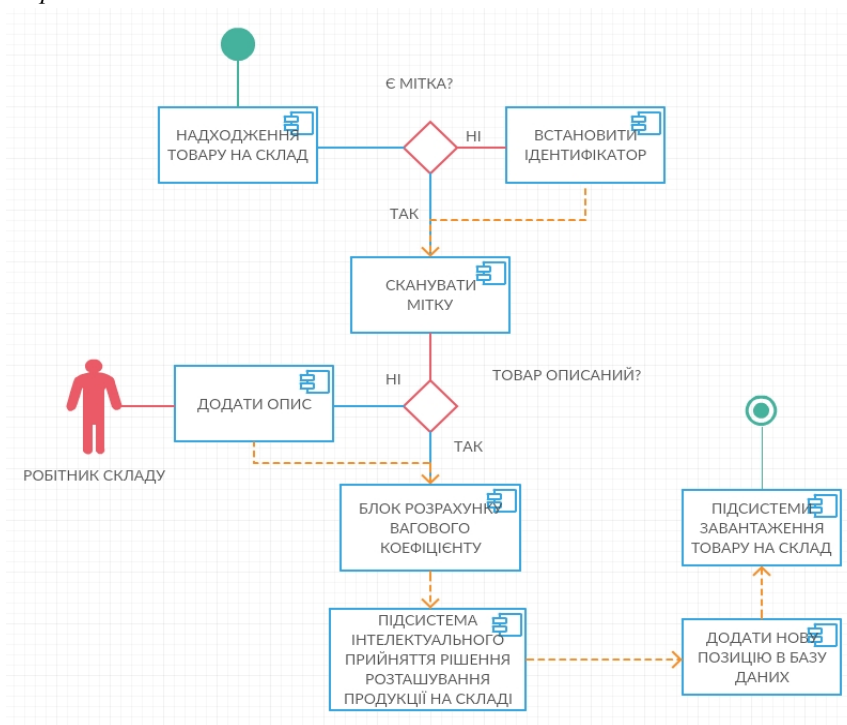
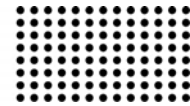
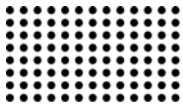


Рис. 2. Удосконалена спрощена діаграма процесу надходження товару на склад

У ході досліджень та аналізу наявних технологій, оптимальним варіантом використання в якості міток є NFC / RFID мітки [9, 10]. На сьогоднішній день такий підхід також використовується, проте використовувати мітки дають можливість зчитати тільки ідентифікатор.

Вся інша інформація має міститися в системній базі даних. Під час інспекції складських приміщень забезпечити зв'язок з центральними системами не завжди виявляється можливим, тому, в роботі пропонується застосувати технологічні NFC / RFID мітки, які дозволять



також містити в собі іншу інформацію, таку, як права власності на товар, його описання, час зберігання, можливі місця призначення та інше [2].

Таким чином, під час організації складського приміщення у вигляді типізованих стійок та введення в дію інтелектуального модуля прийняття рішення розташування продукції на складських площах згідно критеріальних характеристик отримана можливість реалізації автоматичного складу з автономними завантажувальними пристроями, що без участі людини здійснюватимуть всі маніпуляції в складських приміщеннях. За рахунок введення запропонованих NFC / RFID міток з'являється можливість надійної безконтактної ідентифікації товару з переліком всієї додаткової інформації без необхідності введення додаткових каналів зв'язку з центральними базами даних, що також може спотворити інформацію. Також, на кожне місце для зберігання товару можна додатково встановити свої мітки, що дасть змогу автозавантажувачам верифікувати свої дії під час здійснення завантаження [2].

При наявності автоматичних складських завантажувачів для обслуговування складських приміщень достатньо буде лише однієї людини, яка контролюватиме

ведення облікової інформації та встановлюватиме для кожного товарного надходження додаткові вагові коефіцієнти чи інші мітки.

ВИСНОВКИ

Виконані дослідження показали можливість підвищення ефективності розташування в складських приміщеннях з поділом по зонах інтенсивності запитів та інших критеріальних показників. У роботі проведений аналіз роботи складських систем, підходи до зонування складських приміщень. Запропоновано спрощену модель автоматизованої системи прийняття рішення розташування продукції на складі, що може бути застосований для модернізації існуючих WMS систем, чи лягти в основу створення нових. Підібрано повний комплект апаратного забезпечення для реалізації бездротових засобів отримання інформації про продукцію, яка зберігається з розширеним переліком даних про неї. Запропоновано критерій оптимізації під час завантаження складських приміщень. Результати дослідження можуть бути застосовані для проведення автоматизації роботи складських приміщень в будь-якій галузі промисловості.

ЛІТЕРАТУРА:

1. Usachëva M. M. OBOSNOVANYE NEOBKHOODYMOSTY VNEDRENYIA SYSTEMY UPRAVLENYIA SKLADOM / M. M. Usachëva, N. P. Karpova. // Vestnyk molodykh uchenykh SHËU. – 2017. – №1. – S. 169 – 171.
2. Diduk V. A. Informatsiina model diialnosti avtomatychnoho skladu nespuchykh materialiv / V. A. Diduk, T. V. Bodnenko. // Naukovi pratsi : nauk. zhurn. / Chornom. nats. un-t im. Petra Mohyly. Serii: Kompiuterni tekhnolohii;. – 2017. – S. 34–44.
3. Optymyzatsiia razmeshcheniia tovara na sklade [Elektronnyi resurs]. – 2010. – Rezhym dostupu do resursu: <https://works.doklad.ru/view/bsRwBjBE-c4.html>.
4. Kyzub A. Praktykum / Staty AVTOMATYZATSYIA SKLADA: UCHETNYIE SYSTEMY I WMS — EST LY RAZNYTSA? [Elektronnyi resurs] / A. Kyzub // 2012 – Rezhym dostupu do resursu: <https://customsexpert.ru/articles/avtomatizatsiya-sklada-uch.htm>.
5. Dëmyñ V. Optymyzatsiia tekhnolohycheskoho protsessu na sklade (Chast 2) / V. Dëmyñ. // Sklad y Tekhnyka. – 2005. – №10. – S. 25 – 38.
6. Effektivnoe yspolzovanye skladskykh ploshchadei [Elektronnyi resurs] // ABM Cloud. – 2018. – Rezhym dostupu do resursu: <http://abmcloud.com/effektivnoe-ispolzovanie-skladskih-ploshchadei/>.
7. Optymyzatsiia razmeshcheniia tovara na sklade farmdystrybiutora [Elektronnyi resurs] // Lobanov-Logist.ru. – 2017. – Rezhym dostupu do resursu: https://www.lobanov-logist.ru/library/all_articles/54238/.
8. Karpova N.P., Shekhovtsov A.A. Avtomatyzatsiia upravleniia skladskym tekhnolohycheskym protsessom // Sovremennyye nauchnyie yssledovaniia. Выр.
9. M24LR64E-R [Elektronnyi resurs] // STMicroelectronics. – 2014. – Rezhym dostupu do resursu: <http://www.st.com/st-web-ui/static/active/en/resource/technical/document/datasheet/DM00047008.pdf>
10. M24LR16E-R [Elektronnyi resurs] // STMicroelectronics. – 2013. – Rezhym dostupu do resursu: <http://www.st.com/st-web-ui/static/active/en/resource/technical/document/datasheet/DM00031737.pdf>.

*Рецензент: д.т.н., проф. В.Ф. Коваленко
Херсонський національний технічний університет*

СЦЕНАРНО-ПРОЦЕДУРНАЯ МОДЕЛЬ ФУНКЦИОНИРОВАНИЯ ВИРТУАЛЬНЫХ ТРЕНАЖЕРОВ ДЛЯ ПОДГОТОВКИ МЕНЕДЖЕРОВ

УДК 004.946.

ДОРОВСКАЯ Ирина Александровна

к.т.н., доцент кафедры информатики и социально-гуманитарных дисциплин
Криворожского филиала Европейского университета.

Научные интересы: Информационные технологии мониторинга условий труда рабочих мест, разработка инновационного программного обеспечения для улучшения процесса производства, изучение теории принятия решений,
e-mail: irina.dora07@gmail.com

ДОРОВСКОЙ Дмитрий Владимирович

к.т.н., доцент кафедры информатики и социально-гуманитарных дисциплин
Криворожского филиала Европейского университета.

Научные интересы: Интернет, маркетинг, информационные и маркетинговые технологии,
мониторинг и диагностика горно-металлургического оборудования.
e-mail: postmaster@krivrig.e-u.in.ua

ВВЕДЕНИЕ

Последние тренды развития информационно-компьютерных технологий (ИКТ) показывают, что интерес к развитию наукоемких программных решений, включая облачные технологии, технологии умного города и др., имеет тенденцию к росту, в отличие от интереса к развитию аппаратных ресурсов. Тем самым методическая и технологическая база на уровне программного обеспечения, которая может быть использована для решения задач компьютерного моделирования (КМ), быстро расширяется и развивается, и в

краткосрочной перспективе замедления в данных процессах маловероятны.

Важность направления КМ в перспективе и повсеместность применения соответствующих технологий подтверждается тысячами статей, появляющимися по соответствующему поисковому запросу в [1]). Поэтому необходимость в разработке и реализации инструментария для обучения специалистов с использованием ВТ в области компьютерного моделирования не вызывает сомнений.

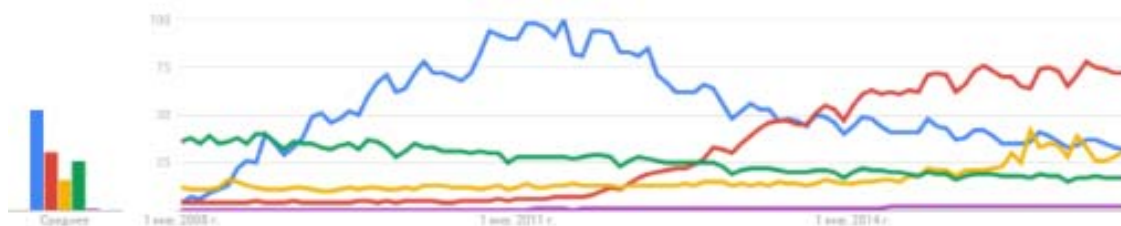


Рис. 1 — Динамика популярности запросов Cloud Computing (синий), Big Data (красный), Smart City (темно-желтый), НРС (зеленый), Hybrid Cloud (фиолетовый)

Анализ литературных данных и постановка проблемы. В таких условиях подготовка менеджеров, включающих различных специалистов, связана не только с отработкой профессиональных навыков их работы, но и с достижением главной цели – навыков работы в команде, обеспечивающих максимальную прибыль и безаварийность. При этом возникает проблема, связанная с управлением процессами тренинга и оценивания профессиональной деятельности команды менеджеров.

Целью работы является разработка и исследование моделей и методов создания и управления виртуальным тренажером, которые позволят повысить качество профессиональной подготовки менеджеров. Для достижения этой цели в работе были поставлены и решены следующие задачи:

1. Анализ опыта подготовки специалистов менеджеров на основе виртуальных тренажеров,
2. Разработка моделей для создания и управления ВТ, позволяющими реализовать индивидуальное и командное обучение различных специалистов менеджеров.

МЕТОДЫ И МАТЕРИАЛЫ ИССЛЕДОВАНИЙ

Методы исследования. В работе использованы методы теории множеств, теории графов, теории искусственного интеллекта, а также технологии объектно-ориентированного программирования.

Объектом исследования являются процессы автоматизации подготовки (переподготовки) специалистов менеджеров в учебно-тренажерных центрах.

Предметом исследования являются модели и методы управления индивидуальным и командным обучением специалистов менеджеров с использованием виртуальных тренажеров.

Материалы исследований. В соответствии с общей концепцией и конкретными требованиями экспертами предметной области формируются логические описания модели ВТ в терминах онтологии, состоящие из трех основных частей – объектов сцены, действий и сценариев. Онтология определяет возможную структуру модели ВТ, связи между компонентами и ограничения целостности. Объекты сцены предназначены для описания структуры каждого

объекта, входящего в виртуальный мир сцены, связей между объектами. Каждый объект обладает набором атрибутов, определяющих присущие ему роль, положение в пространстве, поведение и отображение в зависимости от различных условий. Каждый атрибут имеет имя, критерий обязательности и изменяемости, множество значений и тип. Атрибуты могут задавать как характеристики объекта в целом, так и характеристики состояний объекта. Набор атрибутов объекта зависит от класса объекта. Объекты делятся на следующие основные классы, образующие между собой иерархию: простой объект, изменяемый объект, составной объект, таблица (массив). Описание действий необходимо для детализации возможных вариантов взаимодействия пользователя с объектами виртуальной среды. Действия могут быть командные и интерактивные. В зависимости от назначения их описание включает различные параметры. Они могут как изменять объекты виртуальной среды ВТ, так и возвращать пользователю информационный результат. Результат действия может зависеть от атрибутов объектов и от возможных оценок для различных наборов атрибутов. Результаты действий могут быть простые (конкретные значения) и сложные (функционально получаемые значения). Сценарий описывает последовательность действий, которую должен выполнить менеджер в процессе работы с обучающим ВТ. Он может содержать переходы различных типов, ветвления, циклы, метки. С одной стороны, сценарий представляет собой граф, в узлах которого находятся действия, а дуги являются переходами от одних действий к другим. С другой стороны, сценарий – это последовательность некоторых логических этапов, где каждый этап представляет собой объединенное в группу множество узлов.

Реализация ВТ выполнена реализация обучающего тренажера, включающего обучающие задания по классическим методам исследования в образовании в соответствии с технологией разработки, описанной выше.

РЕЗУЛЬТАТЫ ИССЛЕДОВАНИЙ

Структура сценарно-процедурная модели учебного материала.

Введем конечное множество $E = \{\ell_1, \ell_2, \dots, \ell_i, \dots, \ell_n\}$, где ℓ_i – обучающий блок (ℓ_i – блок) соответствует порции учебного мате-

риала и пару отношений на E, которые являются отображениями.

1) $\alpha(\ell_1, \ell_2, \dots, \ell_n)$ - отношение непосредственной связности по информации (выводимости) блока ℓ из блоков $\{\ell_1, \ell_2, \dots, \ell_n\}$.

2) $\beta(\ell_1, \ell_2, \dots, \ell_n)$ - отношение детализации знания ℓ , которое "состоит из" знаний $E = \{\ell_1, \ell_2, \dots, \ell_i, \dots, \ell_n\}$.

Определение 1. Структурной моделью учебного материала называется тройка (E, α, β) , где E множество учебных блоков, а - отношение информационной связности, b - отношение детализации.

Свойства отношений информационной связности таковы, что ℓ -блоки образуют сцепления, которые имеют начальные блоки $\{\ell_0\}$ и конечные (целевые) блоки $\{\ell_k\}$. Образно говоря, знания имеют источники $\{\ell_0\}$, промежуточные (выводимые) знания и конечные (целевые) $\{\ell_k\}$, связанную сеть передачи потоков знаний от источников к целевым обучающим блокам, поэтому модель знаний $\langle E, \alpha, \beta \rangle$ называется потоковой структурой знаний (Knowledge Flow Structure - KFS).

Граф знаний и обучающий кластер. Обучающие ℓ -блоки связываются в сеть KN (Knowledge Net) следующим образом. Каждой вершине KN сопоставляется единственный ℓ -блок. Каждой дуге KN соотносится маркер, который является кодом формулы (описания) соответствующего знания, заключенного в учебный блок. Далее маркеры обозначаются большими буквами латинского алфавита.

Исходя из того, что ℓ -блок определяет отображение, введем понятие формулы вывода.

Определение 1. Формулой вывода называется выражение вида

$$\ell(A_1, A_2, \dots, A_i, \dots, A_n) \rightarrow B \quad (1)$$

где $(A_1, A_2, \dots, A_i, \dots, A_n)$ - входные, поставляемые в блок ℓ (исходные) знания, а B - выходные (целевые) знания, полученные в результате процедуры обучения (вывода), "R" обозначает некоммутативную операцию "следует".

Каждый ℓ -блок имеет единственный выход и поэтому именем блока может служить маркер исходящей от него дуги. Все исходящие из ℓ -блока дуги имеют одинаковый маркер. Формула (1) читается так: знание B является следствием процесса научения из знаний $A_1 \& A_2 \& \dots \& A_i \& \dots \& A_n$.

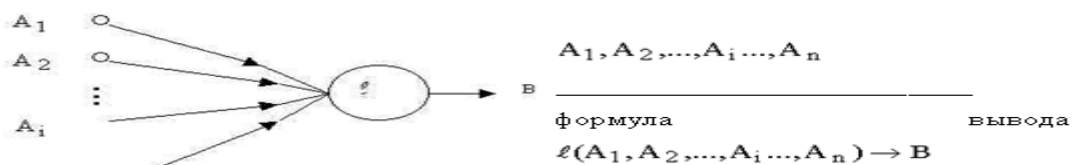


Рис. 2 - Логический блок обучения и его формула вывода

Определение 2. Граф KN есть конечный граф для отношения $\alpha \subset \{E \times E\}$, где $E = \{\ell_1, \dots, \ell_k\}$.

ℓ -блоки, и $\alpha = \langle \ell_i, \ell_j \rangle$ - есть дуга связи с маркером A_{ℓ_i} (принадлежащем блоку ℓ_i , из которого она исходит). Граф KN обладает следующими постулируемыми свойствами:

- асимметричностью
- ациклическостью

Граф KN конечен, имеет множество входных вершин (типа ℓ_0) и единственную выходную вершину (типа ℓ_k). Вывод целевых знаний реализуется системой формул вывода вида (2.1) для каждой из его вершин, исключая входные. Граф показывает из каких

составляющих и как складывается целевое знание. Он закладывает основу методики построения учебного материала, диалектического единства группировки и выделения, обобщения и дифференциации знаний. Для построения графа KN выполняются следующий алгоритм:

1. Отбор есть определение множества $E = \{\ell_1, \ell_2, \dots, \ell_k\}$ логических порций обучения, имеющих законченный смысловой характер, они названы ℓ -блоками.

2. Группировка знаний около ℓ -блока в виде логических формул вывода $\ell(A_1, A_n) \rightarrow B$, где "B" есть "сумма" знаний, выведенная из составляющих знаний " A_1, \dots, A_n ".

3. Связывание ℓ -блоков в логические обучающие кластеры при помощи подстановок знаний в системе формул.

$$(\ell_{i_1}, \ell_{2i}, \dots, \ell_{ni}) \rightarrow \ell_i \approx (\ell_{i_1} \rightarrow \ell_i), (\ell_{2i} \rightarrow \ell_i), \dots, (\ell_{ni} \rightarrow \ell_i) \quad (2)$$

Формула (2) получается из логической формулы $\ell_i(A_{i_1}, A_{2i}, \dots, A_{ni}) \rightarrow B_i$. Стрелки "R" в правой части формулы (2) соответствуют дугам графа KN и помечены маркерами знаний.

Предложение 1. Система формул типа (2) для графа KN определяет подграф, являющийся деревом, который обладает свойством полной выводимости, и, поэтому, является кластером.

$$\begin{aligned} (\ell_2, \ell_3, \ell_4) \ell_{k1} &\approx (\overset{B_2}{\ell_2 \rightarrow \ell_{k1}}), (\overset{B_3}{\ell_3 \rightarrow \ell_{k1}}), (\overset{B_4}{\ell_4 \rightarrow \ell_{k1}}); \\ (\ell_{02}, \ell_4) \ell_2 &\approx (\overset{A_2}{\ell_{02} \rightarrow \ell_2}), (\overset{B_4}{\ell_4 \rightarrow \ell_2}); \\ (\ell_4, \ell_5) \ell_3 &\approx (\overset{B_4}{\ell_4 \rightarrow \ell_3}), (\overset{B_5}{\ell_5 \rightarrow \ell_3}); \\ (\ell_{02}, \ell_{03}) \ell_4 &\approx (\overset{A_2}{\ell_{02} \rightarrow \ell_4}), (\overset{A_3}{\ell_{03} \rightarrow \ell_4}); \end{aligned} \quad (3)$$

На графе KN логически выделяются завершённые подмножества - кластеры.

Определение 3. Обучающим кластером называется направленный граф KN, вершины которого размечены ℓ -блоками, дуги B - маркерами знаний, каждой вершине соотнесена формула вывода и каждая вершина (ℓ -блок) кластера выводима из начальных знаний, либо является начальным знанием (типа ℓ_0).

Таким образом, кластер обладает свойством полной выводимости. Свойство кластерной или полной выводимости является необходимым свойством активного электронного учебника. При отсутствии полной выводимости нельзя построить процесс контроля знаний и управление процессом обучения. Процесс вывода определяется деревом вывода, оно строится по логическим формулам.

Определение 4. Непосредственная окрестность логического блока ℓ_i задается формулой

Доказательство конструктивно и следует из свойств выражения (2) и свойств кластерности графа KN. На рис.3 представлено дерево вывода для графа KN. Дерево отражает свойство полной выводимости каждого ℓ -блока в кластере обучения. Система формул, определяющая дерево вывода в граф KN на рис.3а приведена ниже.

$$(\ell_4, \ell_5)\ell_3 \approx (\overset{B_4}{\ell_4 \rightarrow \ell_3}, \overset{B_5}{\ell_5 \rightarrow \ell_3});$$

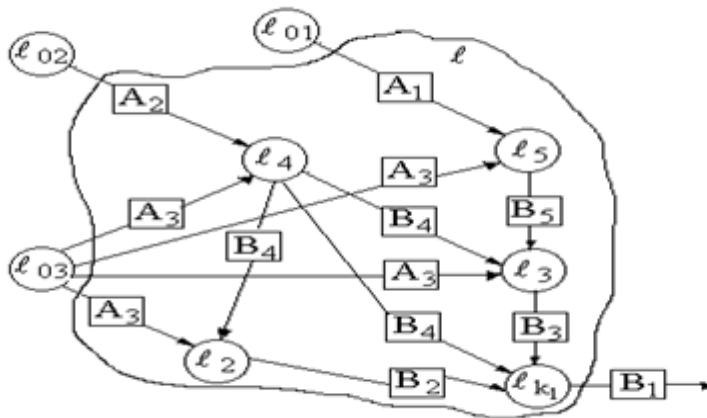
Порождение графа знаний в процессе детализации знаний. Отношение детализации знаний в постулируется как отношение разбиения на составляющие ℓ -

блоки. Составляющие ℓ -блоки получаются в результате операции разбиения b , которая задается выражением

$$\beta(\ell_i^j) = (\ell_{1i}^{j+1}, \ell_{2i}^{j+1}, \dots, \ell_{ni}^{j+1}, \dots, \ell_{ki}^{j+1}) \quad (4)$$

где ℓ_i^j - ℓ -блок, подлежащий разбиению на детали-ные (составляющие) блоки - $\ell_{1i}^{j+1}, \ell_{2i}^{j+1}, \dots, \ell_{ki}^{j+1}$

а) Граф КН



Формулы вывода

- для ℓ -блока
- $\ell_{k1}(B_2, B_3, B_4) \rightarrow B_1$
 - $\ell_2(A_3, B_4) \rightarrow B_2$
 - $\ell_3(A_3, B_4, B_5) \rightarrow B_3$
 - $\ell_4(A_2, A_3) \rightarrow B_4$
 - $\ell_5(A_1, A_3) \rightarrow B_5$

б) Дерево вывода кластера

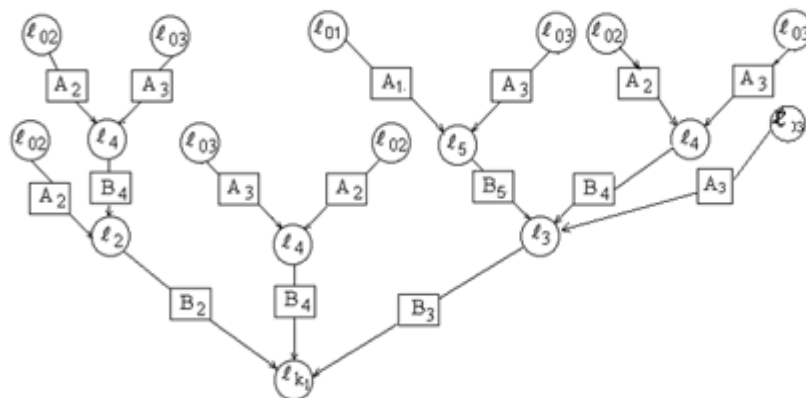


Рис.3. Граф КН и дерево вывода для обучающего кластера.

Понятно, что разбиение (детализацию) можно продолжать как угодно глубоко, применяя рекурсивно

операцию b ко вновь полученным блокам. При этом учитывается следующее свойство: если $\ell_i^j \in \ell_k^{j-1}$,

то $\ell_i^j \notin \ell_k^{j-1}$ $k \neq \ell$ для постулированного отношения разбиения. Таким образом, блок ℓ_i^j , входящий в блок ℓ_k^{j-1} , не может входить в другие блоки. Введем формальную процедуру детализации: 1) начальный слой детализации называется нулевым, если на нем находится единственный блок ℓ^0 ;

2) если имеется блок ℓ_i^j на j -м слое детализации, то составляющие его блоки $(\ell_{i1}^{j+1}, \dots, \ell_{ik}^{j+1})$, полученные операцией разбиения b , считаются находящимися на $j+1$ -м слое детализации.

Процедура детализации по своей природе неоднозначна, т.е. отражает логику построения учебного материала тем или иным преподавателем, или даже одним и тем же преподавателем, но для различных контингентов обучающихся. Число слоев детализации вообще не ограничено и никак не связано с психологической сложностью слишком "мелкой" детализации. ℓ -блоки на самых нижних слоях детализации (с самым большим индексом "j") могут быть элементами конспекта учебника. *Предложение 2.* При разложении, соблюдающем кластерность на каждом j -ом слое, порожденный (терминальный) граф KN будет также кластером.

Ярусно-параллельная форма представления графа знаний (Knowledge Net)

До сих пор модель знаний была ориентирована на логику связи отдельных ℓ -блоков. Оказывается важным при построении учебного материала учитывать и логическую независимость (несвязность) знаний. Независимость ℓ -блоков в графе Knowledge Net (KN) позволяет строить различные варианты последовательностей изложения учебного материала и выби-

рать из них наилучшие с точки зрения тьютора и менеджера.

Определение 6. Ярусно-параллельной формой (ЯПФ) графа KN называется частичное упорядочение вершин по уровням, на которых расположены независимые по логическим связям ℓ -блоки так, что на 0-м уровне расположены входные знания, а на последнем целевое знание. На рис.4 показаны две различные ЯПФ для графа KN, который состоит из двух ℓ -блоков ℓ_1 и ℓ_2 , которые являются кластерами и соответственно их объединение тоже является кластером. Как видно из рисунков, ЯПФ состоит из 8 уровней (ярусов), на 0-м уровне входные знания (A_1, A_2, A_3), на последнем, 7-м уровне - целевое знание. На каждом уровне расположены независимые знания. ЯПФ (рис.4а) и ЯПФ (рис.4б) отличаются друг от друга различным расположением независимых вершин по уровням.

Можно себе образно представить, что по ЯПФ идет фронт обучения, сначала изучаются входные знания, затем знания 1-го уровня и т.д., до целевых знаний на последнем уровне, причем последовательность изложения знаний на каждом из уровней произвольна. Совокупность независимых знаний на каждом уровне ЯПФ названа логическим уровнем. Таким образом, фронт обучения пробегает последовательность логических уровней обучения.

Все связи в ЯПФ разбиты на два класса:

1. непосредственные связи, которые "передают" знания с предыдущего уровня на последующий;
2. отложенные связи, указывающие на полученные ранее знания, которые студент должен помнить (или ему должны напоминать), пока эти знания будут использоваться при прохождении фронта обучения.

Для фиксации этих двух типов связей в ЯПФ предусмотрены соответствующие поля (рис.4 в).

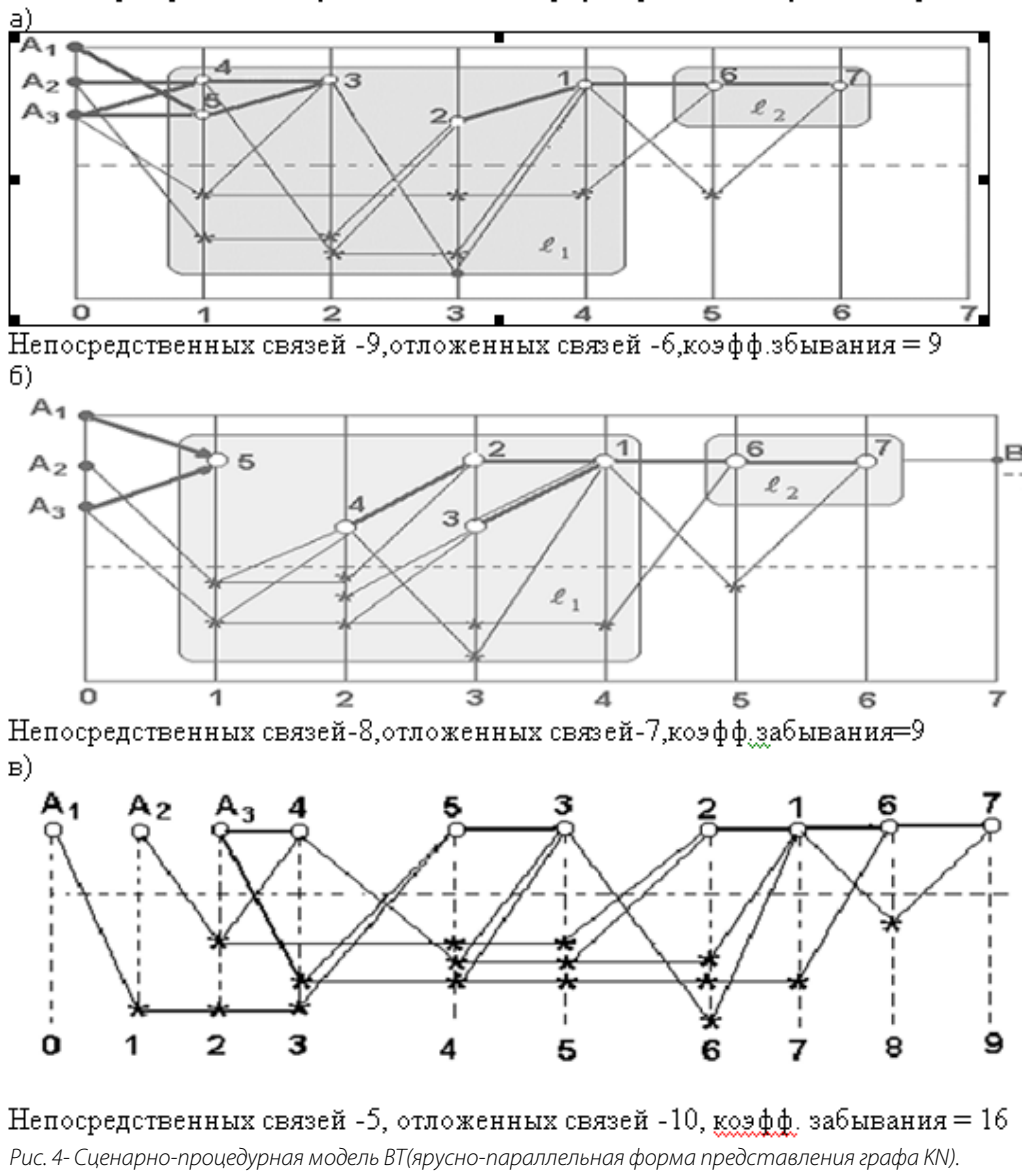


Рис. 4- Сценарно-процедурная модель ВТ(ярусно-параллельная форма представления графа KN).

ВЫВОДЫ ПО РЕЗУЛЬТАТАМ ИССЛЕДОВАНИЙ

1. Виртуальные тренажерные комплексы предназначены для решения широкого круга задач обучения в различных областях, невозможно разработать универсальное программное обеспечение, удовлетворяющее любым технологическим требованиям. В связи с этим представляется целесообразным строить тренажерный комплекс по модульному принципу. 2. Исходя из функций тренажера, он должен состоять из следующих основных модулей: модуль управления базой данных параметров технологического объекта модуль управления базой математических моделей, описывающих

поведение объекта модули реализации математических моделей элементов объекта (оборудования и средств управления им); модуль управления базой видеоклипов и элементов отображения информации о моделируемом объекте; модуль управления базой аварийных ситуаций; модуль конструктора виртуального тренажерного комплекса для конкретной задачи обучения ; модуль управления базой данных менеджера и анализа статистической информации; модуль диспетчера, реализующий управление всем виртуальным комплексом в процессе обучения.



СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ:

1. Лисицына Л.С. Теория и практика компетентностного обучения и аттестаций на основе сетевых информационных систем. СПб: СПбГУ ИТМО, 2006. 147 с.
2. L. Lisitsyna and A. Lyamin Approach to Development of Effective E-Learning Courses// SMART DIGITAL FUTURES 2014. IOS Press, 2014. Vol. 262, p. 732-738.
3. Васильев В.Н., Лисицына Л.С., Лямин А.В. Методический интернет-центр. СПб: Питер, 2005. 96 с.
4. Lisitsyna L.S., Efimchik E.A. An Approach to Development of Practical Exercises of MOOCs based on Standard Design Forms and Technologies // Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering. 2017. Vol. 180. pp. 28-35.
5. Lisitsyna L.S., Efimchik E.A. Making MOOCs more effective and adaptive on the basis of SAT and game mechanics // Smart Education and e-Learning 2017, Springer, 2018, Vol. 75, pp. 56-66.
6. Lisitsyna L.S., Efimchik E.A., Izgareva S.A. RLCP-compatible virtual laboratories with 3D-models and demonstration mode: development and application in e-learning // Smart Education and e-Learning 2017, Springer, 2018, Vol. 75, pp. 75-81.

Рецензент: *д.т.н., проф. В.А. Доровской*
Европейский университет Криворожский филиал

ВИКОРИСТАННЯ ЛЮДИНО-МАШИНОГО ІНТЕРФЕЙСУ ДЛЯ УПРАВЛІННЯ ТЕХНОЛОГІЧНИМИ ПРОЦЕСАМИ

УДК 519.6

ДИМОВ Володимир Степанович

К.т.н., доцент кафедри Інформаційних технологій

Наукові інтереси: методи дослідження складних динамічних систем, оптимізація в телекомунікаційних системах, нейронні мережі.

e-mail: vdymov@rambler.ru

БОСКІН Олег Осипович

ст. викладач кафедри Інформаційних технологій

Наукові інтереси: методи дослідження складних динамічних систем, оптимізація в телекомунікаційних системах, нейронні мережі.

e-mail: bbbosss@i.ua

ВСТУП

Нейрокомп'ютерний інтерфейс (НКИ) (званий також прямий нейронний інтерфейс, мозковий інтерфейс, інтерфейс «мозок - комп'ютер» [1]) - система, створена для обміну інформацією між мозком і електронним пристроєм (наприклад, комп'ютером). У односпрямованих інтерфейсах зовнішні пристрої можуть або приймати сигнали від мозку, або посилати йому сигнали. Двонаправлені інтерфейси дозволяють мозку і зовнішнім пристроїв обмінюватися інформацією в обох напрямках. В основі нейро-комп'ютерного інтерфейсу, часто використовується метод біологічного зворотного зв'язку.

Будь-який рух, сприйняття або внутрішня розумова діяльність пов'язані з певним паттерном активації нейронів, які взаємодіють один з одним за допомогою електричних імпульсів [2]. Ці струми створюють електромагнітне поле, яке можна зареєструвати зовні голови за допомогою методів електроенцефалографії (ЕЕГ) і магнітоенцефалографії (МЕГ).

Метод ЕЕГ, розроблений Гансом Бергером в 1929 році, протягом багатьох років успішно використовується для трьох цілей:

- діагностики неврологічних розладів в клініках і шпиталях;
- для дослідження функцій мозку в нейрофізіологічних лабораторіях;
- для терапевтичних цілей на основі біологічного зворотного зв'язку.

ПОСТАНОВКА ЗАДАЧІ

У зв'язку з розвитком технологій з'явилася можливість створення компактних пристроїв для зчитування даних з людського мозку. Надалі нейроінтерфейси можуть повністю зайняти сферу комунікації, витіснивши при цьому мобільні телефони, смартфони, планшети та ін. Вони надають можливість збільшити швидкість роботи з даними, кількість оброблюваних задач в одиницю часу, швидкість реакції на критичну ситуацію для операторів різних систем, тому пропонується модель нейрокомп'ютерного інтерфейсу, який може бути використаний у багатьох сферах життєдіяльності таких, як:



побут, медицина, контроль виробництва, управління вантажною технікою, у військовій сфері та ін.

ОСНОВНА ЧАСТИНА

Людство завжди прагне постійно, підтримувати між собою зв'язок, що призводить до створення все нових засобів комунікацій.

Комунікації в сучасному суспільстві розглядаються як синонім поняття спілкування (лат. Communication – роблю загальним, пов'язую, спілкуюся) [3], основним завданням засобів комунікацій є передача інформації від людини до людини, як якась форма взаємодії.

З появою нової технології, людство намагається вивчити її специфіку, проблеми впливу на суспільне життя, соціокультурних і повсякденність. Як говорив Маршал Маклюен «Не знаючи дії засобів комунікацій, неможливо зрозуміти громадські та культурні зміни ... Засоби комунікацій, змінюючи середовище, викликають в нас незвичайні співвідношення чуттєвого сприйняття. Розширення будь-якого почуття змінює образ нашого мислення і діяльності нашого сприйняття світу. Коли змінюються ці співвідношення, змінюються і люди» [4].

Поява персональних комп'ютерів суттєво поліпшило зв'язок між людьми, скоротило витрати часу на передачу інформації. Якщо раніше для передачі інформації потрібно значно-більший час, то зараз це можна зробити за лічені секунди.

З усіх головних досягнень у розвитку засобів масової комунікації є Internet. Internet на сьогоднішній день належить до п'ятого етапу технологічного розвитку людства. Попередні чотири етапи – це обробка енергії води, парова енергія, електрику і електроніка.

Internet – це продукт сучасного суспільства, що складається із сукупності мереж. У його специфіку входить надання величезної кількості інформації, через засоби комунікацій, таких як електронна пошта, регіональні, глобальні та локальні мережі зв'язку. За допомогою Internet надається доступ до численних об'ємів інформації, які зберігаються в різних куточках нашої планети. Як говорить Мануель Кастельс «Internet виконує роль технологічного базису для організаційної різновиди інформаційної ери – Мережі».

Розвиток Internet поділяється на три стадії:

- перша стадія, коли людина зверталася до Інтернету лише, для того, щоб знайти якусь цікаву для нього інформацію;
- стадія комерції та комунікації;
- стадія середовища спілкування.

Можливості Internet дуже великі. За допомогою Internet можна управляти персональними фінансами, купити товар, що цікавить, знайти нових друзів, замовити квитки на літак, дізнатися останні новини.

Завдяки еволюції засобів зв'язку доступ в Internet здійснюється зараз не тільки з персонального комп'ютера, але також з мобільного телефону, комунікатора, кишенькового комп'ютера. Маючи при собі мобільні телефони, підключені до Internet, людина забезпечує собі не тільки канал зв'язку, а й більш оперативний доступ до отримання різноманітної інформації в найкоротший термін з будь-якого куточка нашого міста, країни, світу, що говорить про збільшення користувачів мережі Internet. Уже сьогодні в світі використовується понад 600 млн. мобільних телефонів підтримують технологію HTML. Ця мова, поступово набуває статусу універсальної мови обробки інформації.

Існують думки з приводу комунікативної природи Internet. Хтось вважає, що Інтернет проявляє себе як засіб масової інформації, так як здатний поширювати інформацію на необмежене коло об'єктів одержувачів. Інші ж вважають, що Internet складається з безлічі web сторінок, кожна з яких має свою аудиторію, а отже Internet є засобом групової комунікації.

Сучасні діячі наук відзначають, що Internet на сьогоднішній день, як комунікаційний медіум, який є засобом спілкування між мільйонами людей, які не виявляють якісного результату, тим, що допомагає людям підтримувати зв'язок між собою. На їхню думку, між спілкуванням людей в такому глобальному масштабі, не відбивається якісна зміна, а якщо відбивається, то тільки у окремих груп людей.

Сучасний етап розвитку цивілізації характеризується міжнародним масштабом застосування мереж телекомунікацій і формуванням глобальної інфраструктури інформаційних мереж. Вони забезпечують можливість передачі різних видів інформації та дозволяють жителям будь-якої країни долучитися до досягнень науки, культури і виробництва всього світу. При вирішенні цих

задач велику роль відіграє застосування нових технологій, перш за все цифровий і супутникового зв'язку, завдяки яким можуть бути створені універсальні мережі зв'язку, що охоплюють практично весь світ без технічних обмежень.

Очевидно, що в основі brain-computer interface (BCI) має лежати розпізнавання патернів біопотенціалів мозку. Якщо людина, що проходить тести, може змінювати характер своїх біопотенціалів, наприклад, виконуючи певні розумові задачі, то система BCI могла б транслювати ці зміни в контрольні коди, наприклад, по переміщенню курсора миші на екрані комп'ютера або руки робота-маніпулятора. Також ці коди можна використовувати для вибору букв на «віртуальній клавіатурі» або для контролю інвалідного візка.

У 1988 році Фарвел і Дончин (Farwell 1988) вперше реалізували систему «віртуальної клавіатури», яка дала змогу друкувати текст, розпізнаючи компонент P300 при зніманні зорових викликаних потенціалів (ВП). Після цього було розроблено стільки різних модифікацій BCI систем зі зростаючими можливостями, які вже знайшли своє застосування як в клініці для спілкування з пацієнтами, що повністю втратили можливість руху (Birbaumer 1999), так і інноваційні технологічні проекти з дистанційного управління роботами (Millán 2004).

Швидкість передачі інформації цього нового комунікаційного каналу поки невелика. Однак постійний прогрес в області техніки реєстрації ЕЕГ, алгоритмів обробки сигналів і розпізнавання патернів, більш глибоке розуміння нейрофізіології і залучення все більшої кількості вчених в ці роботи забезпечують неухильне збільшення цієї швидкості, зростання числа додатків і динамічний прогрес всього напрямку в цілому.

Якщо в 1994 році було всього 6 дослідних груп, що займалися BCI, то на перший міжнародний з'їзд по BCI в 1999 році приїхали дослідники з 2-х десятків лабораторій. На другому з'їзді в 2012 році були дослідники, які представляли 38 дослідницьких груп, включаючи США, Німеччину, Китай, Фінляндію, Швейцарію, Англію, Канаду та ін. Черговий з'їзд відбудеться в 2018 році.

Найбільше визнання одержало визначення BCI, дає в роботі Уолпал і інших (Wolpaw, 2002):

BCI - це комунікаційна система, в якій повідомлення або команди, що посилаються індивідуумом у зовніш-

ній світ, не проходять через звичайні нормальні вихідні канали мозку у вигляді периферійних нервів і м'язів.

Згідно з цим визначенням моргання не можуть бути використані BCI системою. Використовуються або біопотенціали мозку, зареєстровані з поверхні скальпа - ЕЕГ, або з поверхні кори - електрокортикограми (ЕКОГ), або внутрішньомозкові відведення. Останні два способи відносяться до інвазивної BCI.

BCI - це інтерфейс між людиною і комп'ютером, який отримує команди безпосередньо від мозку без вчинення будь-якого фізичного руху (Levine 2002). BCI використовує електрофізіологічні сигнали для управління зовнішніми пристроями (Bayliss 2001)

До складу BCI системи входять:

- Електроди для відведення біопотенціалів. Мінімальна кількість - 2, частіше записи роблять за допомогою 32, 64 і навіть 128 каналів. При великій кількості електродів використовують електродні шоломи для швидкості установки і збільшення точності позиціонування електродів над визначеними полями мозку, а також відтворюваності їх розташування від експерименту до експерименту.

- Підсилювач біопотенціалів, що підключається до комп'ютера або безпосередньо (наприклад, через USB порт), або через інтерфейсну A / D карту (аналогоцифровий перетворювач).

- Персональний комп'ютер для реєстрації сигналів і їх обробки. Так як у багатьох системах використовується елементи biofeedback, то або цей же комп'ютер, або додатковий ПК показує людині, що тестується, стимули і результати розпізнавання, наприклад, текст, що вводиться.

- Програмне забезпечення для реєстрації та обробки ЕЕГ, розпізнавання патернів і пред'явлення стимулів і результатів розпізнавання.

У електроенцефалографії використовують металеві електроди з хлорсеребряного покриттям. Для забезпечення електричного контакту електрода з шкірою використовують або електропровідний гель, або марлю, просочену фізіологічним розчином.

Схема розташування електродів на поверхні голови називається монтаж (рис. 1). У клінічній та науковій електроенцефалографії стандартом є схема «10-20%»,

яка була введена в 1950-х роках канадським нейрофізіологом Генрі Джаспером.

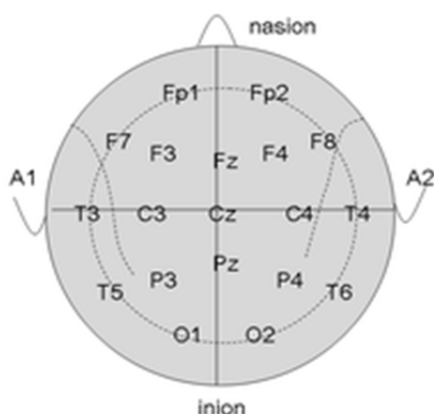


Рис. 1 Розташування електродів на поверхні голови за схемою «10-20%»

Для визначення місць накладення електродів через верхівку (Vertex) проводяться два умовних меридіани - перший від перенісся (Nasion) до потиличного бугра (Inion), другий між зовнішніми слуховими проходами. Через ці точки прокладають умовний меридіан, який ділиться на відрізки по 10 і 20% загальної довжини. Поперечні меридіани відкладаються по осі, яка проходить між зовнішніми слуховими проходами через верхівку. Електроди розміщуються в місцях перетину умовних ліній. Електроди, які розміщуються на лівій стороні голови, мають непарні індекси; на правій стороні - парні; електроди, розміщені на вертексних лініях, мають індекс z. Чим менше індекс електрода, тим ближче він розташований до основних меридіанів. Позначення електродів: F (Frontalis) - лобові; T (Temporalis) - скроневі; C (Centralis) - центральні; P (Parientalis) - тім'яні; O (Occipitalis) - потиличні; A (Auricularis) - вушні. Кількість накладених електродів залежить від конкретної мети дослідження. У разі необхідності схему «10-20%» можна розширити шляхом проведення додаткових меридіанів між основними. Стандартизація схеми накладення електродів дозволяє дослідникам і лікарям зіставляти результати, отримані в різний час в різних лабораторіях. Для реєстрації ЕЕГ необхідна наявність двох електродів, між якими і буде вимірюватися різниця електричних процесів. Пара електродів, між якими

реєструється різниця потенціалів, називається відведенням.

Існують дві категорії відводів: монополярні і біполярні. При монополярному відводі один з кожної пари електродів розміщується над певною ділянкою мозку, а другий - на певному віддаленні від мозку. Перший з цих електродів називається активним або робочим, а другий - пасивним або референтним. Найбільш часто використовують об'єднаний вушний референт. При біполярному відведенні обидва електроди розташовані над мозком, а тому в такому відводі буде реєструватися різниця потенціалів цих двох областей. У сучасній електроенцефалографії більш поширеним є монополярний запис ЕЕГ, оскільки він дозволяє легко перейти до біполярного запису, математично перерахувавши реєстровані сигнали.

Електричний сигнал, який відводиться зі скальпа людини, що тестується, має досить низьку амплітуду (10⁻⁴ - 10⁻⁶ В), а тому для реєстрації сигнал повинен бути посиленним. Для цього використовуються підсилювачі змінного струму. Сучасні ЕЕГ-комплекси реалізовані на базі персональних комп'ютерів і дозволяють одночасно здійснювати запис сигналу і відображати його на моніторі в режимі on-line. Для того, щоб електроенцефалографічний сигнал міг оброблятися комп'ютером, його необхідно перевести з аналогової форми до цифрової. Для цього проводиться періодичне вимірювання його амплітуди і передача результату до комп'ютера (оцифровка сигналу). Зареєстрований ЕЕГ-сигнал може зберігатися в комп'ютері і підлягати обробці за допомогою численних математичних методів.

При проведенні ЕЕГ вимірюють сумарні постсинаптичні струми. Потенціал дії (ПД, короткочасна зміна потенціалу) в передсинаптичній мембрані аксону викликає вивільнення нейромедіатора в синаптичну щілину [3]. Нейромедіатор, або нейротрансмітер, - хімічна речовина, здійснює передачу нервових імпульсів через синапси між нейронами. Пройшовши через синаптичну щілину, нейромедіатор зв'язується з рецепторами постсинаптичної мембрани. Це викликає іонні струми в постсинаптичній мембрані. В результаті у позаклітинному просторі виникають компенсаторні струми. Саме ці позаклітинні струми формують потенціали ЕЕГ. ЕЕГ нечутлива до ПД аксонів.

Хоча за формування сигналу ЕЕГ відповідальні пост-синаптичні потенціали, поверхнева ЕЕГ не здатна зафіксувати активність одного дендрита або нейрона. Вірніше сказати, що поверхнева ЕЕГ являє собою суму синхронної активності сотень нейронів, що мають однакову орієнтацію в просторі, розташованих радіально до шкіри голови. Струми, що спрямовані по дотичній до шкіри голови, не реєструються. Таким чином, під час ЕЕГ реєструється активність радіально розташованих в корі апікальних дендритів. Оскільки вольтаж поля зменшується пропорційно відстані до його джерела в четвертій степені, активність нейронів в глибоких шарах

мозку зафіксувати набагато важче, ніж струми в безпосередній близькості шкіри.

Струми, які зареєстровані на ЕЕГ, характеризуються різними частотами, просторовим розподілом і взаємозв'язком з різними станами мозку (наприклад, сон або неспання). Такі коливання потенціалу представляють собою синхронізовану активність цілої мережі нейронів. Ідентифіковано тільки деякі нейронні мережі, що відповідають за реєстровані осциляції (наприклад, таламокортикальний резонанс, що лежить в основі «сонних веретен» - прискорених альфаритмів під час сну), тоді як багато інших (наприклад, система, що формує потиличний основний ритм) поки не встановлені.

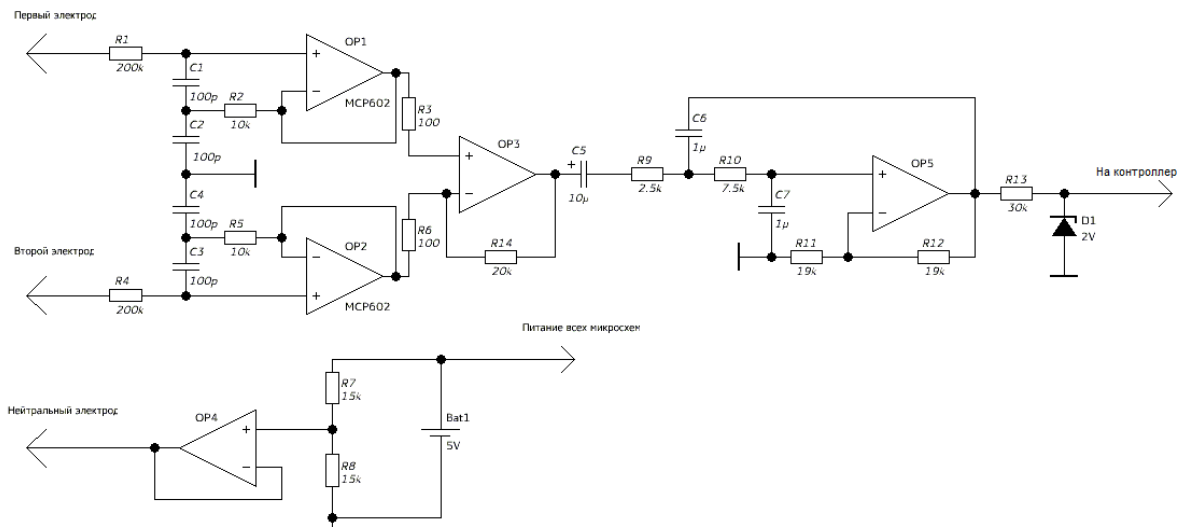


Рис. 2 Функціональна схема підключення вологих електродів

Для отримання традиційного поверхневого ЕЕГ запис роблять за допомогою електродів, що містяться на шкірі волосистої частини голови із застосуванням електропровідного гелю або мазі. Зазвичай перед приміщенням електродів по можливості видаляють омертвілі клітини шкіри, які підвищують опір [4]. Методику можливо вдосконалити, використовуючи вуглецеві нанотрубки, які проникають у верхні шари шкіри і сприяють поліпшенню електричного контакту. Така система датчиків називається ENOBIO; проте представлена методика в загальній практиці (ні в наукових дослідженнях, ні тим більше в клініці) поки не використовується. Звичайно в багатьох системах використовуються електроди, кожен з яких має окремий провід. У деяких системах використовуються спеціальні шапочки

або сітчасті конструкції у вигляді шолома, в яких укладені електроди; найчастіше такий підхід виправдовує себе, коли використовується комплект з великою кількістю щільно розташованих електродів.

Для більшості варіантів застосування в клініці і в дослідницьких цілях (за винятком наборів з великою кількістю електродів) розташування і назва електродів визначена Міжнародною «10-20%» системою. Використання даної системи гарантує, що назви електродів між різними лабораторіями строго узгоджені. У клініці найчастіше використовується набір з 19 відвідних (плюс заземлення та електрод порівняння). Для реєстрації ЕЕГ немовлят зазвичай використовується менша кількість електродів. Щоб отримати ЕЕГ конкретної області мозку з більш високим просторовим дозволом, можна вико-



ристовувати додаткові електроди. Набір з великою кількістю електродів (зазвичай у вигляді шапочки або шолома-сітки) може містити до 256 електродів, розташованих на голові на більш-менш однакової відстані одна від одної.

Кожен електрод з'єднаний з одним входом диференціального підсилювача (тобто один підсилювач доводиться парі електродів) в стандартній системі електрод порівняння з'єднаний з іншим входом кожного диференціального підсилювача. Такий підсилювач збільшує потенціал між вимірювальним електродом і електродом порівняння (зазвичай в 1,000-100,000 раз, або коефіцієнт посилення напруги становить 60-100 дБ). У разі аналогової ЕЕГ сигнал потім проходить через фільтр. На виході сигнал реєструється. Для клінічної поверхневої ЕЕГ частота аналого-цифрового перетворення відбувається при 256-512 Гц; частота перетворення до 10 кГц використовується в наукових цілях.

Для отримання компактною переносною конструкції, яку потрібно змащувати гелем, в ній необхідно виконати попереднє посилення сигналу за допомогою диференціального підсилювача, виконаного прямо на датчику [4]. Це також дозволить відфільтрувати шуми і збільшить якість сигналу, що зчитується.

ЛІТЕРАТУРА:

1. Interfeys mozok-komp'yuter [Elektronnyy resurs] – Rezhym dostupu do resursu: <http://neurobotics.ru/research/bci>.
2. Farwell L. A., Donchin E. // Talking off the top of your head: A mental prosthesis utilizing event-related brain potentials. *Electroencephalography and clinical Neurophysiology* / Farwell L. A., Donchin E., 1988. – S. 70, 510–523.
3. Elektroentsefalohrafiya [Elektronnyy resurs] – Rezhym dostupu do resursu: <http://cnsinfo.ru/encyclopaedia/diagnostics/eeg/>.
4. Dry active electrodes [Elektronnyy resurs] – Rezhym dostupu do resursu: http://radek.superhost.pl/active_electrodes/index.php?action=theory.
5. Shcherbo V.K. Standarty po lokal'nykh vychyslytel'nykh setyam: Spravochnyk / V.K. Shcherbo, V.M. Kyreychev, S.Y. Samoilenko. – M.: Radyo y svyaz', 1997 – 253s.
6. Ynformatsyonnoe obshchestvo: kontseptsyy y ystorycheskaya praktyka. / Voprosy kul'turolohyi. – 2007. – №4.
7. Bol'shoy éntsyklopedychesky slovar' – M, 2003.
8. Maklyuén M. Sredstvo samo est' soderzhanye. YO / Marshal Maklyuén. – M: Yzdatel'stvo, 2004. – 507 s.
9. Maksymov N. V. Komp'yuternye sety / N. V. Maksymov, Y. Y. Popov. – M: Forum, 2007. – 448 s.
10. Broydo V.L. Vychyslytel'nye systemy, sety y telekommunikatsyy. / V.L. Broydo – M.: Dyaloh-Myfy, 1995 – 320 s.
11. Y.D. Cheplyk, D.M. Kartolapov, V.S. Dymov. Neyroynterfeysnyy manipulyator // Materialy V Vseukrayins'koyi naukovy-praktychnoyi konferentsiyi studentiv, aspirantiv ta molodykh vchenykh z avtomatychnoho upravlinnya KHNTU. – Kherson: KHNTU, 2017. – S. 288-293.
12. V.S. Dymov, I.D. Cheplik, D.M. Kartolapov Rozrobka datchykyv dlya realizatsiyi neyrointerfeysnoho manipulyatora // Vymiryval'na ta obchyslyval'na tekhnika v tekhnolohichnykh protsesakh: Materialy XVII mizhnar. nauk.-tekhn. konferentsiyi (8-13 chervnya 2017 r., m.Odesa); Odes. nats. akad. zv'yazku im. O.S.Popova. – Odesa – Khmel'nyts'ky: KHNU, 2017.- S. 130-132.

ОСНОВНІ РЕЗУЛЬТАТИ ТА ВИСНОВКИ

Розглянуто нейрокомп'ютерних інтерфейс, його перспективи та можливості використання. Однією з головних особливостей нейрокомп'ютерного інтерфейсу є можливість його застосування практично в будь-якій сфері життєдіяльності людини. На даний момент це один найпрогресивніших напрямків в даній сфері, що і стало причиною цього дослідження. Основною метою є побудова нейрокомп'ютерного інтерфейсу.

У наш час нейроінтерфейс є важливим придбанням для суспільства. Маючи високий рівень розвитку наше суспільство все ще має проблему з інвалідами, але її можна виправити. Він дозволяє людям зі здоровим розумом, але хворим тілом відчувати і більш того, діяти, нарівні зі здоровими людьми. Його використання полегшує протезування і створення екзоскелета, так як вже буде присутньою керуюча частина системи з відомими протоколами передачі даних. Також дані обладнання дадуть можливість постійно стежити за здоров'ям людського мозку, і при необхідності вчасно звернутися за медичною допомогою.

*Рецензент: д.т.н., проф. Коваленко В.Ф.,
Херсонський національний технічний університет*

SYNTHESIS AND STUDY OF THE MATHEMATICAL MODEL OF A CATERPILLAR MOBILE ROBOT FOR VERTICAL MOVEMENT

UDC 681.5

Oleksandr Serhiiovych GERASIN

lecturer, Admiral Makarov National University of Shipbuilding, Ukraine

Research interests: computer control systems, modeling, fuzzy logic, intelligent robotics, automation.

e-mail: oleksandr.gerasin@nuos.edu.ua

Oleksiy Valeriiovych KOZLOV

Ph.D., associate professor, Admiral Makarov National University of Shipbuilding, Ukraine

Research interests: automation, intelligent control systems, robotics, fuzzy logic, elements and devices of computing systems.

e-mail: oleksiy.kozlov@nuos.edu.ua

Galyna Volodymyrivna KONDRATENKO

Ph.D., associate professor, Admiral Makarov National University of Shipbuilding, Ukraine

Research interests: automation, intelligent decision support systems, fuzzy logic.

e-mail: galvlad09@rambler.ru

Huang MINGXIN

lecturer, Yancheng Polytechnic College, China

Research interests: robotics, automation, computer control systems, modeling.

Introduction and problem statement. Mobile robots (MRs) and robotic systems are becoming more and more widespread in various industries: from transportation and assembly operations to high-tech processing [1-5]. For a successful creation of the systems, the stage of their development commonly implements the means and approaches of mathematical and simulation modeling. In such a way, major mass-size, economic and environmental parameters are optimized, compliance of the control system with the given quality indicators is verified, and so on. This approach allows specifying the structure and parameters of control devices of the robot control system in due time (that is before the development of an experimental sample), as well as checking its main dimensions and the feasibility of choosing particular structural elements (driving wheels, motors, caterpillar tracks, etc.). In addition, it enables pre-estimating the time of the robot's response (change in control coordinates) to the setting action for the

tasks of automatic control, which is crucial for ensuring a high accuracy of the MR's technological operations [5].

Under present-day conditions, of great interest is the possibility of realizing one or more technological operations on inclined and vertical surfaces with the help of MRs [5]. The robots with such a function are complex multicomponent technical objects; their operation requires special computer-aided monitoring and control systems [2]. An important phase in the development of such systems is mathematical and simulation modeling of the future designs [6]. In order to test the algorithms controlling the MR's spatial motion along inclined surfaces and study the effectiveness of the MR's control system, it makes sense to develop models of the whole MR and its individual components taking into account their design features and employing mathematical and computer simulation methods, which are quite effective and cheap in comparison with experimental and other approaches [6].

Latest research and publications analysis. Publications [7-8] present and elaborate on the basic methods for mathematical modeling of control objects, which are used for the analysis and synthesis of an automatic control system. A mathematical description of the main types of mobile robots (wheeled, tracked and walking) is given in [9-16]. Development of mathematical models and transfer functions for DC and AC motors, gear units, driving wheels and other elements for the tasks of automated control is considered separately, for instance, in papers [17-19]. Besides, there is a certain number of publications concerning the synthesis of mathematical models of wheeled [12], tracked [13, 14] and walking [15-16] mobile robots. Meanwhile, the problem of development of the mathematical model of a wheeled MR moving vertically along inclined ferromagnetic surfaces, which would be suitable for further synthesis of its automatic control system, is still unresolved. Analysis of the physical properties and technical characteristics of the MR as an object of control of a spatial position with respective control coordinates [20] proves the feasibility of development of its mathematical model with the use of the principles and algorithms based on the theory of vehicle movement and phased inclusion of the loads on the robot's drive motors work generated by the components of forces and moments of resistance [19].

The article aim is to develop a mathematical model of a vertically moving MR based on the theory of vehicle movement as well as to study its behavior in terms of the impact of the MR's individual components on its spatial position and the rate of response to various specified input actions and disturbances with account for the angle of inclination of the working surface and direction of the MR's movement.

Basic material. Synthesis of the mathematical model of a vertically moving MR. The principal scheme and basic properties of a vertically moving MR capable of performing specified technological operations on large inclined ferromagnetic surfaces are discussed in publications [20, 21]. The MR has a solid frame, two caterpillar tracks and two drive DC motors (DCM), one for each track. The whole structure of the MR holds onto the inclined or vertical ferromagnetic surface with the help of two separate flat permanent magnets, which are hinged to the frame, providing orientation of the resultant vector of the clamping force with regard to the ferromagnetic surface

[15, 21, 22]. The MR is moving as the left-hand and right-hand tracks are rotated by the driving wheels. In turn, the driving wheels are actuated by the drive DCMs, which are connected to the wheels through gear units.

The process of modeling of the MR requires a thorough consideration of its main units and components [10, 11]. Let us render the mathematical models of the main elements of the MR below.

Drive motor model. The required rotational speed of the driving wheels is ensured by direct control of the MR's drive motors. In the design shown in [21], such are DCM, although they can be substituted with AC motors as well. Let us describe the DCM model.

An important property of a separately excited DCM is that the resulting moment of the forces from all the armature conductors [17-18] (dubbed as the electromagnetic moment of the motor M_{EM}) is proportional to the current of the armature I_A supplied to the motor from the power source:

$$M_{EM} = c_M I_A, \quad (1)$$

where c_M is the coefficient of proportionality. According to the laws of electromagnetic induction in a conductor moving in a magnetic field there arises an electromotive force (EMF). The total EMF of the armature coils E is applied through the commutator and brushes to the external terminals of the motor. At the motor operation the EMF is directed against the external voltage U_A supplied to the armature from the power source. The EMF is directly proportional to the angular speed of the motor's shaft rotation ω_E :

$$E = c_E \omega_E, \quad (2)$$

where c_E is the coefficient of proportionality of the motor's constant EMF. Due to the nature of electromagnetic phenomena in the DC motor, the coefficients c_E and c_M are to have approximately equal numerical values if the SI system is used.

In the motor's armature circuit the current I_A flows under the impact of the DC voltage U_A of the power source and the counter-EMF of the motor. This circuit is characterized by the active resistance R_A and the inductance L_A of the

armature winding. Having the moment of inertia J_A , the rotating rotor is driven by the simultaneous action of the electromagnetic moment of the motor M_{EM} and the moment of resistance to external forces M_{SM} applied to the motor shaft.

The output differential equations for the DCM are drawn up on the basis of the laws of physics. The Kirchhoff's second law is applied for the electric circuit to compose the following equation:

$$U_A - E = R_A I_A + L_A \frac{dI_A}{dt}, \quad (3)$$

where $R_A I_A$ characterizes the voltage drop on the active resistance of the armature circuit in accordance with the Ohm's law, and $L_A(dI_A/dt)$ indicates the self-induced EMF that occurs in the winding when changing the armature current. This equation does not take into account the voltage drop on the brushes, which depends on the armature current nonlinearly but commonly has a relatively small value compared to the voltage U_A .

The differential equation describing the processes in the mechanical part of the motor is composed using the Newton's second law:

$$J_A \frac{d\omega_E}{dt} = M_{EM} - M_{SM}. \quad (4)$$

This equation ignores the impact of frictional forces that emerge during the rotor's rotation but have a relatively weak effect on the DCM shaft acceleration.

Using the above formulas and reducing the differential equations to the Cauchy normal form, we obtain the following description of the DCM:

$$\begin{cases} \frac{dI_A}{dt} = \frac{U_A - R_A I_A - c_E \omega_E}{L_A}; \\ \frac{d\omega_E}{dt} = \frac{c_M I_A - M_{SM}}{J_A}. \end{cases} \quad (5)$$

When studying the processes with the help of a personal computer (PC), it is convenient to use a structural representation of the DCM's mathematical model. For that end, let us subject the previously obtained system of linear

differential equations to Laplace transformation under zero initial conditions. The resulting system of algebraic equations is as follows:

$$\begin{cases} pI_A(p) = \frac{U_A(p) - R_A I_A(p) - c_E \omega_E(p)}{L_A}; \\ p\omega_E(p) = \frac{c_M I_A(p) - M_{SM}(p)}{J_A}, \end{cases} \quad (6)$$

where p is the Laplace operator, while the quantities $I_A(p)$, $\omega_E(p)$, $U_A(p)$, $M_{SM}(p)$ are Laplace representations of the variables I_A , ω_E , U_A and M_{SM} , respectively. After equivalent transformations, these equations acquire the following form:

$$I_A(p) = \frac{U_A(p) - c_E \omega_E(p)}{R_A (T_E p + 1)}; \quad (7)$$

$$\omega_E(p) = \frac{c_M I_A(p) - M_{SM}(p)}{J_A p}, \quad (8)$$

where $T_E = L_A/R_A$ is the electromagnetic time constant of the motor's armature circuit.

Gear unit model. If the operating device (OD) of the system is directly connected to the motor's shaft, analysis of the motion of the electromechanical system "motor – operating device" can utilize equation (4). Such a kinematic diagram is typical for fans, pumps and a number of other machines [17]. However, in our case, the operating device of the system (driving wheel) is connected to the motor's shaft through the gear system, namely, through the gear unit. Therefore, an immediate use of equation (4) is impossible, since the moments M_{EM} and M_{SM} are applied to different shafts, and inertial masses rotate at different speeds.

To enable the use of the equation of motion, one has to solve the problem of reducing all the moments of resistance and inertia of individual kinematic links to one shaft, usually to the shaft of the electric motor. This reduction is only a calculating operation and does not require any physical changes in the system. The principle of moment reduction is to preserve the equivalence of power. The reduction of the moments of inertia follows the principle of kinetic energy conservation [17].

As a rule, if the machine's OD is connected to the motor's shaft via a gear unit (Fig. 1, a) with the gear ratio k_G , reducing the actual moment of resistance M_{SM} applied to the driving wheel to the motor shaft requires compliance with the equivalence of power [17]:

$$M_{MSM}\omega_W = M_{SM}\omega_E, \quad (9)$$

where M_{MSM} is the moment of resistance to motion (hereinafter referred to as its static moment); ω_W is the angular speed of the driving wheel.

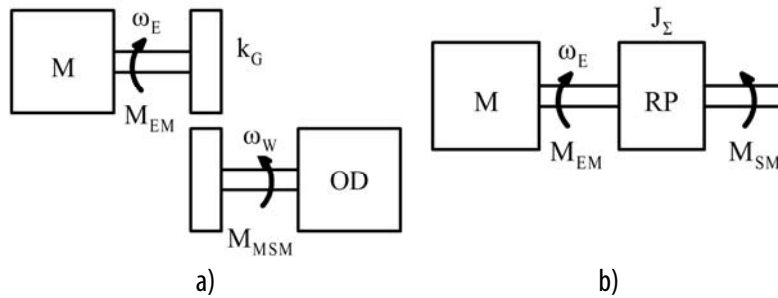


Fig. 1. Reduction of the moments of resistance to the shaft of the motor (M): a – actual kinematic diagram of the motor's connection to the operating device (OD) through the gear unit; b – computational diagram with the reduced part (RP)

Therefore, if one knows the static moment on the shaft of the operating device, one can calculate the static moment applied to the motor's shaft using the following formula (not accounting for the losses in the gear unit):

$$M_{SM} = \frac{M_{MSM}}{\omega_E / \omega_W} = \frac{M_{MSM}}{k_G}. \quad (10)$$

The general rule here is that in order to reduce a static moment to the motor's shaft, one divides the actual static moment on the OD's shaft by the gear ratio. If it is necessary to take into account the losses in the gear unit, the coefficient of transmission efficiency should also be included to the denominator (10).

Thus, the first transfer function of the gear unit takes the following form:

$$W_{G1}(p) = \frac{\omega_E(p)}{\omega_W(p)} = k_G. \quad (11)$$

To reduce the moment of inertia of the driving wheel J_W to the motor's shaft, it is necessary to preserve the equality of kinetic energies

$$\frac{J_W\omega_W^2}{2} = \frac{J_{RM}\omega_E^2}{2}, \quad (12)$$

where J_{RM} is the moment of inertia reduced to the motor's shaft.

Consequently, the OD's moment of inertia to the motor's shaft is calculated according to the formula:

$$J_{RM} = \frac{J_W}{k_G^2}. \quad (13)$$

The general rule here is that in order to reduce the moment of inertia to the motor's shaft, the actual moment of inertia of the kinematic link is to be divided by the squared gear ratio.

Hence, the second transfer function of the gear unit is as follows:

$$W_{G2}(p) = \frac{J_W(p)}{J_{RM}(p)} = k_p^2. \quad (14)$$

As a result of reducing the static moment and the moment of inertia to the motor's shaft, the actual kinematic diagram transforms into the computational one (Fig. 1, b), which enables using the modified equation of motion of the electric drive:

$$M_{EM} - M_{SM} = J_\Sigma \frac{d\omega_E}{dt} = (J_A + J_{RM}) \frac{d\omega_E}{dt}. \quad (15)$$

MR model. Development of MRs' models which would fit various designs usually considers its main operating device [17]. In our case, the main operating device of the

MR is its driving wheels (gears) rotating the caterpillar tracks, which leads to the movement of the entire MR along the inclined surface (Fig. 2, a).

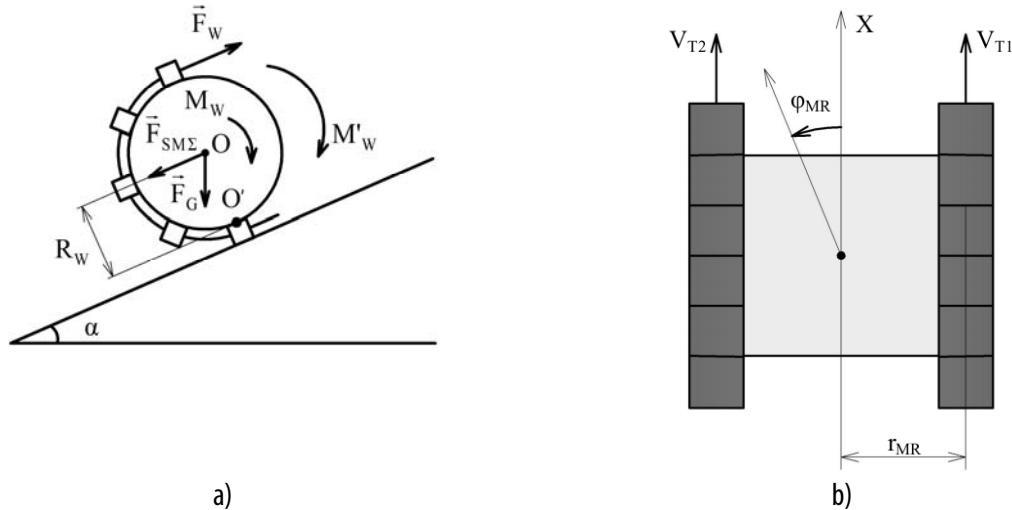


Fig. 2. Schematic design of the MR: the driving wheel with a part of the caterpillar track at its movement along an inclined surface (a) and a generalized diagram of the tracked MR movement (b)

Let us compose the equation of moments on the DCM's shaft:

$$M_{EM} = M_{DE} + M_{MSE} + M_{WE}, \quad (16)$$

where M_{DE} is the dynamic moment of the DCM, $M_{DE} = J_{\Sigma E} d\omega_E/dt$, with $J_{\Sigma E}$ being the total moment of inertia of the motor; M_{WE} is the motor's operating moment (torque) transmitted to the gear unit; M_{MSE} is the moment of friction forces in the motor's bearings, $M_{MSE} = k_{MSE}\omega_E$, with k_{MSE} being the coefficient of friction in the motor's bearings.

The equation of moments on the wheel at its rotational motion can be drawn up as follows:

$$M_W = M_{DW} + M_{MSW} + M_W, \quad (17)$$

where M_W is the total mechanical moment on the driving wheel, $M_W = M_{WE}k_G$, with k_G being the gear ratio; M_{DW} is the dynamic moment of the wheel, $M_{DW} = J_{\Sigma W} d\omega_W/dt$, with $J_{\Sigma W}$ being the total moment of inertia of the wheel and caterpillar track and ω_W being the angular speed of the wheel's rotation; M_W is the wheel's operating moment (torque) transmitted to the caterpillar track; M_{MSW} is the moment of friction forces in the wheel's bearings,

$M_{MSW} = k_{MSW}\omega_W$, with k_{MSW} being the coefficient of friction in the wheel's bearings.

Let us consider the rotational-translational motion of the MR's driving wheel (Fig. 2). The mechanical moment on the operating wheel M_W can be described as

$$M_W = F_W R_w, \quad (18)$$

where R_w is the radius of the operating wheel; F_w is the force that is applied at the top of the wheel to rotate it around the point O' of its contact with the working surface (Fig. 2, a), $F_w = M_W/R_w$.

Fig. 2, a also allows determining the operating moment of the wheel M'_w created by the force F_w relative to the point O' :

$$M'_w = F_w \cdot 2R_w = 2M_w. \quad (19)$$

Let us compose an equation of moments on the wheel at its progressive rotation:

$$M'_w = M_{SME}, \quad (20)$$

where $M_{SM\Sigma}$ is the total mechanical moment of the MR's motion resistance force, $M_{SM\Sigma} = F_{SM\Sigma}R_w$, with $F_{SM\Sigma}$ being the MR's motion resistance force.

The latter ($F_{SM\Sigma}$) is calculated as follows for each caterpillar track:

$$F_{SM\Sigma} = F_{SMG} + \frac{F_{DMR}}{2} + \frac{F_{SMMR}}{2} + \frac{F_L}{2}, \quad (21)$$

where F_{SMG} is the resistance force generated by the force by gravity, F_{DMR} is the dynamic resistance force; F_{SMMR} is the MR's rotation resistance force; F_L is the force of the loading arising when a technological operation is performed, $F_L = \text{var}$.

Next, let us compose the expressions for each component of the MR's motion resistance force:

$$F_{SMG} = \frac{F_G}{2} \sin \alpha \cos \varphi_{MR} = \frac{m_{MR}g}{2} \sin \alpha \cos \varphi_{MR}, \quad (22)$$

$$F_{DMR} = m_{MR} \frac{dV_{MR}}{dt}, \quad (23)$$

$$F_{SMMR} = \frac{M_{SMMR}}{r_{MR}}, \quad (24)$$

where m_{MR} is the mass of the MR; M_{SMMR} is the moment of the MR's resistance to rotation; r_{MR} is the arm of the force F_{SMMR} relative to the MR's center; V_{MR} is the MR's linear speed; α is the angle of the inclined surface; φ_{MR} is the route (course) of the MR (it is assumed that at $\varphi_{MR} = 0$, the robot is moving upward along the inclined plane); g is the acceleration of gravity.

In turn, the moment of the MR's resistance to rotation M_{SMMR} can be expressed as follows:

$$M_{SMMR} = M_{DR} + M_{FR}, \quad (25)$$

$$M_{EM1} = J_{\Sigma E} \frac{d\omega_{E1}}{dt} + k_{MSE} \omega_{E1} + \frac{1}{k_G} \left(J_{\Sigma W} \frac{d\omega_{W1}}{dt} + k_{MSW} \omega_{W1} + \frac{R_w}{4} \left(m_{MR} (g \sin \alpha \cos \varphi_{MR} + \frac{dV_{MR}}{dt}) + F_L + (J_{MR} \frac{d\omega_{MR}}{dt} + k_{FR} \omega_{MR}) \frac{1}{r_{MR}} \right) \right). \quad (34)$$

The equation for the second track has the same form:

$$M_{EM2} = J_{\Sigma E} \frac{d\omega_{E2}}{dt} + k_{MSE} \omega_{E2} + \frac{1}{k_G} \left(J_{\Sigma W} \frac{d\omega_{W2}}{dt} + k_{MSW} \omega_{W2} + \frac{R_w}{4} \left(m_{MR} (g \sin \alpha \cos \varphi_{MR} + \frac{dV_{MR}}{dt}) + F_L + (J_{MR} \frac{d\omega_{MR}}{dt} + k_{FR} \omega_{MR}) \frac{1}{r_{MR}} \right) \right). \quad (35)$$

where M_{DR} is the dynamic moment of resistance to rotation; M_{FR} is the moment of friction forces.

Let us consider each component of M_{SMMR} separately:

$$M_{DR} = J_{MR} \frac{d\omega_{MR}}{dt}, \quad (26)$$

$$M_{FR} = k_{FR} \omega_{MR}, \quad (27)$$

where J_{MR} is the moment of inertia of the MR; ω_{MR} is the angular speed of the MR; k_{FR} is the coefficient of friction at rotation.

Let us compile the equation of interrelation of the kinematic parameters of the MR, taking into account [13, 22] and Fig. 2, b):

$$\omega_w = \frac{\omega_E}{k_G}, \quad (28)$$

$$V_{T1} = \omega_{W1} R_w, \quad (29)$$

$$V_{T2} = \omega_{W2} R_w, \quad (30)$$

$$V_{MR} = \frac{V_{T1} + V_{T2}}{2}, \quad (31)$$

$$\omega_{MR} = \frac{V_{T1} - V_{T2}}{r_{MR}}, \quad (32)$$

$$\omega_{MR} = \frac{d\varphi_{MR}}{dt}, \quad (33)$$

where V_{T1} and V_{T2} are the linear speeds of the lefthand and righthand caterpillar tracks of the MR; ω_{W1} and ω_{W2} are the angular speeds of the driving wheels of the MR.

Accordingly, taking into account expressions (16) – (27), let us compose the equation of the mathematical model for the MR's first caterpillar track:

The obtained mathematical dependencies (34)–(35) represent the mathematical model of the MR for moving over inclined surfaces.

Simulation results. Simulation of the obtained equations of the MR's individual elements and the whole MR employs the structural modeling method. Within this framework, the MR (equations (28) – (35)), the DCM (equations (7) and (8)), and the gear unit (equations (11) and (14)) are regarded as individual elements combined into a single reference structural scheme for the simulation of the MR's spatial motion. Table 1 provides a list of the MR's basic parameters which have been set in order to perform the simulation and obtain the initial dependencies of the MR's essential technological parameters on time.

Table 1.

Basic parameters of the MR

Loaded weight, m_{MR} , kg	300
Length, L_{MR} , m	1
Width, S_{MR} , m	0.7
Wheel radius, R_W , m	0.3
Wheel weight, m_W , kg	10
Caterpillar track weight, m_T , kg	30
Height (thickness) of the caterpillar tape, h_T , m	0.015
Linear speed, V_{MR} , m/s	0.3

Calculation of motor parameters for modeling. As noted above, each track is equipped with a motor 2PB132MG; parameters of the DC motors are given in [18]. The quantities indicated there are as follows: P_N is the nominal (rated) power of the DCM, 1.1 kW; n is the nominal rotational speed, 800 rpm; U_N is the nominal supply voltage, 220 V; I_N – is the nominal current of the armature, 7.46 A; R_A is the resistance of the armature winding, 2.44 Ohm; R_{AP} is the resistance of the additional poles winding, 1.53 Ohm; J is the moment of inertia of the armature, 0.038 kg·m²; L_A is the inductance of the armature circuit, 0.055 H. The electric motor's parameters shall be calculated using the approximate methodology described in papers [18, 23].

Taking into account the technical characteristics, the nominal electromagnetic moment of each motor M_{EMN} is calculated as follows:

$$M_{EMN} = \frac{P_N}{\omega_N}, \quad (36)$$

where ω_N is the nominal angular speed of the motor's shaft rotation, $\omega_N = 2\pi n/60$, rad/s.

Hence, the coefficient of proportionality c_M can be calculated from expression (1):

$$c_M = \frac{M_{EMN}}{I_N}. \quad (37)$$

Next, let us compose the equation of the coefficient of proportionality of the constant EMF of the motor using (2) and (3):

$$c_E = \frac{E}{\omega_N} = \frac{U_N - (R_A + R_{AP})I_N}{\omega_N}. \quad (38)$$

The electromagnetic time constant of the motor's armature circuit T_E can be determined as follows:

$$T_E = \frac{L_A}{R_A + R_{AP}}. \quad (39)$$

The gear unit parameters. In order to decrease the load on the motor's shaft and reduce the rotational speed of the driving wheel, each DCM is equipped with an identical gear unit. The main parameters of the latter (1Ts3U – 160) are the following: k_G is the gear ratio, $k_G = 80$; T_2 is the permissible torque (rotating moment) on the output shaft, $T_2 = 1250$ Nm.

Calculation of the moments of inertia for the MR's rotating parts. Specification of the moments of inertia of geometrically complex bodies (the driving wheels, caterpillar tracks and the MR in general) is associated with a high complexity of the required calculation procedures and the need for experimental measurements [18, 23]. Thus, we shall determine these values approximatively, using the formulas for calculating the moments of inertia for geometrically simple bodies, particularly: a disk and a ring with a rectangular cross section.

On that account, the moment of inertia of the driving wheel J_W is calculated as follows:

$$J_W = \frac{m_W R_W^2}{2}, \quad (40)$$

where m_W is the driving wheel's mass, R_W is the driving wheel's radius.

Using the known value of the MR's length L_{MR} , one can estimate the caterpillar track's total length L_T in the following way:

$$L_T = 2(L_{MR} - 2R_W) + 2\pi R_W. \quad (41)$$

Next, let us calculate the moment of inertia for one caterpillar track, J_T :

$$J_T = m_{TR} \left(R_W^2 + \frac{h_T^2}{2} + R_W h_T \right) + m_{TT} \left(\frac{V_T}{\omega_W} \right)^2, \quad (42)$$

where m_{TR} is the mass of the track's rotating part, $m_{TR} = (2\pi R_W / L_T) m_T$; m_{TT} is the mass of the track's linearly moving part, $m_{TT} = m_T - m_{TR}$; m_T is the mass of the whole track; V_T is the linear speed of the track, assuming that $V_T = V_{MR}$; ω_W is the angular speed of the wheel, $\omega_W = V_T / R_W$.

Finally, the total moment of inertia of the wheel and the caterpillar tracks $J_{\Sigma W}$ can be calculated as follows:

$$J_{\Sigma W} = 2J_W + J_T. \quad (43)$$

The moment of inertia of the MR relative to its center axis shall be approximately calculated with the help of formula (44):

$$J_{MR} = \frac{m_{MR} r_{MR}^2}{2}, \quad (44)$$

where r_{MR} is the minimum radius of the MR's rotation (with opposite connection of back-to-back running motors), assuming that $r_{MR} = S_{MR}/2$.

Using the obtained parameters of the MR, a simulation model has been compiled for computer modeling of the transient processes of the MR's essential technological parameters, which are shown in Fig. 3-6 (in this case, the transient process is the system's response to the input stepwise signal by the corresponding parameter).

Fig. 3 clearly indicates how the MR's linear velocity at the supply voltage of 220 V on both actuators depends on the angle of inclination of the working ferromagnetic surface. When the angle increases from 0 to 90°, the resulting speed decreases due to the load increase generated by gravity. When $\alpha = 270^\circ$, the robot moves downwards, respectively, the gravitational force facilitates the MR's movement, thus making its resulting speed higher than in the previous cases (line 4 in Fig. 3).

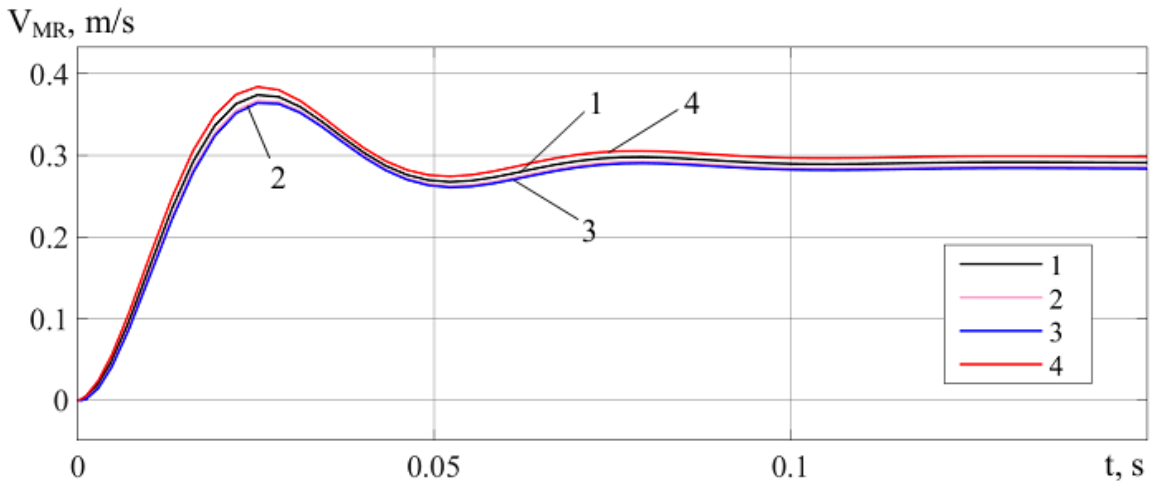


Fig. 3. Change in the MR's linear speed for different values of the angle α :
1 - $\alpha = 0^\circ$; 2 - $\alpha = 45^\circ$; 3 - $\alpha = 90^\circ$; 4 - $\alpha = 270^\circ$

Fig. 4 shows that for $\alpha = 60^\circ$ and DCM supply voltage of 220 V, with introduction of the load (such as that generated by the technological operation of cleaning at the simulation

time of 1 second), the steady-state speed decreases after the transient process and does not return to the previous value.

The robot's rotation to change its course – the φ_{MR} angle – is carried out by changing the supply voltage on each of the drive motors. At that, the linear speed of the whole

MR's movement also changes. The changes in the φ_{MR} and V_{MR} coordinates for different values of DCM supply voltage are shown in Fig. 5 and 6, respectively.

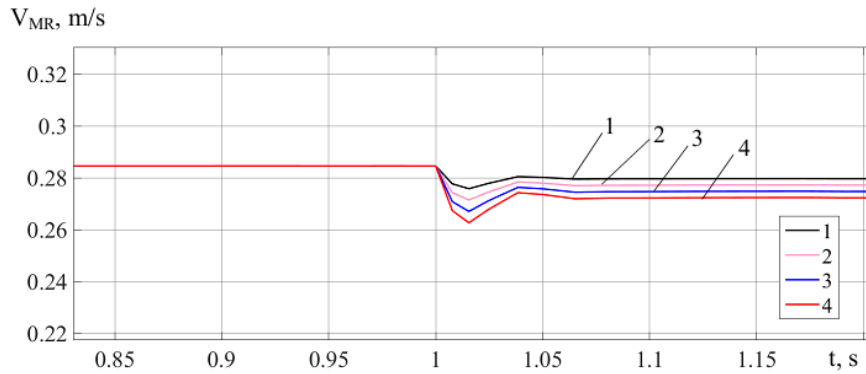


Fig. 4. Change in the linear speed of movement for different values of load: 1 – 2 kN; 2 – 3 kN; 3 – 4 kN; 4 – 5 kN

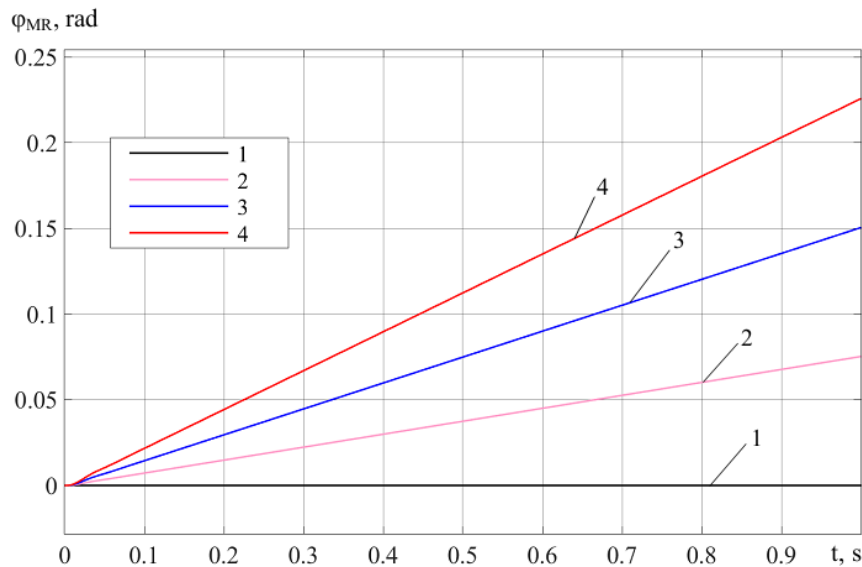


Fig. 5. Change in the MR's angle of rotation at the following modulus difference in the DCM supply voltages ΔU_{12} : 1 – $\Delta U_{12} = 0$ V; 2 – $\Delta U_{12} = 20$ V; 3 – $\Delta U_{12} = 40$ V; 4 – $\Delta U_{12} = 60$ V

Fig. 5 attests to the linearity of the dependencies of the MR's angle of rotation on the difference between the DCM supply voltages for $\alpha = 60^\circ$, simulation time of 0.5 seconds and the load of 2 kN generated by the technological equipment. In this case, the latter does not affect the dependency.

Fig. 6 shows the effect of the difference in the supply voltages of the two DC motors on the MR's linear speed at $\alpha = 60^\circ$, the load of 2 kN, generated by the technological equipment, and the simulation time of 0.5 seconds. Name-

ly, as this difference increases, the MR's speed decreases. Introduction of the load does not alter the nature of the speed's decrease.

Fig. 3-6 prove that the results obtained for the nature of the change of the MR's essential technological parameters under these conditions correspond to the actual ones. In future, it is appropriate to consider the synthesis of the system for monitoring and automatic control of the MR's spatial movement along an inclined working surface.

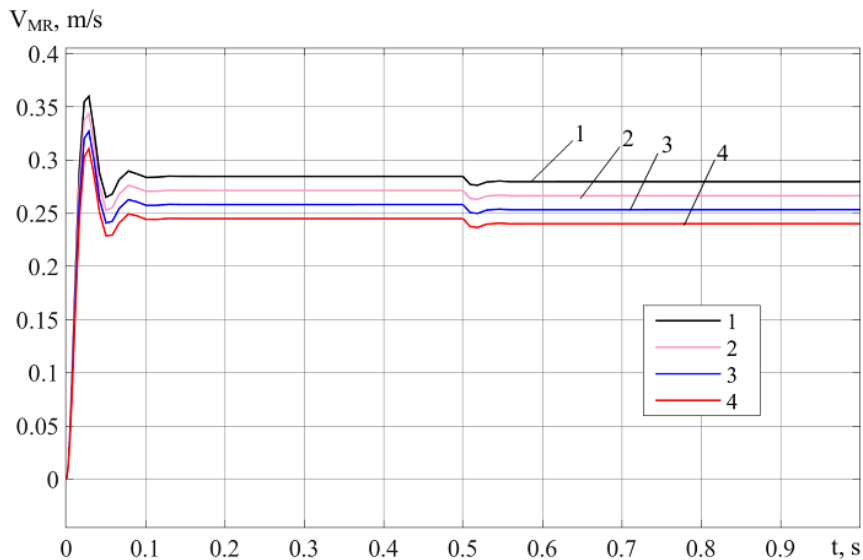


Fig. 6. Change in the MR's linear speed at the following modulus difference in the DCM supply voltages ΔU_{12} :
 1 – $\Delta U_{12} = 0 V$; 2 – $\Delta U_{12} = 20 V$; 3 – $\Delta U_{12} = 40 V$; 4 – $\Delta U_{12} = 60 V$

CONCLUSION

The article presents the procedure of synthesis of the mathematical models of the entire MR and its individual components taking into account the special features of the MR's structural design. Also, the results of MR's model study via computer modeling in terms of the influence of the MR's individual components on its spatial position and the rate of response to particular input actions are shown with account for the angle of inclination of the working surface and the direction of the MR's movement.

The mathematical model has been developed with the application of the mathematical apparatus of the theory of vehicle movement and load reduction to drive motor's shafts. The model enables studying the behavior of the

object of spatial movement control under various production conditions and transient modes, in particular, calculating the MR's speed of movement and angle of rotation at specified and disturbing actions (caused by the load from the working tool). The study results display the major properties of the MR as a complex control object with substantial dependence of MR's spatial motion parameters from surface and acting load features.

The obtained mathematical model can be further applied in the development and study of the effectiveness of a multi-circuit system for monitoring and automatic control of the MR's spatial movement, as well as its software and algorithms operating in real time.

REFERENCES:

1. Tosun O. Mobile robotics: solutions and challenges / O. Tosun, H.L. Akin, M.O. Tokhi, G.S. Virk (ed.). // Proceedings of the Twelfth International Conference on Climbing and Walking Robots and the Support Technologies for Mobile Machines. Istanbul. – 2009.
2. Yurevich Ye. I. Osnovy robototekhniki: Uchebnoe posobiye. – Saint-Petersburg, BKhV-Peterburg, 2005. – 416 s.
3. Siciliano B., Khatib O. (ed.). Springer handbook of robotics / B. Siciliano, O. Khatib (ed.). // Springer. – 2016.
4. Niku S.B. Introduction to robotics: analysis, systems, applications, New Jersey: Publisher Prentice Hall. – 2001.
5. Gerasin O.S. Analiz osoblyvostei mobilnykh robotiv bahatotsilovoho pryznachennia // Naukovi pratsi: Naukovo-metodychnyi zhurnal. Seriya "Compiuterni tekhnologii". – 2014. – vol. 250, no. 238. – S. 25-32.
6. Kustovska O.V. Metodolohiia systemnoho pidkhodu ta naukovykh doslidzhen: Kurs leksii. – Ternopil: Ekonomichna dumka, 2005. – 124 s.
7. Usov A.V., Dubrov A.V., Dmitrishin D.V. Modelirovaniye sistem s raspredelennymi parametrami. – Odessa: Astroprint, 2002. – 664 s.
8. Sharuda V.H. Praktikum z teorii avtomatichnoho upravlinnia. – Dnipropetrovsk: NHU, 2002. – 414 s.
9. Kurfess T.R. (ed.). Robotics and automation handbook. CRC press. – 2004.
10. Spong M.W. Robot modeling and control / M.W. Spong, S. Hutchinson, M. Vidyasagar M. // New York: Publisher Wiley. – 2006.

11. Chernousko F.L. Printsipy dvizheniya i problemy dinamiki mobilnykh robotov // Sbornik materialov Mezhdunarodnoy Vystavki-kongressa Mekhatronika i robototekhnika, 2007. – S. 89-90.
12. Kondratenko Y.P. Synthesis and research of neuro-fuzzy observer of clamping force for mobile robot automatic control system / Y.P. Kondratenko, O.V. Kozlov, O.S. Gerasin, Y.M. Zaporozhets // Proceedings of the 2016 IEEE First International Conference on Data Stream Mining & Processing (DSMP), Lviv, Ukraine, August 23 – 27, 2016. – P. 90-95.
13. Kabanov A.A. Modelirovaniye i upravleniye dvizheniem gusenichnogo mobilnogo robota // Upravleniye v tekhnicheskikh, ergaticheskikh, organizatsionnykh i setevykh sistemakh. – Saint Petersburg, 2012. – S. 740-743.
14. Koloskov S.S. Razrabotka mobilnogo robota i vizualizatsiya ego dvizheniya // Molodezhnyy nauchno-tekhnicheskiiy vestnik. Available at: <http://sntbul.bmstu.ru/doc/602336.html>.
15. Kondratenko Y. Modeling of clamping magnets interaction with ferromagnetic surface for wheel mobile robots / Y. Kondratenko, Y. Zaporozhets, J. Rudolph, O. Gerasin, A. Topalov, O. Kozlov // International Journal of Computing. – 2018. – 17(1). – P. 33-46.
16. Souto D. A Robot for the Unsupervised Grit-Blasting of Ship Hulls / D. Souto, A. Faiña, A. Deibe, F. Lopez-Peña, R.J. Duro // International Journal of Advanced Robotic Systems. – 2012. – vol. 9. – P. 1–16.
17. Onishchenko G.B. et al. Avtomatizirovanny elektroprivod promyshlennykh ustanovok. – Moscow: RASKhN, 2001. – 520 s.
18. Pitsan R.M., Bardachevskiy V.T., Boichuk B.H. Zbirnyk zadach do kursu "Elektropryvid". Persha chastyna "Rozimkneni systemy elektropryvida". – Lviv, Vydavnytstvo Derzhavnoho universytetu "Lvivska politekhnika", 1999. – 425 s.
19. Vikovych I.A. Teoriia rukhu transportnykh zasobiv. – Lviv, Vydavnytstvo Lvivskoi politekhniki, 2013. – 672 s.
20. Kozlov O.V. Complex of tasks of monitoring and automatic control of mobile robots for vertical movement / O.V. Kozlov, O.S. Gerasin, G.V. Kondratenko // Shipbuilding & Marine Infrastructure. – 2017. – no. 2 (8). – P. 77-87. (<http://smi.nuos.edu.ua/article/view/120596/116517>) DOI: <https://doi.org/10.15589/SMI20170215>
21. Kushnir V.O., Kondratenko Y.P., Topalov A.M., Gerasin O.S. Mobilnyi robot dlia mekhanichnoho ochyshchennia korpusu sudna [Mobile robot for mechanical cleaning of a ship hull]. Patent UA, no. u201500063, 2015.
22. Kondratenko Y.P. Neuro-nechitki sposterihachi dlia identyfikatsii prytysknoho zusyllia mahnitokerovanykh rushiiv mobilnykh robotiv / Y.P. Kondratenko, J. Rudolph, O.V. Kozlov, Y.M. Zaporozhets, O.S. Gerasin // Tekhnichna elektrodynamika. – 2017. – no. 5. – S. 53-61.
23. Sew-eurodrive. Praktika privodnoy tekhniki. Proektirovanie privodov [Sew-eurodrive. Practice of drive engineering. Designing of drives]. Available at: www.sew-eurodrive.ru/files/pdf/11322853.pdf.

*Рецензент: д.т.н., проф. Рябенкий В.М.,
завідувач кафедри теоретичної електротехніки
та електронних систем НУК ім. адм. Макарова*



ПІДГОТОВКА КОРИСТУВАЧІВ НОВОЇ ВЕРСІЇ ПРОГРАМИ 1С: ПІДПРИЄМСТВО 8.3. ДЛЯ УКРАЇНИ

УДК 510.6

ЗАХАРЧЕНКО Леонід Миколайович

аспірант кафедри інформаційних технологій
Херсонського національного технічного університета,

Наукові інтереси: нові інформаційні технології.

e-mail: zraissa2@gmail.com

ЗАХАРЧЕНКО Раїса Миколаївна

к.т.н., доцент кафедри інформаційних технологій
Херсонського національного технічного університета,

Наукові інтереси: нові інформаційні технології.

e-mail: zraissa2@gmail.com

КІРЮШАТОВА Тетяна Григорівна

к.т.н., доцент кафедри інформаційних технологій
Херсонського національного технічного університета,

Наукові інтереси: нові інформаційні технології.

e-mail: tanyakir1963@gmail.com

КІРЮШАТОВА Катерина Володимирівна

асистент кафедри інформаційних технологій
Херсонського національного технічного університета,

Наукові інтереси: нові інформаційні технології.

e-mail: tanyakir1963@gmail.com

ВСТУП

У даній статті проведено дослідження нових можливостей версії програми 1С: Підприємство 8.3 для України та необхідність підготовки кадрів для роботи з нею. Приведений приклад методики обліку по нарахуванню та виплаті зарплати в програмі.

Мета дослідження. В 2018 році планується виправити помилки і ввести нову версію в Україні. Всі спеціалісти, які працювали з програмою 1С: Підприємство з попередніми версіями, у зв'язку зі змінами в бухгалтерському обліку та з появою нових можливостей цієї версії програми, бажають навчитися та працювати по-новому. Нова конфігурація працює тільки на платформі 8.3, має нові можливості: крупний шрифт, великі пробі-

ли між об'єктами, активізовані елементи виділені кольором і т.д. Програма дає можливість працювати через Інтернет за допомогою мобільних пристроїв, за допомогою сенсорного екрану. Особливо приваблює фахівців і нові можливості по налаштуванню програми самими користувачами. У користувача залишилася також можливість налаштування конфігурації в режимі Конфігуратор з використанням мови програмування 1С [2].

В новій версії в Конфігураторі, де постійно необхідно користуватися пошуком, полегшена робота програміста. Типові конфігурації достатньо об'ємні і пошук займає достатньо багато часу, а тому було добавлено нове поле пошуку, яке розташоване над деревом кон-

фігурації. Нове поле пошуку в конфігурації *Поиск* заповнюється необхідними даними для пошуку, а далі виконується аналіз властивостей об'єктів конфігурації: *Имя*, *Синоним* і *Комментарий*. Дерево конфігурації фільтрується дуже швидко і в ньому залишаються тільки об'єкти, які задовольняють введеному фільтру. При роботі з підсистемами для швидкого відображення об'єктів, які входять в одну конкретну підсистему, в контекстному меню додано новий пункт *Объекты подсистемы* рис. 1 [8].

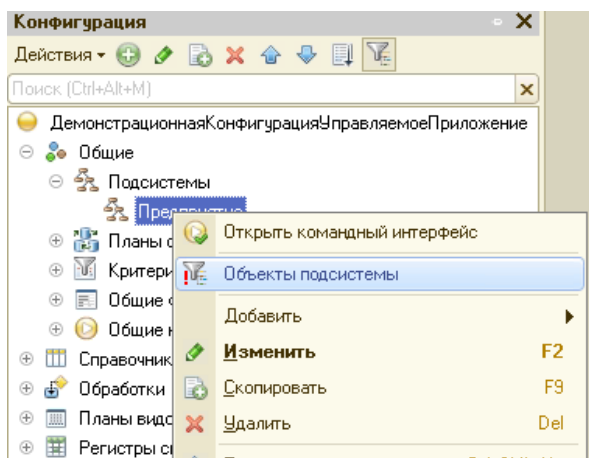


Рисунок 1. – Екранна форма вікна програми в режимі Конфігуратора

Програмістами частіше всього використовується відбір саме по одній підсистемі, а тому таке нововведення економить їх час.

При роботі групи програмістів з конфігурацією нової платформи не завжди вдається контролювати зрозумілість коду, відповідність стандартам. У новій версії з'явилися так необхідні механізми рефакторингу коду та інструменти роботи з модальними викликами в контекстному меню текстового редактору Конфігуратора в окремому меню *Рефакторинг* [8].

З перерахованого випливає необхідність підготовки як користувачів нової версії програми 1С: Підприємство 8.3 так і програмістів для налаштування конфігурації відповідно до представлених вимог.

Основна частина. Очевидна необхідність підготовки користувачів програми 1С: Підприємство 8.3 для України [1,4]. Важливо навчити користувачів користуватися новими можливостями інтерфейсу «Таксі» у

програмі, де вони можуть: виконати налаштування початкової сторінки, налаштування панелей, переглядати історію або користуватися можливістю обирати вибране. У програмі на весь екран відкривається лише одне вікно і користувачеві необхідно економити простір, тобто закривати відкриті вікна. За рахунок розташування основного ресурсу на сервері програма працює набагато швидше [5].

Користуватися програмою зручно і бухгалтерам і менеджерам. В розділі *Керівнику* є можливість формувати звіти і отримувати їх, знаходячись віддалено, по залишках на складах, є можливість переглянути заборгованість покупців, а в розділі *Адміністрування* переглянути активність користувачів та виконати аналіз їх активності і т.д. Звіти у програмі мають графічне та табличне представлення.

Важливо навчитися користувачеві, при необхідності, виконувати доступні налаштування програми в режимі користувача, наприклад, додавати додаткові реквізити в розділі *Адміністрування*.

Методику роботи з новою версією програми розглянемо на прикладі використання розділу *Зарплата і кадрові*. Вона може бути розрахована за схемою на рис. 2.

Для користувачів нової версії програми важливо виконати попередні налаштування та вірно внести всю необхідну інформацію про співробітників [9,10]. Для цього необхідно вибрати в розділі *Адміністрування* пункт *Параметри обліку* і вказати де ведеться облік розрахунків з заробітної плати: у цій програмі чи у зовнішній програмі, облік ведеться по кожному працівнику чи зведено по всім працівникам та обрати кадровий облік спрощений чи повний. Наступна дія - необхідно заповнити довідники *Фізичні особи* і *Співробітники*. Так як нарахування в програмі заробітної плати ведеться автоматично, то в довіднику *Фізичні особи* необхідно внести інформацію про фізичних осіб на всіх закладах: *Основне*, *Робота*, *Особові рахунки працівників за зарплатними проектами* (для виплати заробітної плати через банківські картки), *Склад родини* та *Спеціальні статуси фізичних осіб*. Аналогічно необхідно внести інформацію в довідник *Співробітники* і звернути особливу увагу на закладку *Облік витрат*.

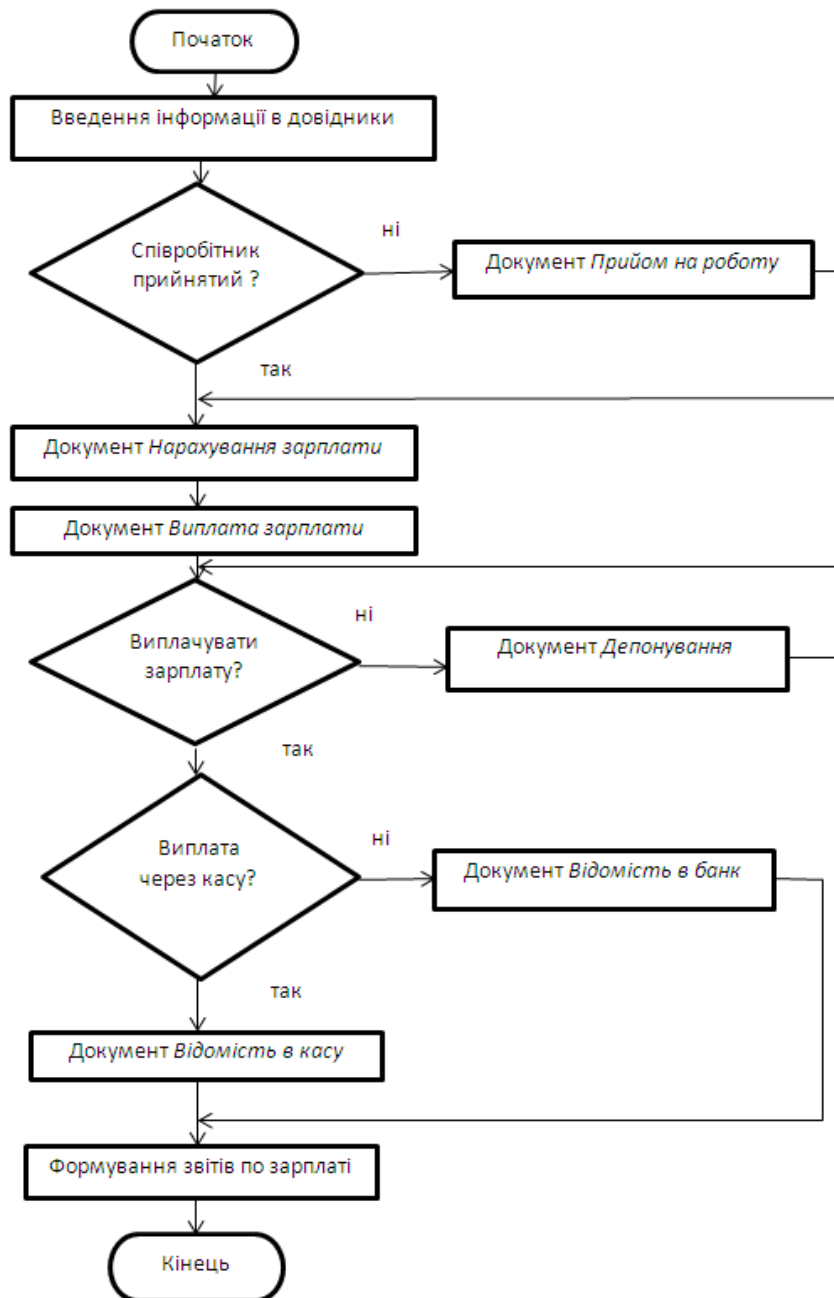


Рисунок 2. – Схема нарахування зарплати

В розділі *Зарплата і кадри* обрати *Налаштування обліку зарплати по організації*. На цій закладці необхідно вказати *Графік роботи* та *Округлення*. На закладці *Відображення в обліку* необхідно обрати: *Настройка відображення за підрозділами* чи *Настройка відображення за співробітниками* (рис. 3).

Самі способи відображення за співробітниками необхідно внести до довідника *Способи відображення*

зарплати в регламентованому обліку. Приклад внесення цієї інформації приведено на рис. 4. За допомогою даного режиму виконується віднесення затрат по витратам.

На закладці *Податки та внески* обрати категорію застрахованих осіб та вказати ставку ЄСВ ФОП працівників організації (якщо вона не буде зазначена то буде використовуватися загальна ставка 0.22) і т.д.

Після внесення інформації до довідників *Фізичні особи* і *Співробітники* та налаштування зарплати по організації необхідно виконати прийом співробітників на роботу, де вказати: організацію, підрозділ, посаду, вид зайнятості, дату прийому, категорію ЄСВ [9, 10].

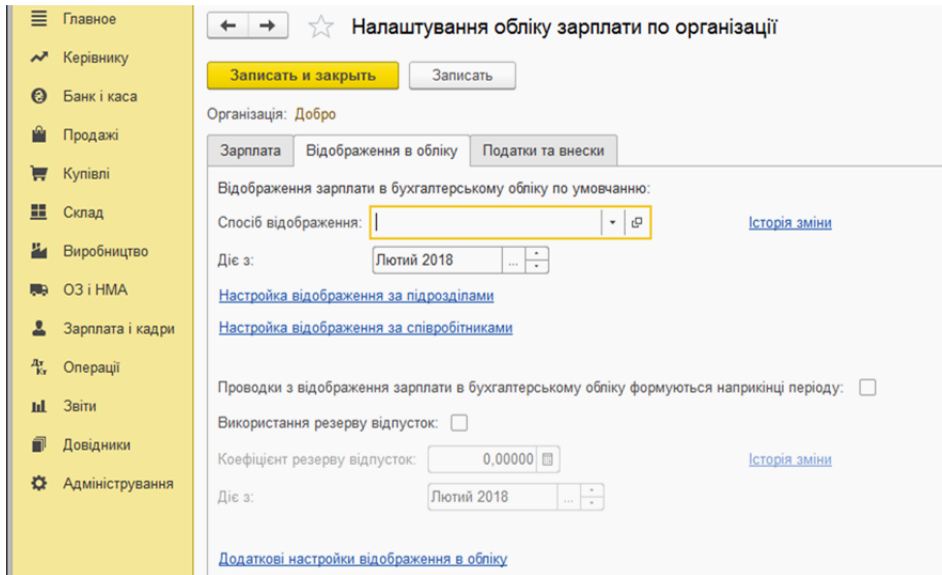


Рисунок 3. Екранна форма вікна програми в режимі *Наляштування обліку зарплати по організації*

Перед нарахуванням заробітної плати, при наявності пільги у співробітника, необхідно сформувати *Заяву на застосування пільги ПДФО*. Пільгу користувач має можливість вибрати зі списку, а якщо така відсутня, то є можливість додати необхідну. Таке налаштування дасть можливість виконати нарахування заробітної плати з урахуванням пільги автоматично. На рис. 5 приведено приклад нарахування зарплати з урахуванням базової пільги 169.1.1.

Після виконаних всіх налаштувань та внесення необхідної інформації для нарахування заробітної плати

необхідно перейти в розділ *Зарплата і кадрові*, де обрати режим *Нарахування зарплати*. У відповідному документі заповнити всі реквізити, а для заповнення табличної частини скористатися кнопкою *Заповнити*. Всі нарахування та утримання будуть виконані автоматично і при необхідності користувач зможе виконати редагування за допомогою кнопок *Корегувати внески*, *Корегувати ПДФО*, які для роботи необхідно активізувати за допомогою прапорця.

Наименование	Раху... Дт	Субконто	Раху... Кт	Субконто	Под. призн. витрат (до 01...	Под. призн. (ПДВ)	Стратегія відображення ЄСВ	Спосіб відображення ЄСВ
Зарплата (ОПЗ)	91	Оклади й тарифи (ОПЗ) (не вкл.) Госп. д-сть	661		Пропорц. опод. ПДВ		Як нарахування	Як нарахування
Зарплата (Сб)	93	Оклади й тарифи (Сб)	661				Як нарахування	Як нарахування
Зарплата електроенергії	232	Електроенергія	661				Як нарахування	Як нарахування

Рисунок 4. - Екранна форма вікна програми в режимі *Способи відображення зарплати в регламентованому обліку*

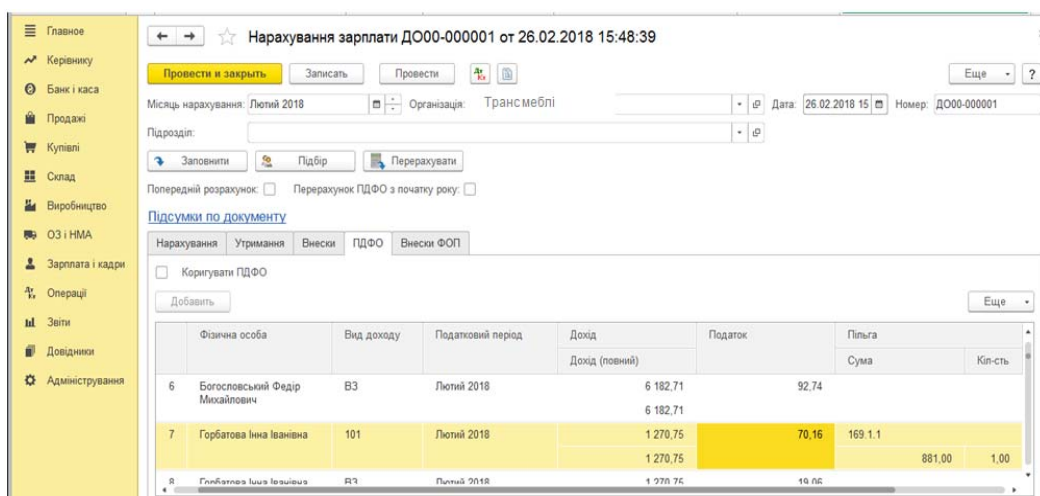


Рисунок 5. - Екранна форма вікна програми в режимі Нарахування зарплати

Разові нарахування (відпускні, лікарняні і т.д.) вводяться вручну. Після виконання різного роду коригування та корегування відпрацьованого часу (для тих співробітників, які працювали не всі дні місяця) необхідно використати кнопку *Перерахувати*. В режимі *Підсумки по документу* можна отримати інформацію по нарахуванню заробітної плати, а при натисканні на кнопку із зображенням знаку питання можна отримати більш детальну інформацію.

При необхідності виконати розрахунок заробітної плати за першу половину місяця в програмі в режимі нарахування заробітної плати є режим *Попередній розрахунок*.

Після проведення документу *Нарахування зарплати* автоматично виконується формування проводок по відображенню зарплати в бухгалтерському обліку, або їх формування можна виконати за допомогою документу *Відображення зарплати в бухгалтерському обліку*.

Необхідно звернути увагу користувачів на заміну відомості на виплату зарплати в попередніх версіях на документи *Відомість в банк*, *Відомість в касу* для видачі зарплати готівкою.

Для зручності у роботі з документами користувачеві доступна можливість коригування їх шапки за допомогою кнопки *Еще*, де необхідно обрати пункт *Изменить форму* (рис. 6). В попередніх версіях така процедура виконувалася тільки в режимі Конфігуратора [9, 10].

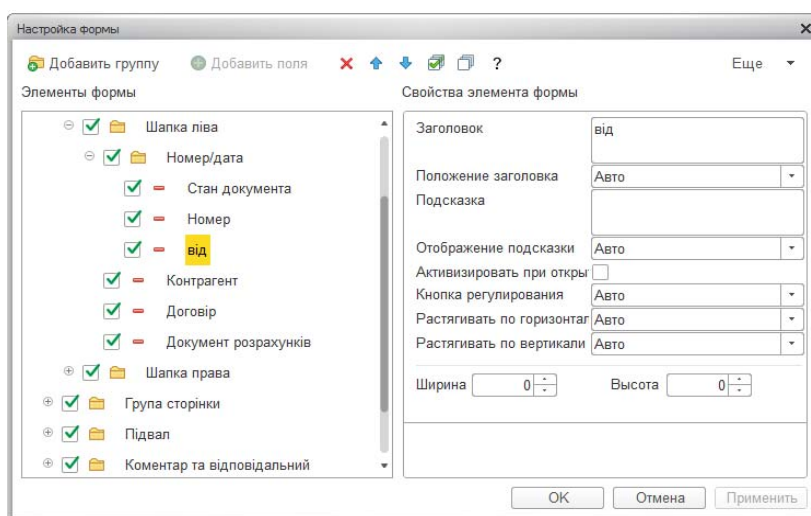


Рисунок 6. – Екранна форма вікна налаштування форми

Також є можливість в новій версії програми виконати депонування зарплати за допомогою документу *Депонування*. Для виплати такого виду формується

документ *Відомість в касу* або *Відомість в банк* з видом виплати *Депоненти* рис. 7.

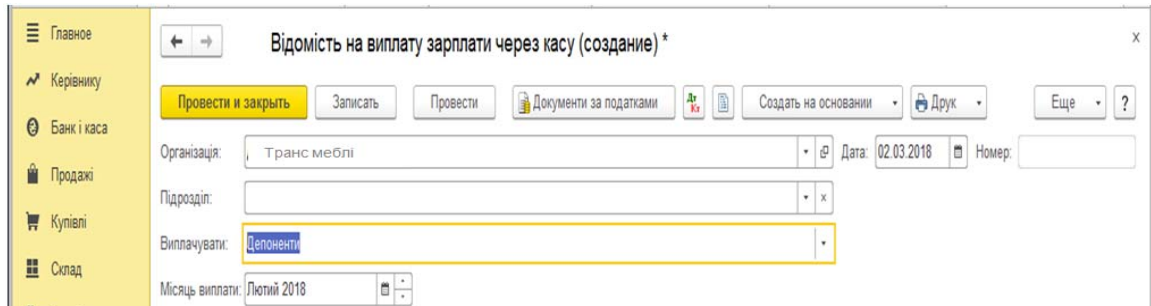


Рисунок 7. - Екранна форма вікна документу *Відомість на виплату зарплати через касу*

Як і в попередніх версіях, в програмі є можливість автоматичного формування стандартних та регламентованих звітів. У розділі *Зарплата і кадри* можна сформувати *Розрахунковий лист*, звіт *Аналіз ПДФО по співробітникам* та інші звіти [3].

ВИСНОВКИ

В статті описані нові можливості версії програми 1С: Підприємство 8.3 та ефективність їх використання.

Приведений приклад методики обліку по нарахуванню та виплаті зарплати в програмі, показані нові можливості для фахівців. Окремо виділені нові можливості роботи з програмою для менеджерів. Приведено нові можливості роботи в конфігураторі для програмістів з використанням мови програмування 1С. Показана необхідність підготовки як користувачів так і програмістів для користування новою версією програми.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ:

1. Hodakov V.E., Bardachev YU.N., Martynov A.N. Avtomatizaciya i komp'yuterizaciya informacionnyh processov uchrezhdenij: Uchebnoe posobie.–Kiev–Herson: UMO, 1997. – 264 s.
2. Dubyanskij V.M. 1S: Predpriyatje. Konfigurirovanie i administrirovanie dlya nachinayushchih. EHkspress-kurs. - SPB.: BHV – Peterburg, 2005.- 176 s.
3. Hodakov V.E, Kiryushatova T.G., Zaharchenko R.N., Karamushka M.V. Komp'yuternye tekhnologii obrabotki uchetnoj informacii. Uchebnoe posobie dlya studentov vuzov / pod red. V.E. Hodakova. – Herson: Grin' D.S., 2012. – 522 s.
4. Pecherskih V., Bel'cev G. Vnedrenie ERP-reshenij na platforme 1S:Predpriyatje 8 /BHV-Peterburg. 2015. - 146 s.
5. Hrustaleva E. YU. 101 sovet nachinayushchim razrabotchikam v sisteme 1S:Predpriyatje 8./1S-Publishing. 2015. - 284 s.
6. Radchenko M. G., Hrustaleva E. YU. 1С:Predpriyatje 8.3. Prakticheskoe posobie razrabotchika /Izdatel'stvo 000 "1S-Publishing". 2016. - 965 s.
7. <https://profbu1c.ru/articles/1s-bukhgalteriya/Vyacheslav Krasnov. 1S 8.3 Samouchitel' - ponyatnyj kurs dlya nachinayushchih>.
8. <https://profbu1c.ru/articles/1s-bukhgalteriya/Vyacheslav Krasnov. Ispravlenie oshibok v pervichnyh dokumentah v 1s 8.3 s provodkami>.
9. <http://programmist1s.ru/nachislenie-zarplatyi-v-1s-8-3-bukhgalteriya-po-shagam/> Nachislenie i vyplata zarabotnoj platy v 1S 8.3 Buhgalteriya 3.0 poshagovo.
10. <https://1s83.info/obuchenie/1s-buhgalteriya/nachislenie-zarplatyi-v-1s-8-3.html>. Nachislenie zarplaty v 1S 8.3.

Рецензент: д.т.н., проф. Марасанов В. В.
Херсонський національний технічний університет



МОДЕЛІ ТА МЕТОДИ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ УПРАВЛІННЯ АГРАРНОГО СЕКТОРУ ЕКОНОМІКИ ЗА ДОПОМОГОЮ ДОСТАТНІХ УМОВ ОПТИМАЛЬНОСТІ

УДК 681.3.06

ЛОБОДА Олена Миколаївна

кандидат технічних наук, доцент, ДВНЗ «Херсонський державний аграрний університет», м.Херсон

Наукові інтереси: математичне моделювання, економіко-математичне моделювання, інформаційні технології

ДИМОВ Володимир Степанович

К.т.н., доцент кафедри Інформаційних технологій

Наукові інтереси: методи дослідження складних динамічних систем,
оптимізація в телекомунікаційних системах, нейронні мережі.

e-mail: vdymov@rambler.ru

ВСТУП

Одним з головних напрямків, в умовах складної ринкової економіки, є підвищення ефективності функціонування аграрних підприємств, що здійснюється шляхом побудови автоматизованих систем управління і використання сучасних інформаційних технологій. Рішення задачі оптимального управління, в цих умовах, призводить до вирішення завдання управління у вигляді розподілу ресурсів між галузями. Знаходження оптимальних управлінь, що визначають найбільшу ефективність результатів функціонування, передбачає побудову моделей об'єктів управління, а також рішення багатокрокового завдання знаходження оптимальних управлінь при заданому функціоналі ефективності функціонування.

У сучасних умовах вимоги до ефективності функціонування підприємства не відповідають можливостям традиційного управління. Дослідження орієнтоване на створення інформаційних методів і моделей автоматизованих систем управління на базі сучасних комп'ютерних засобів дозволяє вирішувати завдання вибору управлінських рішень по окремим галузям, а також по господарству в цілому на основі порівняльного аналізу виробничих функцій. Завдання особливо актуальна, в

умовах ринкової економіки і спроба вирішувати цю задачу в умовах конкуренції, безумовно, може бути використана керівником господарства. Тому проведення нових досліджень, розробка моделей, алгоритмів, методів, програм, інформаційних технологій для удосконалення функціонування підприємств є актуальною науковою задачею.

ПОСТАНОВКА ПРОБЛЕМИ

Виробництво - складний керований процес перетворення ресурсів в суспільний продукт. При розробці економіко-математичного апарату для аналізу, планування і прогнозування виробництва створюється система моделей, яка базується на уявленні економіки аграрного підприємств як складної ієрархічної системи. При математичному моделюванні взаємозв'язок між факторами виробництва і його результатом зазвичай відображають за допомогою виробничих функцій. При побудові виробничих функцій слід мати на увазі, що витрати факторів виробництва на випуск продукції завжди невід'ємні. Крім того, при моделюванні виробничих функцій треба відзначити, що відсутність одного з факторів призводить до нульового випуску продукції. Вважають також, що фактори виробництва змінюються

безперервно, а випуск продукції змінюється досить гладко при зміні факторів, що природно при розгляді виробництва на макрорівні.

Економічно доцільно також, щоб при збільшенні кількості використовуваного ресурсу випуск продукції зростає, тобто для диференційованої виробничої функції можна записати наступні нерівності:

$$\frac{\partial F(K, L)}{\partial K} > 0, \frac{\partial F(K, L)}{\partial L} > 0,$$

де K - основні виробничі фонди;

L - трудові ресурси.

Переліченим умовам відповідають мультиплікативні виробничі функції виду

$$X = aK^\alpha L^\beta, \alpha > 0, \beta > 0,$$

де X - випуск продукції;

α, β - параметри виробничої функції.

Мультиплікативна виробнича функція дає можливість відобразити ефект масштабу виробництва, який існує тільки при одночасній зміні факторів K і L . Нехай ці фактори змінюються в λ разів, тоді

$$F(\lambda K, \lambda L) = \lambda^{\alpha+\beta} F(K, L).$$

В цьому випадку:

1) якщо $\alpha+\beta > 1$, то має місце інтенсивний спосіб розвитку, тобто з ростом масштабу виробництва в λ разів випуск продукції зростає більш ніж в λ разів;

2) якщо $\alpha+\beta < 1$, то зростання масштабу виробництва негативно позначається на випуску продукції, тобто при зростанні витрат в λ разів випуск продукції зростає менш ніж в λ разів;

3) якщо $\alpha+\beta = 1$, то відбувається екстенсивне зростання економіки тільки за рахунок факторів виробництва.

Тривалі спостереження показують, що в умовах чисто екстенсивного виробництва збільшення витрат тільки одного з факторів виробництва призводить до зниження ефективності його використання, тобто $\frac{\partial^2 F(K, L)}{\partial K^2} < 0, \frac{\partial^2 F(K, L)}{\partial L^2} < 0$. Це означає, що кожна наступна одиниця зростаючого фактора з'єднується з

меншою кількістю іншого фактора і його зростання дає зменшення приросту продукції.

Для екстенсивного способу розвитку характерно
$$\lim_{K \rightarrow 0} \frac{\partial F(K, L)}{\partial K} = \infty, \lim_{L \rightarrow 0} \frac{\partial F(K, L)}{\partial L} = \infty,$$
 і
$$\lim_{K \rightarrow \infty} \frac{\partial F(K, L)}{\partial K} = 0, \lim_{L \rightarrow \infty} \frac{\partial F(K, L)}{\partial L} = 0.$$

Виробнича функція Кобба-Дугласа є моделлю екстенсивного способу розвитку

$$X = aK^\alpha L^\beta, \alpha + \beta = 1,$$

де α - коефіцієнт еластичності випуску по виробничим фондам;

β - коефіцієнт еластичності випуску по фонду оплати праці.

Під еластичністю виробничої функції по фактору розуміється відношення відносного приросту функції до відносного приросту фактора. Еластичність чисельно дорівнює числу відсотків, на яке зміниться випуск продукції при зміні фактора на 1%. Неважко показати, що коефіцієнти еластичності можна визначити як відношення граничної ефективності функції по фактору до середньої ефективності:

$$\alpha = \frac{\partial F(K, L) / F(K, L)}{\partial K / K} \quad \text{і} \quad \beta = \frac{\partial F(K, L) / F(K, L)}{\partial L / L}.$$

Важливою характеристикою виробничих функцій є еластичність заміни ресурсів σ , так як вона буває постійною для більшості виробничих функцій, використовуваних в економіко-математичному моделюванні. Еластичність заміни ресурсів показує, на скільки відсотків змінилася фондоозброєність $k=K/L$ при зміні граничної норми заміщення $s=dK/dL$ (граничної фондоозброєності) на 1% при незмінному випуску продукції:

$$\sigma = \left. \frac{d \ln k}{d \ln s} \right|_{F = \text{const}}. \quad \text{Тут під граничною нормою}$$

заміщення розуміють кількість фондів, яке необхідно додатково ввести при зменшенні витрат праці на одиницю, якщо випуск продукції залишиться незмінним. Гранична норма заміщення s визначається з рівняння ізокванти (лінія рівного випуску продукції):

$$dy = \frac{\partial F(K, L)}{\partial K} dK + \frac{\partial F(K, L)}{\partial L} dL \equiv 0.$$

Звідси

$$s = \frac{dK}{dL} = - \frac{\frac{\partial F(K, L)}{\partial L}}{\frac{\partial F(K, L)}{\partial K}},$$

де $\frac{\partial F(K, L)}{\partial L}$ - гранична ефективність по праці;

$\frac{\partial F(K, L)}{\partial K}$ - гранична ефективність по основним виробничим фондам.

Еластичність заміни ресурсів σ для функції Кобба-Дугласа дорівнює $\sigma = \frac{d \ln k}{d \ln s} = 1$, так як для неї гранична норма заміщення $S = \frac{1-\alpha}{\alpha} k$, де $k = K/L$. Часто

економічні міркування підказують, що хоча еластичність заміщення ресурсів і можна вважати постійною, але все-таки вона відмінна від одиниці. У зв'язку з цим еластичність заміни ресурсів для функції Солоу

$$\sigma = \frac{d \ln k}{d \ln s} = \frac{1}{1 + \rho}.$$

ВИРІШЕННЯ ЗАДАЧІ

Розглянемо економіку сільськогосподарського підприємства, що характеризується в кожен момент часу t набором змінних X, Y, C, K, L, I , де X - інтенсивність валового продукту; Y - інтенсивність кінцевого продукту; C - невиробниче споживання; I - валові капітальні вкладення; K - обсяг основних виробничих фондів; L - трудові ресурси. Ці змінні взаємопов'язані. Перш за все має місце умова балансу в кожен момент часу $X = aX + Y$, де $0 < a < 1$.

У свою чергу, кінцевий продукт розподіляється на валові капітальні вкладення і невиробниче споживання $Y = I + C$, де валові капітальні вкладення витрачаються на приріст основних виробничих фондів і їх відновлення за рахунок амортизаційних відрахувань: $I = \dot{K} + \mu K$, де μ - коефіцієнт амортизації. Тоді $\dot{K} = I - \mu K$ або

$$\dot{K} = (1 - a)(1 - u)X - \mu K \quad (1)$$

де $u = C/Y$ - доля невиробничого споживання:

$$0 \leq u \leq 1. \quad (2)$$

Будемо вважати, що розміри валового продукту визначаються заданою виробничою функцією, що характеризує можливості виробництва в залежності від величини виробничих фондів K , трудових ресурсів і часу t , тобто

$$0 \leq X \leq F(K, L, t). \quad (3)$$

Передбачається, що виробнича функція $F(K, L, t)$ неперервна і двічі диференційована, причому виконуються наступні умови:

- 1) функція завжди невід'ємна: $F(K, L, t) > 0$;
- 2) функція зростає по кожному з аргументів

$$\frac{\partial^2 F}{\partial K^2} > 0, \quad \frac{\partial^2 F}{\partial L^2} > 0;$$

- 3) якщо хоча б один з ресурсів K або L дорівнює нулю, то і $F(K, L, t) = 0, F(0, L, t) = 0, F(K, 0, t) = 0$;

- 4) передбачається, що з ростом кожного з аргументів приріст валового продукту зменшується:
- $$\frac{\partial^2 F}{\partial K^2} < 0, \quad \frac{\partial^2 F}{\partial L^2} < 0$$

$$5) \lim_{K \rightarrow 0} \frac{\partial F}{\partial K} = \infty, \quad \lim_{L \rightarrow 0} \frac{\partial F}{\partial L} = \infty$$

- 6) функція має властивість однорідності по аргументам K і L , тобто зміна масштабу виробництва призводить до пропорційної зміни випуску продукту: $F(\lambda K, \lambda L, t) = \lambda F(K, L, t)$. Параметр t вводиться в виробничу функцію, щоб врахувати цілий ряд зовнішніх факторів, що впливають на модель, в тому числі вплив науково-технічного прогресу;

- 7) функція зростає за часом: $\frac{\partial F}{\partial t} > 0$.

Рішення завдання будемо шукати за умови

$$K \geq K_3 \quad (4)$$

де K_3 - заданий рівень основних виробничих фондів.

Нехай задані виробничі фонди в початковий момент часу:

$$K(0) = K_0 \quad (5)$$

Допустима множина M в розглянутій задачі описується умовами (2)-(5). Допустимий процес представлений сукупністю функцій $v = (K(t), X(t), u(t))$, що задоволь-

няє цим умовам. Він описує стан господарства, а X і u - управління [1]. Очевидно, що такий процес не єдиний.

Задача управління даною економікою господарства полягає в тому, щоб знайти такий процес $v=(K(t),X(t),u(f))$, який забезпечував би найбільше середнє споживання на даному часовому інтервалі з урахуванням дисконтування споживання, тобто

$$f = \int_0^T e^{-\delta t} \frac{C}{L} dt.$$

Проведемо редукцію задачі. Для цього введемо в диференціальне рівняння (1) відносні змінні: $k=K/L$ - фондоозброєність, $c=C/L$ - середнє споживання, $x=X/L$ - продуктивність праці. Так як $K=kL$, $X=xL$, то рівняння (1) набуде вигляду $(\dot{k}L) = (1-a)(1-u)xL - \mu kL$. Враховуючи правило диференціювання складної функції, одержимо $\dot{K} = (\dot{k}L) = \dot{k}L + k\dot{L}$.

Будемо вважати, що приріст трудових ресурсів здійснюється з постійним темпом, тобто $\dot{L} = nL$. Тоді $(\dot{k}L) = (\dot{k} + kn)L$. Остаточнo диференціальне рівняння зв'язку в відносних змінних набуде вигляду

$$k = (1-a)(1-u)x + (\mu + n)k.$$

Обмеження на управління u залишається, тобто

$$0 \leq u \leq f \tag{6}$$

а на продуктивність праці x набуде вигляду

$$0 \leq x = f(k,t), \tag{7}$$

де $f(k,t) = \frac{1}{L} F(K,L,T)$.

Обмеження на виробничі фонди замінимо обмеженнями на фондоозброєність:

$$k(t) \geq k_3(t). \tag{8}$$

$$k(0) = k_0. \tag{9}$$

Проведемо перетворення функціоналу до відносних змінних:

$$f = \int_0^T e^{-\delta t} (1-a)ux dt \rightarrow \max \text{ або} \tag{10}$$

$$I = \int_0^T e^{-\delta t} (1-a)ux dt \rightarrow \min$$

Потрібно визначити процес $v=(k(t),u(t),x(t))$, що звертає в мінімум функціонал (10) на безлічі (6)-(9).

Таким чином, у зредукованому завданні станом системи є фондоозброєність k управлінням - продуктивність праці x і частка споживання u . Рівнянням процесу служить диференціальне рівняння зростання фондоозброєності.

Для вирішення поставленого завдання скористаємося теоремою про достатні умови оптимальності. Введемо функцію R [2].

$$R(k,x,u,t) = \frac{\partial \phi(k,t)}{\partial k} =$$

$$= [(1-a)(1-u)x - (\mu+n)k] + e^{-\delta t} (1-a)ux + \frac{\partial \phi(k,t)}{\partial t}$$

де $\phi(k,t)$ - функція, підібрана з конкретних передумов про тип процесу і необхідного наближення.

Виділимо в R складові, що містять компоненти вектора управління (u,x) , прирівняємо суму коефіцієнтів при ньому до нуля. Тим самим на ϕ накладається вимога $-(1-a)\phi_k + e^{-\delta t}(1-a) = 0$; отже, $\phi_k(t,k) = e^{-\delta t}$. Тоді $\phi(t,k) = ke^{-t} + c(t)$, де $c(t)$ - довільна функція. Припустимо $c(t) = 0$, тоді $\phi(t,k) = ke^{-\delta t}$ і $\phi'(t,k) = -\delta ke^{-\delta t}$.

При цій умові функція R не залежить від u : $R(t,k,x) = e^{-t}[(1-a)x(\mu+n)k] - e^{-\delta t}\delta k = e^{-\delta t}[(1-a)x(\mu+n+\delta)k]$. Оптимальні $\bar{k}(t), \bar{x}(t)$ знайдемо з умови $\bar{k}(t), x(t) \rightarrow \max_{0 \leq x \leq f(k,t)} R(t,k,x)$, так як $a < 1$, то $(1-a) > 0$ і, отже, $\max R$ досягається при $x = f(k,t)$.

Для однопродуктової моделі це рівняння очевидно, але в багатогалузевий моделі може виявитися, що деякі галузі недовантажені.

Проведемо тепер максимізацію R по k при оптимальному $x = \bar{x}$. Позначимо:

$$R_1(t,k) = \max_{0 \leq x \leq f(k,t)} (t,k,x) = e^{-\delta t} [(1-a)f(k,t) - (\mu+n+\delta)k]$$

. Отже, максимум $k = \bar{k}$ буде результатом максимізації R_1 по k .

Введемо $r(t,k)=(1-a)f(k,t)-(\mu+n+\delta)k$. Тоді, враховуючи, що $e^{-\delta t} > 0$, можна записати $\bar{k}(t) = \arg \max_k r(t,k) \forall t \in [0, T]$. Проаналізуємо поведінку функції $r(t,k)$ по k . Ця функція є сумою двох доданків: виробничої функції з точністю до постійного множника і лінійного вираження.

Необхідною умовою максимуму $r(t,k)$ по k є рівність нулю частинній похідної: $\frac{\partial r(t,k)}{\partial k} = 0$. З огляду на те, що $f(k,t) = be^{\rho t} k^\alpha$, маємо $(1-a)bae^{\rho t} k^{\alpha-1} - (\mu+n+\delta) = 0$. Так як $0 < \alpha < 1$ і $1-a = \beta$, то

$$\hat{k}(t) = \left(\frac{(1-a)ba}{\mu+n+\delta} \right)^{\frac{1}{\beta}} e^{\frac{\rho t}{\beta}}. \quad (11)$$

Знайдене $\hat{k}(t)$ назвемо магістраллю даної динамічної моделі економіки підприємства. Вона грає важливу роль в структурі оптимального рішення. Управління, що реалізує цю магістраль, знайдемо підстановкою знайденого $\hat{k}(t)$ в диференціальне рівняння розвитку системи (1): $\dot{k}(t) = (1-a)(1-u)x(t) - (\mu+n)\hat{k}(t)$. Так як $\bar{x}(t) = f(k,t)$, де $f(k,t) = be^{\rho t} k^\alpha$ є виробничою функцією, то, вирішуючи рівняння процесу щодо u , отримаємо

$$\hat{u}(t) = 1 - \frac{\dot{\hat{k}}(t) - (\mu+n)\hat{k}(t)}{(1-a)be^{\rho t}\hat{k}^\alpha}.$$

З формули (11) знайдемо $\dot{\hat{k}}(t) = \hat{k}(t) \frac{\rho}{\beta}$.

Тоді

$$\hat{u}(t) = 1 - \frac{\hat{k}(t) \left(\mu+n + \frac{\rho}{\beta} \right)}{(1-a)be^{\rho t}\hat{k}^\alpha}.$$

Або

$$\hat{u}(t) = 1 - \frac{\mu+n + \frac{\rho}{\beta}}{(1-a)be^{\rho t}\hat{k}^\alpha}.$$

Так як

$$\hat{k}^{\alpha-1} = k^{-\beta} = \left(\frac{(1-a)ba}{\mu+n+\delta} \right)^{-1} e^{-\rho t},$$

то отримаємо оптимальне управління

$$\hat{u}(t) = 1 - \alpha \frac{\mu+n + \frac{\rho}{\beta}}{\mu+n+\delta} \quad (12)$$

в припущенні, що $0 \leq \hat{u} \leq 1$.

Розглянемо спеціальний випадок, коли крайові умови лежать на магістралі:

$$k_0 = \hat{k}(0), k_1 = \hat{k}(T). \quad (13)$$

Тоді процес $\hat{v} = (\hat{k}, \hat{u}, f(k)) \in M$ оптимальний. Дійсно цей процес забезпечує максимум R при кожному t :

- а) по u - в силу незалежності R від управління u , що досягається вибором функції $\phi(k,t)$;
- б) по k і x - з побудови.

З іншого боку \hat{v} представляє допустимий процес, так як:

- а) задовольняє рівняння процесу (u знаходили підстановкою \hat{k} в рівняння процесу);
- б) $0 \leq u \leq 1$;
- в) граничні умови були спеціально підібрані.

Відзначимо, що умови реалізованості $0 \leq u \leq 1$ в даній задачі виконуються. Це можна перевірити [3, 4]. Для функції Кобба-Дугласа економічної магістраллю є крива постійного темпу зростання фондоозброєності, пропорційного темпу зростання технічного прогресу ρ , а оптимальне керування, що реалізує дану магістраль, постійна величина (12).

Таким чином, для спеціально підібраних крайових умов (13) магістраль є оптимальним режимом розвитку економіки господарства: $\hat{k}(t) = \arg \max_{-\infty < k < \infty} R(t,k)$. У

всіх випадках магістралі в структурі рішення відводиться суттєва роль. Насправді дуже рідко зустрічаються випадки, коли крайові умови належать магістралі. Розглянемо загальний випадок. Нехай $k_0 \neq \hat{k}(0), k_1 \neq \hat{k}(T)$. Для вирішення цього завдання можна застосувати прийом, аналогічний вирішенню завдання, лінійної щодо управління. Знайдемо $\bar{k}(t) = \arg \max_{k \in \bar{V}_t^i} R(t,k)$. У реальних економічних задачах мінімальний рівень споживання строго позитивний: $0 < u_1 \leq u \leq 1$.

Побудуємо границі $\gamma_{ij}(t), i=1,2, j=0,1$, допустимої області V . Функції $\gamma_{ij}(t)$ є рішеннями диференціального рівняння процесу

$$\dot{k} = (1-a)(1-u)f(t, k) - (\mu+n)k \quad (14)$$

при відповідних крайових умовах (якщо $j=0$, то береться $k(0)=k_0$, якщо $j=1$, то використовується $k(T)=k_1$) і обмеженнях на керування (якщо $i=1$, то береться нижня межа $u=u_1$, якщо $i=2$, то $u=1$).

ОСНОВНІ РЕЗУЛЬТАТИ

Розглянемо приклад, коли $k_0 < \hat{k}(0), k_1(T) > \hat{k}(T)$, тоді оптимальна траєкторія буде складатися з трьох ділянок з моментами перемикавання τ_1 і τ_2 , де τ_1 є точкою перетину границі γ_{10} з магістраллю $\hat{k}(t)$, а τ_2 - точкою перетину магістралі $\hat{k}(t)$ з границею γ_{11} . Спочатку на тимчасовому інтервалі $(0, \tau_1)$ майже все вкладається в накопичення (споживання в цей період на мінімальному рівні u_1). Починаючи з τ_1 розвиток йде по магістралі $\hat{k}(t)$ аж до моменту τ_2 , з якого знову майже все вкладається в економіку (споживання знову знаходиться на нижньому рівні u_1).

Знайдемо рішення диференціального рівняння (14). З огляду на, що $f(t) = be^{\rho t} k^\alpha$, отримаємо

$$\dot{k} = (1-a)(1-u)be^{\rho t} k^\alpha - (\mu+n)k. \quad (15)$$

Перепишемо рівняння (15) у вигляді:

$$\dot{k} + \lambda k = b(1-a)(1-u)e^{\rho t} k^\alpha, \quad (16)$$

де $\lambda = \mu + n$

Введемо нову змінну $z = k^\beta$, де $\beta = 1 - \alpha$. Так як $\dot{z} = (1-\alpha)k^{-\alpha} \dot{k}$, то маємо

$$\dot{z} = \frac{k^\alpha}{(1-\alpha)} \dot{z}. \quad (17)$$

Підставляючи (17) в диференціальне рівняння (16), отримуємо

$$(1-\alpha)^{-1} k^\alpha \dot{z} + \lambda k = b(1-a)(1-u)e^{\rho t} k^\alpha. \quad (18)$$

Розділивши обидві частини диференціального рівняння (18) на k^α , отримаємо

$$(1-\alpha)^{-1} \dot{z} + \lambda z = b(1-a)(1-u)e^{\rho t}. \quad (19)$$

Загальне рішення лінійного неоднорідного диференціального рівняння дорівнює сумі загального рішення однорідного диференціального рівняння Z_{00} і приватного рішення неоднорідного рівняння $Z_{\text{чн}}$: $Z = Z_{00} + Z_{\text{чн}}$. Знайдемо загальний розв'язок лінійного однорідного рівняння $(1-\alpha)^{-1} z + \lambda z = 0$, характеристичним рівнянням, якого є $(1-\alpha)^{-1} q + \lambda = 0$ [5, 8, 9]. Звідси визначимо корінь характеристичного рівняння: $q = -\lambda\beta$. Тоді загальне рішення однорідного диференціального рівняння набуде вигляду $z_{00} = C_j e^{-\lambda\beta t}$, $i=0,1$.

Частинне рішення неоднорідного диференціального рівняння шукаємо у вигляді правої частини (19) [6]:

$$z_{\text{чн}} = B e^{\rho t}, \quad (20)$$

де B - невизначений коефіцієнт, який підлягає визначенню.

Диференціюючи (20) його по t , отримаємо $\dot{z}_{\text{чн}} = B\rho e^{\rho t}$. Підставивши $z_{\text{чн}}(t)$ в рівняння (19): $(1-\alpha)^{-1} B\rho e^{\rho t} + \lambda B e^{\rho t} = b(1-a)(1-u)e^{\rho t}$. Після скорочення на $e^{\rho t}$ отримаємо $(1-\alpha)^{-1} B\rho + \lambda B = b(1-a)(1-u)$. Звідки $B = \frac{b(1-a)(1-u)}{\rho/\beta + \lambda}$.

Тоді загальне рішення неоднорідного диференціального рівняння (19) має вигляд $z_{00}(t) = C_j e^{-\lambda\beta t} + \frac{b(1-a)(1-u)}{\rho/\beta + \lambda} e^{\rho t}$. Так як

$$z = \frac{1}{k^{\alpha-1}} = k^\beta \quad \text{тобто } k = z^{1/\beta}, \text{ то загальне рішення диференціального рівняння (16) буде мати вигляд}$$

$$k(t) = \left[C_j e^{-\lambda\beta t} + \frac{b(1-a)(1-u)}{\rho/\beta + \lambda} e^{\rho t} \right]^{1/\beta}, \text{ де } j=0,1.$$

Визначимо умови для моментів перемикавання [4,10]. За визначенням, γ_{ij} , $i=1, j=0,1$, є границями допустимої області \tilde{V}^t та виходять як частинні рішення диференціального рівняння (16) при заміні u на граничні значення u_i , $i=1,2$, і виборі C_j , $j=0,1$, в залежності від крайової умови. Тоді

$$\gamma_{ij}(C_j, u_j, t) = \left[C_j e^{-\lambda \beta t} + \frac{b(1-a)(1-u)}{\rho/\beta + \lambda} e^{\rho t} \right]^{1/\beta},$$

де $i=1,2, j=0,1$.

Знайдемо інтегральні константи $C_{ij}=0,1$, в залежності від граничних умов. Так як

$$k_0 = k(0) = \gamma_{i0}(C_0, u, 0) = \left[C_0 + \frac{b(1-a)(1-u_i)}{\rho/\beta + \lambda} \right]^{1/\beta},$$

то

$$C_0 = k_0 - \frac{b(1-a)(1-u)}{\rho/\beta + \lambda}.$$

Аналогічно визначаємо C_1 з граничної умови

$$k_1 = k(T) = \gamma_{i1}(C_1, u_i, T) = \left[C_1 e^{-\lambda \beta T} + \frac{b(1-a)(1-u)}{\rho/\beta + \lambda} e^{\rho T} \right]^{1/\beta}.$$

Отримуючи $C_1 = \left[k_1^\beta - \frac{b(1-a)(1-u_i)}{\rho/\beta + \lambda} e^{\rho T} \right] e^{\lambda \beta T}$, знайдемо

точки перемикання. Позначимо через $\tau_{ij}, i=1,2, j=0,1$, точки перетину границь $\gamma_{ij}, i=1,2, j=0,1$, з магістраллю $\hat{k}(t)$. Моменти перемикання τ_{ij} отримаємо, привівнявши $\hat{k}(t_{ij}) = \lambda_{ij}(C_j, u_i, t)$, отримаємо

$$\left(\frac{(1-a)b\alpha}{\mu+n+\delta} \right)^{\frac{\rho}{\beta}} e^{\frac{\rho}{\beta} t} = \left[C_j e^{-\lambda \beta t} + \frac{b(1-a)(1-u_i)}{\rho/\beta + \lambda} e^{\rho t} \right]^{1/\beta}.$$

ЛІТЕРАТУРА:

1. Lyung L. Identifikatsiya sistem. Teoriya dlya pol'zovatelya M.: Nauka, 1991. 432 s.
2. Anfilatov V.S., Yemel'yanov A.A., Kukushkin A.A. Sistemnyy analiz v upravlenii. M.: Finansy i statistika, 2003. 368 s.
3. Brizhan' Í.A. Doslidzhennya faktoriv, shcho vplivayut' na yekonomichniy rizik pidpriemstva. // Regional'ni perspektivi. 2000. №8.S.15-21.
4. Buslenko N.P. Modelirovaniye slozhnykh sistem. M.: Nauka, 1968. 355 s.
5. Gill F., Myurrey U., Rayt M. Prakticheskaya optimizatsiya. M.: Nauka, 1984.
6. Dubrov A.M. Modelirovaniye riskovykh situatsiy v ekonomike i biznese/ A.M. Dubrov, B.A. Lagosha, Ye.YU. Khrustalev. – M.: Finansy i statistika, 1999. – 172 s.
7. Grop D. Metody identifikatsii sistem. Moskva: Mir, 1979. 302 s.
8. Osnovy teorii optimal'nogo upravleniya. / Red. V.F.Krotova Moskva: Mir, 1984. 430 s.
9. Seydzh E. P., Uayt III CH. S. Optimal'noye upravleniye sistemami. Moskva: Radio i svyaz', 1982. 392 s.
10. Ekonomiko-matematicheskiye metody i prikladnyye modeli / Pod red. V.V. Fedoseyeva. Mmoskva: YUNITI, 1999. 391 s.

Звідси:

$$\frac{(1-a)b\alpha}{\mu+n+\delta} e^{\rho t} = C_j e^{-\lambda \beta t} + \frac{b(1-a)(1-u_i)}{\rho/\beta + \lambda} e^{\rho t}.$$

Інформаційні моделі дозволяють виявити зміни зведених показників і дають цінну інформацію про темпи і пропорції розвитку господарства.

ВИСНОВКИ

В роботі показана необхідність адаптації та доопрацювання моделей і методів управління сільськогосподарськими підприємствами, використовуючи в якості керуючого впливу обсяг інвестицій, а також зроблено уточнення моделі запізнювання при освоєнні капітальних вкладень. Встановлено необхідність створення, на основі достатніх умов оптимальності, моделі оптимального розвитку сільськогосподарського підприємства, що дозволило розробити основну характеристику збалансованого зростання (магістраль) сільськогосподарського підприємства та розглянуто задачу оптимізації моделі з урахуванням запізнювання введення основних виробничих засобів, вибираючи в якості критерій оптимальності, загального для будь-якої економіки, максимум споживання.

Рецензент: д.т.н., проф. Коваленко В.Ф.,
Херсонський національний технічний університет

МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ АВТОМАТИЧНОГО КОНТРОЛЮ ЕНЕРГОЕФЕКТИВНОСТІ ПОДРІБНЕННЯ РУДИ КУЛЬОВИМИ МЛИНАМИ ПРИ СПРАЦЮВАННІ ФУТЕРОВКИ

УДК 681.5.017

МАЦУЙ Анатолій Миколайович

доцент, кандидат технічних наук, доцент кафедри автоматизації виробничих процесів
Центральноукраїнського національного технічного університету, м. Кропивницький.

Наукові інтереси: автоматизація процесів керування, комп'ютеризовані системи управління.

E-mail: matsuyan@mail.ru.

КОНДРАТЕЦЬ Василь Олександрович

професор, доктор технічних наук, Заслужений діяч науки і техніки України, професор кафедри автоматизації виробничих процесів
Центральноукраїнського національного технічного університету, м. Кропивницький.

Наукові інтереси: автоматизація процесів керування, комп'ютеризовані системи управління, інформаційні засоби.

E-mail: kondratetsva@gmail.com.

ВСТУП

Україна є розвинутою металургійною державою, яка за даним показником входить в першу десятку країн світу. В Україні діють шість найбільших у світі гірничо-збагачувальних комбінатів, у складі яких використовуються велика кількість кульових млинів. В той же час вітчизняні залізородні збагачувальні підприємства виробляють магнетитовий концентрат з підвищеною собівартістю порівняно з зарубіжними аналогами, що веде до значних перевитрат електричної енергії, куль і футеровки в процесах рудопідготовки та до зниження конкурентоспроможності продукції на світовому ринку. Особливо великі втрати відмічаються в перших стадіях подрібнення-класифікації вихідної руди. Удосконалення автоматичного управління процесами рудопідготовки є одним з головних шляхів виходу з даної ситуації. Зважаючи на важливість, дані задачі постійно відмічаються в урядових документах України і вносяться в тематичні плани наукових організацій. Розв'язання ряду таких задач, зокрема, включені до тематики наукових досліджень Центральноукраїнського національного техніч-

ного університету. Оскільки дана публікація спрямована на розв'язання частини однієї з таких задач, її тема є актуальною.

Автоматизації рудопідготовки на магнітозбагачувальних фабриках приділяли увагу практично три покоління вчених і практиків. В далекому зарубіжжі найбільш відомими вченими цього напрямку є А.Д. Лінч [1] і Дж.А.Хербст [2, 3]. В Україні такі роботи проводяться досить активно. Основи сучасних підходів заклав О.М. Марюта [4]. В теперішній час особливо активно розробляють ці задачі В.С. Моркун [5, 6], А.І. Купін [7], Є.В. Кочура [8], О.В. Поркуян [9], Н.В. Моркун [10], А.А. Азарян [11], В.В. Тронь [12] та інші. Розвитку автоматизованих систем керування процесами збагачення бідних залізних руд в останні роки присвячена робота [13], в якій зроблено висновок, що існуючі проблеми до кінця не розв'язані. Крім того, в цій же роботі вказується на необхідність розробки сучасних інформаційних засобів. На відсутність надійних засобів контролю необхідної точності та на значну їх вартість звертається увага в [7]. Глибоко проаналізовані системи, що вико-

ристовуються нині на залізрудних збагачувальних фабриках в роботі [14]. В ній сказано, що поряд зі значними досягненнями не всі задачі розв'язані. Авторами роботи [11] доведено, що реальний шлях підвищення ефективності збагачення залізних руд – це автоматизація першої стадії подрібнення, класифікації та магнітної сепарації. В роботі «Подрібнення. Енергетика і технологія» [13] зроблено ухил в напрямі енергетики рудопідготовки. Зокрема, відмічається, що автоматизовані системи, в яких в якості інформації використовують середню потужність електродвигуна і шум кульового млина, в промисловості широкого використання не знайшли. Тому проблему автоматичної оптимізації рудопідготовки вирішують шляхом компенсації збурних впливів вихідного живлення (що проблематично) або пошуком і обґрунтуванням параметрів, що піддаються автоматичному контролю, які безпосередньо характеризують енергетичну ефективність процесу руйнування матеріалу в барабані млина [13]. Авторами даної роботи запропоновано такий підхід оцінки енергетичної ефективності подрібнення руди в кульовому млині першої стадії [15], однак роботу цих засобів при спрацюванні футеровки ніхто не вивчав. Тому дана публікація передбачає виконати такі дослідження.

Постановка проблеми. Метою даної публікації є математичне моделювання автоматичного контролю, що здійснюється безпосередньо в барабані кульового млина, енергетичної ефективності руйнування руди з отриманням даних для забезпечення необхідної точності в умовах спрацювання футеровки.

Викладення матеріалу та результати. Реалізація енергоефективного подрібнення руди в кульовому млині потребує його контролю безпосередньо в барабані. Такий контроль можливо здійснити, вимірюючи деформацію пружного елемента при ударах куль. Як показано в роботі [16], кульові млини працюють в умовах неперервної зміни стану куль і футеровки, однак можливо створити умови автоматичної стабілізації усталеного режиму найкращого кульового навантаження як за об'ємом, так і за складом. В межах $1/3 \dots 1/2$ довжини барабана від завантаження утворюються широкі зони з практично незмінним розміром куль і осередненим розміром крупних шматків руди. У зовнішньому шарі біля футеровки кулі мають практично незмінний розмір і при падінні у водоспадному

режимі вони наносять удар по барабану, де знаходиться пульпа з крупними шматками руди. Дану задачу можливо розглянути на математичній моделі, яка описує деформацію пружного елемента при ударі кулі в центральній частині. Деформацію центра пружного елемента можливо визначити відповідно залежності

$$x_1 = \frac{mg + \sqrt{mg(mg + 2ch)}}{c}, \quad (1)$$

де m – маса падаючої кулі; g – прискорення земного тяжіння; c – жорсткість пружного елемента; h – висота, з якої падає куля – еквівалент її швидкості руху.

Якщо між футеровкою барабана і падаючою кулею є крупні шматки руди, залежність (1) прийме вигляд

$$x_2 = \frac{mg + \sqrt{mg(mg + 2ch) - 2ckk_1V_p}}{c}, \quad (2)$$

де k – коефіцієнт пропорційності, що залежить від міцності руди; k_1 – стала, що характеризує зв'язок між загальним і деформованим об'ємом шматка руди; V_p – об'єм шматка руди.

Аналіз показує, що при роботі кульового млина можуть виникати збурні фактори в наслідок неідеальної сегрегації куль і зносу футеровки. При цьому в певних межах може змінюватись діаметр і маса молоткових тіл, які використовуються при вимірюванні, та швидкість їх руху.

Як відомо [17], кількість обертів барабана кульового млина визначається залежністю

$$n = \frac{30\sqrt{g}}{\pi\sqrt{R}} \sqrt{\cos\alpha}, \text{ об/хв.}, \quad (3)$$

де $\pi=3,14$; R – внутрішній радіус барабана млина; α – кут відриву куль зовнішнього шару.

Визначивши з (3) $\sqrt{\cos\alpha}$ і піднісши до квадрату ліву і праву сторони отриманої залежності, запишемо

$$\alpha = \arccos \frac{\pi^2 n^2 R}{900g}. \quad (4)$$

Оскільки при спрацюванні футеровки швидкість обертання барабана не змінюють, то при $n=const$ можливо за (4) визначити кут відриву куль зовнішнього шару від футеровки при різних R .

Швидкість падіння куль зовнішнього шару у точці зіткнення з футеровкою можливо визначити за формулою [17]

$$v_p = v\sqrt{1 + 8\sin^2 \alpha}, \quad (5)$$

де v – швидкість руху куль зовнішнього шару при відриванні від футеровки, яка дорівнює $v = \pi Rn/30$, м/с.

В моделях (1) і (2) швидкість руху куль при зіткненні з футеровкою подана через висоту їх падіння. Еквівалентна висота падіння кулі буде дорівнювати

$$h = v_p^2 / 2g. \quad (6)$$

Розглянуті залежності дозволяють отримати вихідні дані для моделювання процесу за моделями (1) і (2). Промодельємо процес руйнування руди на прикладі кульового млина МШЦ-45-60, у якого внутрішній діаметр з неспрацьованою футеровкою 130..150 мм складає 4,5 м. Вихідні дані для математичного моделювання занесемо в табл.1.

Таблиця 1

Вихідні дані для моделювання процесу руйнування руди в кульовому млині МШЦ-45-60

Величина спрацювання футеровки, мм	Внутрішній радіус барабана R , м	Швидкість руху куль зовнішнього шару при відриві від футеровки v , м/с	Кут відриву куль зовнішнього шару α , град	Швидкість руху куль при зіткненні з футеровкою v_p , м/с	Висота h падіння куль еквівалентна швидкості v_p , м
0	2,25	3,8269	48°25'	8,95489	4,087
50	2,30	3,91	47°15'	9,010	4,139
100	2,35	3,995	46°10'	9,078	4,2

З даних табл.1 видно, що при спрацюванні футеровки зростає швидкість зіткнення куль з нею і, відповідно, висота h в моделях (1) і (2). Зміна висоти падіння кулі відбувається на 2,76%, що складає суттєву систематичну похибку. Вона на результати вимірювання буде впливати по-різному в залежності від розміру кулі.

Результати моделювання впливу спрацювання футеровки на точність контролю енергоефективності подрібнення руди в кульовому млині в залежності від розміру куль показані на рис.1. З рис.1 видно, що похибка контролю енергетичної ефективності подрібнення руди в кульовому млині при неспрацьованій футеровці не залежить від розміру кулі, оскільки вона відсутня і є базовою величиною. Похибка залежить від величини зносу футеровки (криві 1 і 2). Особливо вона велика при малих розмірах куль і стрімко зменшується при зростанні їх діаметра від 40 до 50 мм. При руйнуванні 1 см³ руди кулею діаметром 40 мм відносна похибка відповідно складає 1,9 та практично 4% для спрацювання футеровки на 50 і 100 мм. Розглянуті більш діаметри куль забезпечують похибку 0,85...0,65 та 1,84...1,46% відповідно при тих же спрацюваннях. Дані рис.1 характеризують загальний механізм виникнення похибки

при спрацюванні футеровки. Однак на їх підставі не можливо сформулювати підходи організації такого контролю та мінімізації похибки, яка виникає.

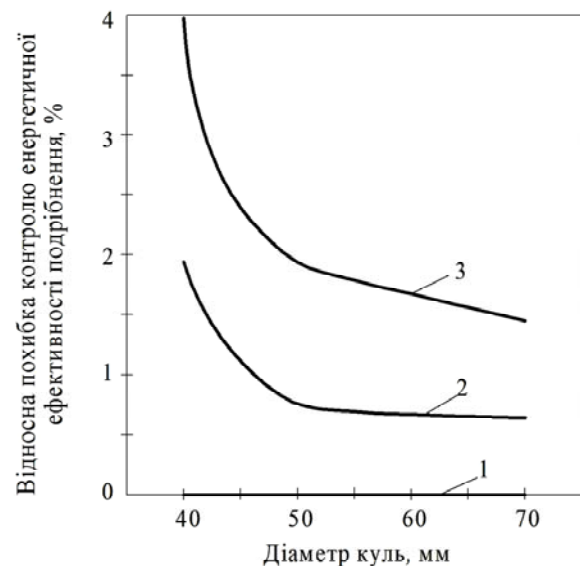


Рис. 1. Залежність відносної похибки контролю енергетичної ефективності подрібнення руди об'ємом 1 см³ від розміру куль при спрацюванні футеровки: 1 – неспрацьована; 2 – спрацьована на 50 мм; 3 – спрацьована на 100 мм.

Перш за все звертає на себе увагу факт, що при відносно малому об'ємі руди і великих розмірах куль похибка зменшується. Максимальний розмір (за висотою) зруйнованого шматка руди можливо визначити відповідно залежності

$$h_{p\max} = R_k \left(1 - \frac{1}{\sqrt{2}} \right), \quad (7)$$

де R_k – радіус кулі.

Залежність (7) дозволяє знайти найбільшу товщину руди, яка буде ефективно подрібнена певною кулею, якщо у неї буде достатньо енергії. Об'єм ефективно подрібненої руди можливо визначити за залежністю

$$V_{pp} = \frac{\pi h_{p\max}^2}{3} (3R_k - h_{p\max}). \quad (8)$$

Результати моделювання процесу подрібнення руди падаючою кулею з використанням залежностей (7) і (8) приведені в табл.2. З даних табл.2 слідує, що максимальна товщина шматка зруйнованої падаючою кулею руди майже вдвічі збільшується при розглянутому зростанні діаметра молольних тіл. Об'єми кульових сегментів зруйнованої руди відрізняються більше ніж у п'ять разів, однак відношення об'ємів зруйнованої руди до об'ємів падаючих куль залишається незмінним. Об'єм зруйнованої руди порівняно з об'ємом падаючої кулі складає незначну частку. Тому цю частку подрібненої руди доцільно розглядати в якості критерію оптимальності подрібнення матеріалу кулями різних діаметрів.

Таблиця 2

Моделювання найбільшого ефективно зруйнованого об'єму матеріалу падаючими кулями різних розмірів

Радіус кулі R_k , см	Об'єм кулі V_k , см ³	Максимальна товщина шматка руди $h_{p\max}$, см	Об'єм зруйнованої руди кульового сегмента V_{kr} , см ³	Відношення об'ємів V_{kr}/V_k відн. одиниць
2,0	33,49336	0,585780	1,944524	0,058057
2,5	65,41672	0,732225	3,797899	0,058057
3,0	113,04009	0,878670	6,562700	0,058057
3,5	179,50318	1,025115	10,421435	0,058057

Другим критерієм оптимальності подрібнення матеріалу кулями різних діаметрів може бути ефективність роботи маси молольного тіла. Результати розрахунку даного показника представлені в табл.3. З даних табл.3 слідує, що ефективність роботи маси куль при подрібненні руди з максимальним значенням товщини шматка матеріалу не залежить від їх діаметра і є достатньо високою – близько 7,4431 см³/кг. Тому визначимо деформації пружного елемента і різниці деформацій незавантаженого і завантаженого пружного елемента при подрібненні руди кулями різних діаметрів при максимальній для них товщині шматків руди. Моделювання проведемо на матеріалі з $k=0,45$ кГм/см³ при $h=4,0872$ м. Результати моделювання приведені на рис.2, з якого видно, що деформація центральної частини пружного елемента (залежність 2) практично лінійно зростає при збільшенні діаметра кулі, яка подрібнює руду максимальної для неї товщини. Практично

така ж залежність (графік 1) отримана для різниці деформацій незавантаженого і завантаженого пружного елемента, однак сигнали тут набагато вищі. Отже, при однакових умовах подрібнення руди різними за діаметром кулями деформації пружного елемента зростають зі збільшенням розміру молольних тіл. В чому ж тут причина? Її можливо уточнити співставленням відхилень пружного елемента і залишкової кінетичної енергії кулі після руйнування руди, яке виконано на рис.3. З нього видно, що деформація пружного елемента майже пропорційно зростає при збільшенні залишкової кінетичної енергії куль. Найбільша залишкова кінетична енергія у крупних молольних тіл. Тому при контролі енергетичної ефективності подрібнення руди кульовими млинами слід по можливості вибирати кулі з більшим діаметром. Виходячи з умов кульового млина МШР-40-50 [16], доцільно вибрати кулі діаметром 50 мм, центр зони яких розташований на відстані

253 см від завантажувальної горловини, а сама зона має ширину 44 см і розташована на середині довжини барабана, де здебільшого встановлюють кришку люка [18]. Це підтверджується також і даними рис.1, оскільки при $d_k=50$ мм відносна похибка контролю енергетичної ефективності подрібнення руди досягає практично мінімального значення, а розмір крупних шматків руди тут значно менший порівняно з їх величиною у завантаженні.

Відносна похибка вимірювання технологічних параметрів у збагачувальній галузі повинна знаходитись в межах $\pm 3\%$ [19]. Для куль $d_k=50$ мм при неспрацьованій футеровці відносна похибка складає 0% при $\Delta x=0,2847$ мм в разі руйнування 1 см^3 матеріалу. Спрацюванню футеровки на 50 мм відповідає деформація $\Delta x=0,2823$ мм, що складає 0,843%. Коли футеровка спрацюється на 100 мм, деформація складе $\Delta x=0,2795$ мм, що відповідає відносній похибці $\delta=1,826\%$. Це буде систематичною похибкою при вимірюванні енергетичної ефективності подрібнення руди.

Вона є недостатньо великою, але і не бажаною, оскільки буде зміщувати результати вимірювання даного технологічного параметра. Тому її необхідно ліквідувати або зменшити.

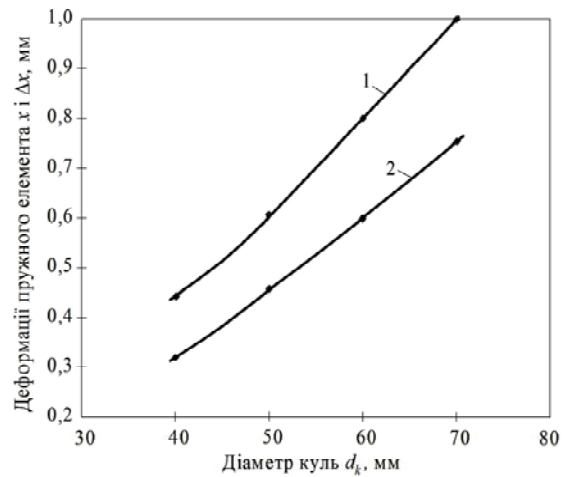


Рис. 2. Залежність деформації пружного елемента при подрібненні руди з максимальною товщиною шматка: 1 – для деформації Δx ; 2 – для деформації x .

Таблиця 3

Дані розрахунку ефективності роботи маси куль при подрібненні руди з максимальним значенням товщини шматка руди

Діаметр кулі d_k , см	Маса кулі m_k , кг	Об'єм зруйнованої руди $V_{кр}$, см^3	Відношення $V_{кр}/m_k$, $\text{см}^3/\text{кг}$
4	0,26125	1,944524	7,4432
5	0,51	3,797899	7,4469
6	0,88117	6,562700	7,4431
7	1,4	10,421435	7,4439

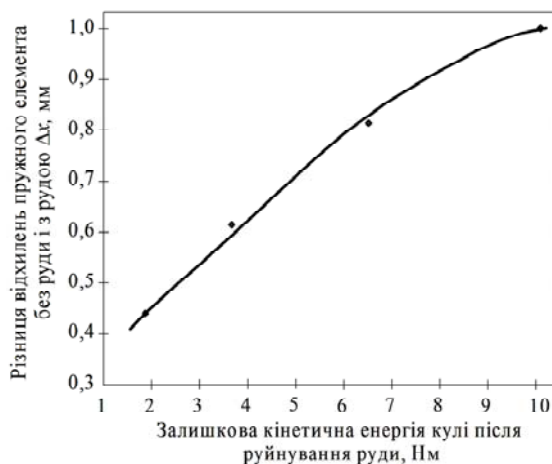


Рис. 3. Залежність різниці відхилень пружного елемента без руди і з рудою від залишкової кінетичної енергії кулі

Похибку, що виникає при спрацюванні футеровки, можливо компенсувати. Для цього необхідно визначити спрацювання футеровки і внести корекцію за результатами вимірювання. Виміряти товщину футеровки кульового млина можливо стержневим перетворювачем, що проходить через всю товщину футеровальної плити. При спрацюванні футеровки однаково буде спрацьовуватись і стержневий перетворювач, зменшуючи свою довжину. Вимірювання можливо здійснити пристроєм зображеним на рис.4, який працює таким чином. При обертанні барабана млина 4 стержневий перетворювач входить в зону

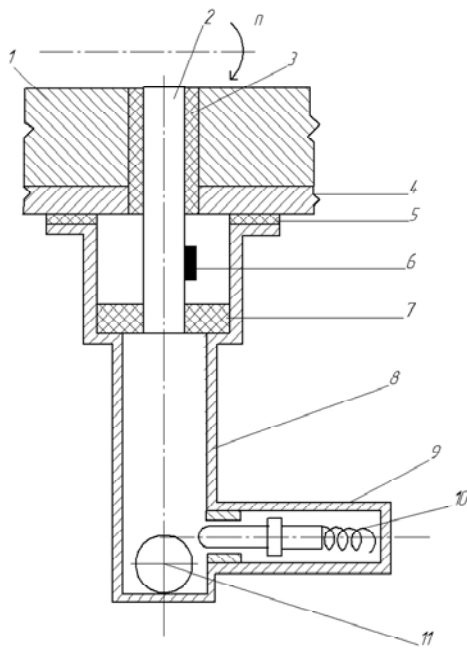


Рис. 4. Схематичне зображення стержневого перетворювача зі збуджувачем коливань:

- 1 – футеровка; 2 – стержневий первинний перетворювач;
- 3 – гідроізоляційне та гнучке амортизаційне кріплення перетворювача; 4 – фрагмент барабана кульового млина;
- 5 – прокладка; 6 – вторинний перетворювач деформації;
- 7 – жорстке амортизаційне кріплення перетворювача;
- 8 – кришка - канал збуджувача коливань; 9 – колпачок;
- 10 – фіксатор кулі; 11 – куля – збуджувач

підстиляючого шару куль, які потім рухаються відповідно коловій траєкторії, а потім відриваються від футеровки. В цій зоні фіксатор 10 утримує кулю 11 у верхньому положенні. При підході стержневого перетворювача в зону, близьку до вертикального стану, фіксатор 10 в наслідок збільшення зусилля з боку кулі 11 спрацьовує і звільняє збуджувач коливань 11. Куля 11, долаючи відцентрову силу, рухається практично вертикально і наносить удар в зовнішній торець стержневого перетворювача 2. Цей удар є еталонним з зусиллям, прикладеним до зовнішнього торця стержневого перетворювача. При цьому збуджена хвиля фіксується вторинним перетворювачем 6 і проходить від зовнішнього торця до вторинного перетворювача, далі проходить до внутрішнього торця відстань l і в зворотному напрямі. Збуджена хвиля спочатку долає відстань u від зовнішнього торця до вторинного перетворювача, далі проходить до внутрішнього торця відстань $l-u$ і повертається, пройшовши ще раз відстань $l-u$ на зворотному

шляху. Проміжок часу t_y від фіксування хвилі вторинним перетворювачем до повторного фіксування хвилі відповідає шляху $2(l-u)$ або шляху $2l_1$, де l_1 – відстань від вторинного перетворювача до кінця стержня, l – повна довжина стержня в момент вимірювання. Швидкість розповсюдження хвилі $v_x = \sqrt{E/\rho}$, де E – модуль Юнга матеріалу стержневого перетворювача, а ρ – його густина. Тоді відстань від вторинного перетворювача до кінця стержня (товщина футеровки) буде дорівнювати

$$l_1 = \frac{t_y}{2} \sqrt{\frac{E}{\rho}}. \quad (9)$$

Оскільки u – це конструктивна стала, то повна довжина стержневого перетворювача буде $l=l_1+u$.

Отже, пристрій (рис.4) дозволяє визначати товщину футеровки в процесі експлуатації кульового млина і вносити корекцію при вимірюванні енергетичної ефективності руйнування матеріалу. Однак це значно ускладнить апаратуру. У даному випадку є більш доцільним зменшення допущеної при спрацюванні футеровки похибки наполовину. Тобто, тарування вимірювального пристрою енергетичної ефективності подрібнення руди здійснити з розрахунком на 50% умовного спрацювання футеровки, що відповідає 50 мм. Тоді при неспрацьованій футеровці систематична похибка складає 0,843%, поступово зменшуючись до нуля при зносі 50 мм. Подальше спрацювання футеровки до 100 мм приведе до зростання похибки від нуля до 0,983%, що суттєво не вплине на результати вимірювання технологічного параметра, а систему контролю ускладнювати не буде.

ВИСНОВКИ

Одним з ефективних шляхів розв'язання проблем, що накопичилися у вітчизняній рудопідготовці на залізорудних збагачувальних фабриках, є удосконалення її автоматизації як створенням новітніх керуючих систем, так і засобів контролю технологічних параметрів, одним з найбільш важливих серед яких є пристрій безпосереднього контролю в барабані кульового млина енергетичної ефективності руйнування матеріалу. Він не досліджувався в умовах експлуатаційного спрацювання футеровки.

Запропонована математична модель такого контролю дозволяє здійснити математичне моделювання процесу.

Встановлено, що швидкість руху куль при зіткненні з футеровкою при її спрацюванні змінюється на 2,7%, що призводить до виникнення похибки контролю енергетичної ефективності руйнування руди. Вона має найбільше значення при малих діаметрах куль близько 40 мм і стрімко зменшується при кулях розміром 50 мм. Показано, що об'єми кульових сегментів зруйнованої руди відрізняються більш ніж у п'ять разів, однак відношення об'ємів зруйнованої руди до об'ємів падаючих куль залишається незмінним. Відношення кульових сегментів зруйнованої руди до маси падаючих куль різної величини складає близько 7,4431 см³/кг. Деформація центральної частини пружного елемента практично лінійно зростає при збільшенні діаметра кулі, яка подрібнює руду максимальної для неї товщини. Практично така ж залежність отримана для різниці деформацій незавантаженого і завантаженого пружного елемента. Це пояснюється збіль-

шенням залишкової кінетичної енергії після руйнування матеріалу при зростанні розміру куль. Найбільш підходящим для контролю енергетичної ефективності руйнування руди є куля розміром 50 мм. При спрацюванні футеровки в млині МШР-40-50 на 50 мм виникає похибка 0,843%, на 100 мм – 1,826%. Є можливість її компенсувати, вимірюючи спрацювання футеровки, однак це сильно ускладнює апаратуру. Тому більш доцільним є здійснення тарування пристрою при умовно спрацьованій наполовину футеровці, коли буде допускатись систематична похибка у два рази менша порівняно з отриманою, що суттєво не вплине на результати вимірювання технологічного параметра.

Перспективною подальших досліджень є практична реалізація засобу автоматичного контролю енергетичної ефективності подрібнення руди кульовими млинами при спрацюванні футеровки.

ЛІТЕРАТУРА

1. Linch A.Dzh. Tsikly drobleniya i izmel'cheniya: [per. s angl.]. – M.: Nedra, 1981. – 342 s.
2. Herbst J.A., Alba F.A., Pate W.T., Oblad A.E. Optimal Control of Communion Operations // International Journal of Mineral Processing. – 1988. – Vol.22. – №1-4. – P. 275-296.
3. Herbst J.A., Pate W.T., Oblad A.E. Model-based control of mineral processing operations // Powder Technology. – 1992. – Vol. 69. – P. 21-32.
4. Maryuta A.N. Avtomaticheskaya optimizatsiya protsesa obogashcheniya rud na magnitobogatitel'nykh fabrikakh. – M.: Nedra, 1987. – 230 s.
5. Morkun V.S., Tsokurenko A.A., Lutsenko I.A. Adaptivnye sistemy optimal'nogo upravleniya tekhnologicheskimi protsessami. – Krivoi Rog: Mineral, 2005. – 261 s.
6. Morkun V.S., Morkun N.V., Tron' V.V. Formirovanie robastnogo avtomatizirovannogo upravleniya zamknutym tsiklom izmel'cheniya na osnove ∞ -normy // Gimichii visnik: nauk.-tekhn. zb. DVNZ «KNU». – 2014. – Vip. 98. – S. 83-85.
7. Kupin A.I. Intel'ektual'na identifikatsiya ta keruvannya v umovakh protsesiv zbagachuval'noi tekhnologii. – Krivii Rig: Vid-vo KTU, 2008. – 204 s.
8. Kochura E.V., Faris Samir Rasmi Al'khorī. Modelirovanie protsesa raskrytiya rudy v sharovoi mel'nitse s pozitsii zadach upravleniya // Naukovii visnik Natsional'nogo gimichogo universitetu. – 2013. – №1. – S.102-107.
9. Porkujan Ol'ga Viktorivna. Keruvannya nelinejnyimi dynamichnyimi ob'ektamy zbagachuval'nyh vyrobnyctv na osnovi gibrydnyh modelej Gamershtejna: avtorefer. dys. na zdobuttja nauk. stupenja dokt. tehn. nauk: spec. 05.13.07 «Avtomatyzacija procesiv keruvannya». – Kryvyj Rig, 2009. – 36 s.
10. Morkun Natalja Volodymyrivna. Rozpodilene optymal'ne keruvannya vzajemopov'jazanyimi procesami zbagachuval'noho vyrobnyctva na osnovi dynamichnoi' prostorovo-chasovoi' modeli: dys. . . . d-r. tehn. nauk: spec. 05.13.07 "Avtomatyzacija procesiv keruvannya". – Kryvyj Rig, 2017. – 357 s.
11. Azaryan A.A., Krivenko Yu.Yu., Kucher V.G. Avtomatizatsiya pervoi stadii izmel'cheniya, klassifikatsii i magnitnoi separatsii – real'nyi put' povysheniya effektivnosti obogashcheniya zheleznykh rud // Visnik Krivoriz'kogo natsional'nogo universitetu: zb. nauk. prats'. – 2014. – Vip. 36. – S. 276-280.
12. Tron' V.V., Maevs'kii K.V. Formuvannya adaptivnogo keruvannya protsesom podribnennya zalozrudnoi' sirovini v umovakh neviznachenosti kharakteristik ob'ekta // Gimichii visnik: nauk.-tekhn. zb. DVNZ «KNU». – 2015. – Vip. 99. – S. 27-32.
13. Pivnyak G.G., Vaisberg L.A., Kirichenko V.I., Pilov P.I., Kirichenko V.V. Izmel'chenie. Energetika i tekhnologiya. – M.: Izd. dom "Ruda i Metally", 2007. – 296 s.
14. Morozov V.V., Topchaev V.P., Ulitenko K.Ya., Ganbaatar Z., Delgerbat L. Razrabotka i primeneniye avtomatizirovannykh sistem upravleniya protsessami obogashcheniya poleznykh iskopaemykh. – M.: Izd. dom «Ruda i Metally», 2013. – 512 s.
15. Kondratec' V.O., Karchev'ska M.O. Identyfikatsiya zavantazhenija kul'ovogo mlyna rudoju po parametram energetychnoi' efektyvnosti rujnuvannja materialu // Visnyk Vinnyc'kogo politekhnichnogo instytutu. – 2011. – №5. – S. 9-13.
16. Kondratec' V.O., Reva O.M., Karchev'ska M.O. Teoretyчне doslidzhennja ustalenyh i perehidnyh rezhymiv roboty kul' ta futerovky v mlynah // Tehnika v sil'skogospodars'komu vyrobnyctvi, galuzeve mashynobuduvannja, avtomatyzacija: zb. nauk. prac' KNTU. – 2008. – Vyp.21. – S.187-196.
17. Andreev S.E., Perov V.A., Zverevich V.V. Droblenie, izmel'chenie i grokhochenie poleznykh iskopaemykh. – M.: Nedra, 1980. – 415 s.
18. Kryukov D.K. Futerovki sharovykh mel'nits. – M.: Mashinostroenie, 1965. – 183 s.
19. Nesterov G.S., Nesterova N.A., Batanov A.I. Upravlenie tekhnologicheskimi protsessami na obogatitel'nykh fabrikakh. – M.: Nedra, 1966. – 220 s.

Рецензент: д.т.н., проф. Рудакова Г.В.,
Херсонський національний технічний університет



CONSTRUCTING THE NON-LINEAR REGRESSION EQUATION TO ESTIMATE THE SOFTWARE SIZE OF OPEN SOURCE PHP-BASED INFORMATION SYSTEMS

UDC 004.412:519.237.5

Sergiy PRYKHODKO

Dr.Sc., Professor at the Department of Software of Automated Systems,
the Head of Department, Admiral Makarov National University of Shipbuilding.

Scientific interests: mathematical modeling of random variables and processes in information technologies.

Natalia PRYKHODKO

PhD, Associate Professor at the Finance Department, Admiral Makarov National University of Shipbuilding.

Scientific interests: mathematical modeling of random variables and processes in information technologies.

Tatyana SMYKODUB

Senior Teacher at the Department of Software of Automated Systems, Admiral Makarov National University of Shipbuilding.

Scientific interests: mathematical modeling of random variables and processes in information technologies.

Alexander SPINOV

a student of master's degree program in specialty 121 "Software Engineering",
the Department of Software of Automated Systems, Admiral Makarov National University of Shipbuilding.

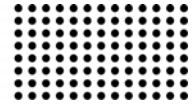
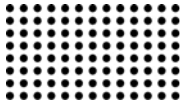
Scientific interests: mathematical modeling of random variables in information technologies.

INTRODUCTION

PHP is a free programming language used primarily in information systems and web applications. Software size is one of the most important internal metrics of software including software of PHP-based open-source information systems. The information obtained from estimating the software size is useful for predicting the software development effort by such models as COCOMO 81, COCOMO II and COCOMO 2000. The papers [1, 2] proposed the linear regression equations for estimating the software size of some programming languages, such as VBA, PHP, Java and C++. The proposed equations are constructed by multiple linear regression analysis on the basis of the metrics that can be measured from class diagram. However, there are four basic assumptions that justify the use of linear regression models, one of which is normality of the error distribution. But this assumption is valid only in particular cases. This leads to the need to use the non-linear regression equa-

tions including for estimating the software size of PHP-based open-source information systems.

A normalizing transformation is often a good way to build the equations, confidence and prediction intervals of multiply non-linear regressions [3-8]. According to [4] transformations are used for essentially four purposes, two of which are: first, to obtain approximate normality for the distribution of the error term (residuals), second, to transform the response and/or the predictor in such a way that the strength of the linear relationship between new variables (normalized variables) is better than the linear relationship between dependent and independent random variables. Well-known techniques for building the equations, confidence and prediction intervals of multivariate non-linear regressions are based on the univariate normalizing transformations, which do not take into account the correlation between random variables in the case of normalization of multivariate non-Gaussian data. This leads to



the need to use the multivariate normalizing transformations.

The goal of the article is to construct the non-linear regression equation for estimating the software size of open-source PHP-based information systems. The software size prediction results by constructed equation should be better in comparison with other regression equations, both linear and nonlinear, primarily on such standard evaluations as the multiple coefficient of determination and mean magnitude of relative error.

In this article, we build the equation, confidence and prediction intervals of multivariate non-linear regression for estimating the software size of open-source PHP-based systems on the basis of the Johnson multivariate normalizing transformation (the Johnson normalizing translation) with the help of appropriate techniques proposed in [8, 9].

The techniques. The techniques to build the equations, confidence and prediction intervals of non-linear regressions are based on the multiple non-linear regression analysis using the multivariate normalizing transformations. A multivariate normalizing transformation of non-Gaussian random vector $\mathbf{P} = \{Y, X_1, X_2, \dots, X_k\}^T$ to Gaussian random vector $\mathbf{T} = \{Z_Y, Z_1, Z_2, \dots, Z_k\}^T$ is given by

$$\mathbf{T} = \boldsymbol{\psi}(\mathbf{P}) \tag{1}$$

and the inverse transformation for (1)

$$\mathbf{P} = \boldsymbol{\psi}^{-1}(\mathbf{T}). \tag{2}$$

The linear regression equation for normalized data according to (1) will have the form [4]

$$\hat{Z}_Y = \bar{Z}_Y + (\mathbf{Z}_X^+)^T \hat{\mathbf{b}}, \tag{3}$$

where \hat{Z}_Y is prediction linear regression equation result for values of components of vector $\mathbf{z}_X = \{Z_1, Z_2, \dots, Z_k\}$; \mathbf{Z}_X^+ is the matrix of centered regressors that contains the values $Z_1 - \bar{Z}_1, Z_2 - \bar{Z}_2,$

$\dots, Z_k - \bar{Z}_k$; $\hat{\mathbf{b}}$ is estimator for vector of linear regression equation parameters, $\mathbf{b} = \{b_1, b_2, \dots, b_k\}^T$.

The non-linear regression equation will have the form

$$\hat{Y} = \psi_Y^{-1} \left[\bar{Z}_Y + (\mathbf{Z}_X^+)^T \hat{\mathbf{b}} \right]. \tag{4}$$

where ψ_Y is the first component of vector $\boldsymbol{\psi} = \{\psi_Y, \psi_1, \psi_2, \dots, \psi_k\}^T$.

The technique to build a confidence interval of non-linear regression is based on transformations (1) and (2), equation (3) and a confidence interval of linear regression for normalized data

$$\hat{Z}_Y \pm t_{\alpha/2, v} S_{Z_Y} \left\{ \frac{1}{N} + (\mathbf{z}_X^+)^T \left[(\mathbf{Z}_X^+)^T \mathbf{Z}_X^+ \right]^{-1} (\mathbf{z}_X^+) \right\}^{1/2}. \tag{5}$$

where $t_{\alpha/2, v}$ is a quantile of student's t -distribution with v degrees of freedom and $\alpha/2$ significance level; $(\mathbf{z}_X^+)^T$ is one of the rows of \mathbf{Z}_X^+ ; $S_{Z_Y}^2 = \frac{1}{v} \sum_{i=1}^N (Z_{Y_i} - \hat{Z}_{Y_i})^2$, $v = N - k - 1$; $(\mathbf{Z}_X^+)^T \mathbf{Z}_X^+$ is the $k \times k$ matrix

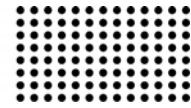
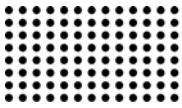
$$(\mathbf{Z}_X^+)^T \mathbf{Z}_X^+ = \begin{pmatrix} S_{Z_1 Z_1} & S_{Z_1 Z_2} & \dots & S_{Z_1 Z_k} \\ S_{Z_2 Z_1} & S_{Z_2 Z_2} & \dots & S_{Z_2 Z_k} \\ \dots & \dots & \dots & \dots \\ S_{Z_k Z_1} & S_{Z_k Z_2} & \dots & S_{Z_k Z_k} \end{pmatrix},$$

where $S_{Z_q Z_r} = \sum_{i=1}^N [Z_{q_i} - \bar{Z}_q][Z_{r_i} - \bar{Z}_r]$, $q, r = 1, 2, \dots, k$.

The confidence interval for non-linear regression is built on the basis of the interval (5) and inverse transformation (2)

$$\psi_Y^{-1} \left(\hat{Z}_Y \pm t_{\alpha/2, v} S_{Z_Y} \left\{ \frac{1}{N} + (\mathbf{z}_X^+)^T \left[(\mathbf{Z}_X^+)^T \mathbf{Z}_X^+ \right]^{-1} (\mathbf{z}_X^+) \right\}^{1/2} \right). \tag{6}$$

The technique to build a prediction interval is based on multivariate transformation (1), the inverse transformation



(2), linear regression equation for normalized data (3) and a prediction interval for normalized data

$$\hat{Z}_Y \pm t_{\alpha/2, \nu} S_{Z_Y} \left\{ 1 + \frac{1}{N} + (\mathbf{z}_X^+)^T \left[(\mathbf{Z}_X^+)^T \mathbf{Z}_X^+ \right]^{-1} (\mathbf{z}_X^+) \right\}^{1/2}. \quad (7)$$

The prediction interval for non-linear regression is built on the basis of the interval (7) and inverse transformation (2)

$$\Psi_Y^{-1} \left(\hat{Z}_Y \pm t_{\alpha/2, \nu} S_{Z_Y} \left\{ 1 + \frac{1}{N} + (\mathbf{z}_X^+)^T \left[(\mathbf{Z}_X^+)^T \mathbf{Z}_X^+ \right]^{-1} (\mathbf{z}_X^+) \right\}^{1/2} \right). \quad (8)$$

The Johnson normalizing translation. For normalizing the multivariate non-Gaussian data, we use the Johnson translation system. In our case the Johnson normalizing translation is given by [10]

$$\mathbf{T} = \boldsymbol{\gamma} + \boldsymbol{\eta} \mathbf{h} \left[\boldsymbol{\lambda}^{-1} (\mathbf{P} - \boldsymbol{\varphi}) \right] \sim N_m(\mathbf{0}_m, \boldsymbol{\Sigma}), \quad (9)$$

where $\boldsymbol{\Sigma}$ is the covariance matrix; $m = k + 1$; $\boldsymbol{\gamma}$, $\boldsymbol{\eta}$, $\boldsymbol{\varphi}$ and $\boldsymbol{\lambda}$ are parameters of translation (9); $\boldsymbol{\gamma} = (\gamma_Y, \gamma_1, \gamma_2, \dots, \gamma_k)^T$; $\boldsymbol{\eta} = \text{diag}(\eta_Y, \eta_1, \eta_2, \dots, \eta_k)$; $\boldsymbol{\lambda} = \text{diag}(\lambda_Y, \lambda_1, \lambda_2, \dots, \lambda_k)$; $\boldsymbol{\varphi} = (\varphi_Y, \varphi_1, \varphi_2, \dots, \varphi_k)^T$; $\mathbf{h}[(y_Y, y_1, \dots, y_k)] = \{h_Y(y_Y), h_1(y_1), \dots, h_k(y_k)\}^T$; $h_i(\cdot)$ is one of the translation functions

$$\mathbf{h} = \begin{cases} \ln(y), & \text{for } S_L \text{ (log normal) family;} \\ \ln[y/(1-y)], & \text{for } S_B \text{ (bounded) family;} \\ \text{Arsh}(y), & \text{for } S_U \text{ (unbounded) family;} \\ y & \text{for } S_N \text{ (normal) family.} \end{cases} \quad (10)$$

Here $y = (X - \varphi)/\lambda$; $\text{Arsh}(y) = \ln\left(y + \sqrt{y^2 + 1}\right)$. In our case X equals Y , X_1 , X_2 or X_3 respectively.

The equation, confidence and prediction intervals of non-linear regression to estimate the software size of open-source PHP-based systems. The equation, confidence and prediction intervals of non-linear regression to estimate the software size of open-source PHP-based systems are constructed on the basis of the

Johnson multivariate normalizing transformation for the four-dimensional non-Gaussian data set: actual software size in the thousand lines of code (KLOC) Y , the average number of attributes per class X_3 , the total number of classes X_1 and the total number of relationships X_2 in conceptual data model from 32 information systems developed using the PHP programming language with HTML and SQL. Table I contains the data from [1] on four metrics of software for 32 open-source PHP-based systems.

For detecting the outliers in the data from Table 1 we use the technique based on multivariate normalizing transformations and the squared Mahalanobis distance [11]. There are no outliers in the data from Table I for 0.005 significance level and the Johnson multivariate transformation (9) for S_B family. The same result was obtained in [12] for the transformation (9) for S_U family. In [1] it was also assumed that the data contains no outliers. Although note that without using normalization, the data of system 11 is multivariate outlier, since for this data row the squared Mahalanobis distance equals to 15.44 is greater than the value of the quantile of the Chi-square distribution, which equals to 14.86 for 0.005 significance level.

Parameters of the multivariate transformation (9) for S_B family were estimated by the maximum likelihood method. Estimators for parameters of the transformation (9) are: $\hat{\gamma}_Y = 9.63091$, $\hat{\gamma}_1 = 15.5355$, $\hat{\gamma}_2 = 25.4294$, $\hat{\gamma}_3 = 0.72801$, $\hat{\eta}_Y = 1.05243$, $\hat{\eta}_1 = 1.58306$, $\hat{\eta}_2 = 2.54714$, $\hat{\eta}_3 = 0.54312$, $\hat{\varphi}_Y = -1.4568$, $\hat{\varphi}_1 = -1.8884$, $\hat{\varphi}_2 = -6.9746$, $\hat{\varphi}_3 = 3.2925$, $\hat{\lambda}_Y = 153102.605$, $\hat{\lambda}_1 = 243051.0$, $\hat{\lambda}_2 = 311229.5$ and $\hat{\lambda}_3 = 13.900$. The sample covariance matrix S_N of the \mathbf{T} is used as the approximate moment-matching estimator of $\boldsymbol{\Sigma}$

$$S_N = \begin{pmatrix} 1.0000 & 0.9514 & 0.9333 & 0.1574 \\ 0.9514 & 1.0000 & 0.9006 & 0.1345 \\ 0.9333 & 0.9006 & 1.0000 & 0.0554 \\ 0.1574 & 0.1345 & 0.0554 & 1.0000 \end{pmatrix}.$$

After normalizing the non-Gaussian data by the multivariate transformation (9) for S_B family the linear regression equation (3) is built for normalized data

$$\hat{Z}_Y = \hat{b}_0 + \hat{b}_1 Z_1 + \hat{b}_2 Z_2 + \hat{b}_3 Z_3. \quad (11)$$

Parameters of the linear regression equation (11) were estimated by the least square method. Estimators for parameters of the equation (11) are such: $\hat{b}_0 = 1.02 \cdot 10^{-5}$, $\hat{b}_1 = 0.56085$, $\hat{b}_2 = 0.42491$, $\hat{b}_3 = 0.05846$.

Parameters of the linear regression equation (11) were estimated by the least square method. Estimators for pa-

Table I

The data and prediction result by regression equations for 32 open-source PHP-based systems

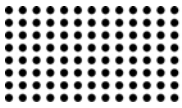
No	Y	X ₁	X ₂	X ₃	Linear regression		Non-linear regression			
					\hat{Y}	RME	univariate transformation		multivariate transformation	
							\hat{Y}	RME	\hat{Y}	RME
1	3.038	5	2	10.6	3.237	0.0656	4.675	0.5388	4.550	0.4976
2	22.599	17	7	7	24.142	0.0683	19.965	0.1166	19.990	0.1154
3	32.243	21	13	4.524	37.524	0.1638	32.098	0.0045	33.535	0.0401
4	16.164	13	11	7.077	25.916	0.6033	23.171	0.4335	21.292	0.3173
5	83.862	35	24	6.571	74.624	0.1102	80.265	0.0429	83.618	0.0029
6	24.22	13	9	8.077	23.224	0.0411	20.524	0.1526	18.901	0.2196
7	63.929	35	19	8.029	67.215	0.0514	65.913	0.0310	70.647	0.1051
8	2.543	5	3	9.4	4.127	0.6228	5.789	1.2764	5.169	1.0328
9	6.697	5	5	7	5.906	0.1181	7.353	0.0980	6.356	0.0509
10	55.537	25	14	8.64	46.843	0.1565	42.098	0.2420	43.126	0.2235
11	55.752	39	10	9.077	57.814	0.0370	67.070	0.2030	49.823	0.1064
12	62.602	30	17	7	56.995	0.0896	53.497	0.1454	56.651	0.0951
13	67.111	23	22	14.957	61.856	0.0783	65.500	0.0240	60.617	0.0968
14	2.552	3	1	8.333	-2.395	1.9384	2.202	0.1370	2.447	0.0412
15	12.17	10	5	3.7	9.959	0.1816	9.693	0.2035	10.029	0.1759
16	12.757	13	9	5	21.218	0.6632	18.682	0.4644	18.105	0.4192
17	5.695	7	3	8.429	5.976	0.0493	7.083	0.2438	6.687	0.1743
18	7.744	9	6	9.222	13.991	0.8067	12.911	0.6673	11.301	0.4593
19	7.514	4	1	8	-1.371	1.1825	2.496	0.6678	3.096	0.5880
20	11.054	9	9	3.667	15.385	0.3918	13.301	0.2032	12.850	0.1625
21	29.77	17	15	3.412	35.179	0.1817	27.321	0.0823	29.061	0.0238
22	11.653	9	8	8.778	17.045	0.4627	15.461	0.3268	13.268	0.1386
23	6.847	5	4	3.6	2.017	0.7054	5.435	0.2062	5.112	0.2534
24	13.389	7	5	11.714	11.462	0.1440	10.367	0.2257	8.661	0.3531
25	14.45	12	6	16.583	22.513	0.5580	20.191	0.3973	15.888	0.0995
26	4.414	6	3	3.667	1.630	0.6307	5.318	0.2048	5.260	0.1916
27	2.102	3	1	3.333	-5.655	3.6902	2.142	0.0192	1.873	0.1090
28	42.819	20	18	3.5	43.975	0.0270	37.967	0.1133	38.631	0.0978
29	4.077	4	2	9	0.953	0.7662	3.892	0.0454	3.732	0.0846
30	57.408	33	14	9.242	57.164	0.0043	53.121	0.0747	54.381	0.0527
31	7.428	7	3	7	5.044	0.3209	6.861	0.0764	6.571	0.1154
32	8.947	15	5	4	16.360	0.8285	12.934	0.4456	14.258	0.5936

After that the non-linear regression equation (4) is built where \hat{Z}_Y is prediction result by the equation (11),

$$\hat{Y} = \hat{\phi}_Y + \hat{\lambda}_Y \left[1 + e^{-\frac{(\hat{Z}_Y - \hat{\gamma}_Y)}{\hat{\eta}_Y}} \right]^{-1}. \quad (12)$$

$$Z_j = \gamma_j + \eta_j \ln \frac{X_j - \phi_j}{\phi_j + \lambda_j - X_j}, \quad \phi_j < X_j < \phi_j + \lambda_j,$$

$j = 1, 2, 3.$



The prediction results by equation (12) for values of components of vector $\mathbf{X} = \{X_1, X_2, X_3\}$ from Table I and values of magnitude of relative error MRE are shown in the Table I for two cases: the Johnson univariate and multivariate normalizing transformations. Table I also contains the prediction results by linear regression equation from [1] for values of components of vector \mathbf{X} from Table I and MRE values. Note the prediction results by linear regression equation from [1] are negative for the three rows of data: 14, 19 and 27. All prediction results by non-linear regression equation (12) are positive.

For univariate normalizing transformations (10) of S_B family the estimators for parameters are such: $\hat{\gamma}_Y = 0.77502$, $\hat{\gamma}_1 = 0.59473$, $\hat{\gamma}_2 = 0.57140$, $\hat{\gamma}_3 = 0.68734$, $\hat{\eta}_Y = 0.44395$, $\hat{\eta}_1 = 0.48171$, $\hat{\eta}_2 = 0.49553$, $\hat{\eta}_3 = 0.51970$, $\hat{\phi}_Y = 2.063$, $\hat{\phi}_1 = 2.900$, $\hat{\phi}_2 = 0.900$, $\hat{\phi}_3 = 3.304$, $\hat{\lambda}_Y = 83.059$, $\hat{\lambda}_1 = 36.695$, $\hat{\lambda}_2 = 23.525$ and $\hat{\lambda}_3 = 13.660$. In the case of univariate normalizing transformations the estimators for parameters of the equation (11) are such: $\hat{b}_0 = 3.11 \cdot 10^{-7}$, $\hat{b}_1 = 0.43519$, $\hat{b}_2 = 0.52239$ and $\hat{b}_3 = 0.08546$.

Also the non-linear regression equation (4) is built by the decimal logarithm transformation

$$\hat{Y} = 10^{b_0} X_1^{b_1} X_2^{b_2} X_3^{b_3}, \quad (13)$$

where the estimators for parameters of the equation (13) are such: $\hat{b}_0 = -0.26161$, $\hat{b}_1 = 0.99151$, $\hat{b}_2 = 0.33232$ and $\hat{b}_3 = 0.13777$.

The values of multiple coefficient of determination R^2 , mean magnitude of relative error MMRE and percentage of prediction PRED(0.25) equal respectively 0.9491, 0.4919 and 0.5313 for linear regression equation from [1], and equal respectively 0.9375, 0.2455 and 0.625 for the equation (13). The values of R^2 , MMRE and PRED(0.25) are better for the equation (12), in comparison with both the equation from [1] and equation (13), and are 0.9692, 0.2199 and 0.7188 for the Johnson multivariate transformation and equal 0.9591, 0.2535 and 0.7188 for the Johnson univariate transformation respectively. The acceptable values of MMRE and

PRED(0.25) are not more than 0.25 and not less than 0.75 respectively. The values of MMRE indicate that only the values for equation (12) on the basis of the Johnson multivariate normalizing transformation and the decimal logarithm transformation are less than 0.25. Although all values of PRED(0.25) are less than 0.75 nevertheless the values are greater for equation (12) with estimators of parameters for the Johnson transformations, both multivariate and univariate.

The confidence and prediction intervals of non-linear regression are defined by (6) and (8) respectively for the data from Table I. Table II contains the lower (LB) and upper (UB) bounds of the confidence intervals of linear and non-linear regressions on the basis of univariate and multivariate transformations respectively for 0.05 significance level. Note the lower bounds of the confidence interval of linear regression from [1] are negative for the seven rows of data: 1, 14, 19, 23, 26, 27 and 29. The upper bound for the data row 27 is negative too. All the lower and upper bounds of the confidence interval of non-linear regressions are positive. The widths of the confidence interval of non-linear regression on the basis of the Johnson multivariate transformation are less than for linear regression from [1] for the twenty rows of data: 1, 6, 8, 9, 14-20, 22-27, 29, 31 and 32. Also the widths of the confidence interval of non-linear regression on the basis of the Johnson multivariate transformation are less for more data rows than for non-linear regressions following the univariate transformations, both decimal logarithm and the Johnson. The widths of the confidence interval of non-linear regression on the basis of the Johnson multivariate transformation are less than following the decimal logarithm univariate transformation for the twenty-seven rows of data: 1-4, 6-12, 15-26, 28, 30-32. And ones are less than following the Johnson univariate transformation for the twenty-five rows of data: 1-4, 6, 8-11, 15-18, 20-26, 28-32. Approximately the same results are obtained for the prediction intervals of regressions.

Table III contains the lower (LB) and upper (UB) bounds of the prediction intervals of linear and non-linear regressions on the basis of univariate and multivariate transformations respectively for 0.05 significance level. Note the lower bounds of the prediction interval of linear regression from [1] are negative for the thirteen

rows of data: 1, 8, 9, 14, 15, 17, 19, 23, 24, 26, 27, 29, 31. All the lower bounds of the prediction interval of non-linear regressions are positive. The widths of the prediction interval of non-linear regression on the basis of the Johnson multivariate transformation are less than for linear regression from [1] for the twenty rows of data: 1, 6, 8, 9, 14-20, 22-27, 29, 31 and 32. Also the widths of the prediction interval of non-linear regression on the basis of the Johnson multivariate transformation are less for more data rows than for non-linear regressions fol-

lowing the univariate transformations, both decimal logarithm and the Johnson. The widths of the prediction interval of non-linear regression on the basis of the Johnson multivariate transformation are less than following the decimal logarithm univariate transformation for the twenty-nine rows of data: 1-13, 15-18, 20-26, 28-32. And ones are less than following the Johnson univariate transformation for the twenty-three rows of data: 1-4, 6, 8-10, 15-18, 20-26, 28, 29, 31 and 32.

Table II

Bounds of the confidence intervals

No	Y	Bounds for linear regression		Bounds for non-linear regressions					
				univariate transformations				Johnson multivariate transformation	
		LB	UB	decimal logarithm		Johnson			
1	3.038	-0.402	6.877	3.725	5.947	3.673	6.267	3.655	5.601
2	22.599	21.413	26.871	18.933	27.172	15.473	25.455	16.856	23.660
3	32.243	34.344	40.704	26.415	39.621	24.791	40.266	29.157	38.539
4	16.164	23.172	28.660	16.855	24.285	17.982	29.365	18.542	24.421
5	83.862	69.187	80.062	55.076	87.173	74.107	83.078	68.095	102.603
6	24.22	21.015	25.433	16.557	22.438	16.129	25.819	16.935	21.075
7	63.929	62.690	71.740	52.309	83.045	56.434	72.961	59.182	84.277
8	2.543	1.013	7.241	4.288	6.544	4.484	7.748	4.260	6.223
9	6.697	3.084	8.728	4.569	7.951	5.456	10.203	5.135	7.803
10	55.537	43.863	49.824	35.573	52.195	33.947	50.366	37.344	49.768
11	55.752	49.560	66.068	41.448	87.698	42.891	79.359	36.261	68.257
12	62.602	53.265	60.725	43.512	65.766	44.275	61.787	48.298	66.405
13	67.111	54.146	69.566	35.747	69.156	49.897	75.572	45.429	80.723
14	2.552	-5.673	0.883	1.639	2.897	2.125	2.375	1.806	3.214
15	12.17	6.609	13.309	8.838	13.632	6.979	13.684	8.320	12.037
16	12.757	18.574	23.862	15.339	21.222	14.673	23.576	16.253	20.151
17	5.695	3.165	8.787	6.139	8.644	5.548	9.233	5.655	7.870
18	7.744	11.381	16.601	10.039	14.139	9.902	16.849	9.960	12.799
19	7.514	-4.587	1.845	2.106	3.945	2.253	3.046	2.391	3.930
20	11.054	11.684	19.085	9.137	15.776	9.186	19.255	10.550	15.591
21	29.77	30.767	39.591	20.246	34.593	16.796	41.072	22.899	36.782
22	11.653	14.250	19.840	10.430	16.253	11.581	20.525	11.347	15.478
23	6.847	-1.579	5.613	3.900	6.687	4.071	7.662	4.063	6.360
24	13.389	7.648	15.276	7.099	11.493	7.462	14.583	7.262	10.286
25	14.45	16.199	28.828	13.006	22.695	10.971	34.746	12.197	20.576
26	4.414	-1.967	5.227	4.421	7.029	4.083	7.261	4.241	6.461
27	2.102	-9.730	-1.580	1.360	2.712	2.092	2.281	1.087	2.902
28	42.819	38.873	49.077	25.212	43.588	25.181	51.940	30.491	48.845
29	4.077	-2.236	4.142	2.947	4.616	3.177	5.048	2.959	4.640
30	57.408	52.335	61.993	44.400	73.879	41.599	63.278	44.841	65.885
31	7.428	2.314	7.774	6.042	8.344	5.463	8.784	5.585	7.695
32	8.947	12.515	20.205	12.503	22.001	9.080	18.449	11.119	18.180

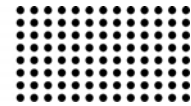
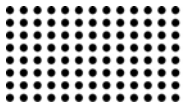


Table III

Bounds of the prediction intervals

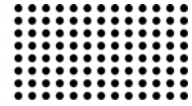
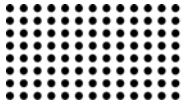
No	Y	Bounds for linear regression		Bounds for non-linear regressions					
				univariate transformations				Johnson multivariate transformation	
		decimal logarithm		Johnson					
		LB	UB	LB	UB	LB	UB	LB	UB
1	3.038	-8.886	15.361	2.264	9.787	2.507	15.664	2.053	8.822
2	22.599	12.260	36.024	11.075	46.451	5.800	53.204	11.088	35.207
3	32.243	25.530	49.517	15.704	66.644	9.341	65.987	19.149	57.962
4	16.164	14.031	37.802	9.874	41.455	6.642	57.342	11.955	37.129
5	83.862	61.845	87.403	33.367	143.886	59.920	84.392	47.603	146.045
6	24.22	11.451	34.998	9.475	39.211	5.956	53.906	10.617	32.866
7	63.929	54.797	79.633	31.724	136.931	31.210	81.247	40.528	122.355
8	2.543	-7.849	16.103	2.565	10.940	2.713	20.215	2.431	9.838
9	6.697	-5.998	17.810	2.856	12.722	2.996	26.099	3.097	11.949
10	55.537	34.901	58.785	20.980	88.499	13.397	72.304	24.761	74.346
11	55.752	43.606	72.022	27.405	132.638	26.251	82.571	26.759	91.726
12	62.602	44.844	69.146	25.940	110.319	19.861	77.358	32.563	97.782
13	67.111	47.957	75.755	23.063	107.190	28.542	81.562	33.153	109.857
14	2.552	-14.415	9.625	1.030	4.613	2.084	2.994	0.811	5.262
15	12.17	-2.080	21.999	5.307	22.704	3.441	33.425	5.255	18.197
16	12.757	9.355	33.081	8.849	36.788	5.492	51.258	10.150	31.513
17	5.695	-5.925	17.877	3.565	14.884	2.964	24.822	3.336	12.381
18	7.744	2.136	25.846	5.831	24.343	4.145	40.894	6.095	20.095
19	7.514	-13.374	10.632	1.346	6.172	2.127	4.916	1.198	6.351
20	11.054	3.243	27.527	5.697	25.303	4.154	42.480	6.867	23.133
21	29.77	22.801	47.556	12.581	55.670	7.324	63.400	15.978	51.960
22	11.653	5.148	28.943	6.285	26.973	4.693	46.152	7.200	23.590
23	6.847	-10.093	14.128	2.426	10.749	2.635	19.103	2.367	9.829
24	13.389	-0.715	23.638	4.334	18.826	3.576	35.238	4.477	15.796
25	14.45	9.337	35.689	8.136	36.280	5.323	56.560	8.396	29.076
26	4.414	-10.481	13.741	2.683	11.585	2.621	18.450	2.464	10.048
27	2.102	-17.916	6.606	0.885	4.168	2.073	2.648	0.410	4.484
28	42.819	31.335	56.615	15.725	69.883	10.895	70.978	21.432	68.748
29	4.077	-11.043	12.949	1.779	7.647	2.371	12.014	1.575	7.423
30	57.408	44.632	69.696	27.354	119.916	19.170	77.441	30.902	94.883
31	7.428	-6.838	16.926	3.483	14.475	2.926	23.959	3.273	12.168
32	8.947	4.173	28.547	7.842	35.078	4.090	41.560	7.530	26.021

Following [13] multivariate kurtosis β_2 is estimated for the data on metrics of software from Table I and the normalized data on the basis of the decimal logarithm transformation, the Johnson univariate and multivariate transformations for S_B family. The estimator of multivariate kurtosis given by [13]

$$\hat{\beta}_2 = \frac{1}{N} \sum_{i=1}^N \left\{ (\mathbf{z}_i - \bar{\mathbf{z}})^T S_N^{-1} (\mathbf{z}_i - \bar{\mathbf{z}}) \right\}^2 \quad (14)$$

In our case, in the formula (14), the vectors \mathbf{Z} and $\bar{\mathbf{z}}$ should be replaced by the vectors \mathbf{P} and $\bar{\mathbf{P}}$ or \mathbf{T} and $\bar{\mathbf{T}}$, respectively, for the initial (non-Gaussian) or normalized

data. It is known that $\beta_2 = m(m + 2)$ holds under multivariate normality. The given equality is a necessary condition for multivariate normality. In our case $\beta_2 = 24$. The estimators of multivariate kurtosis equal 28.66, 23.87, 37.29 and 23.08 for the data from Table I, the normalized data on the basis of the decimal logarithm transformation, the Johnson univariate and multivariate transformations respectively. The values of these estimators indicate that the necessary condition for multivariate normality is practically performed for the normalized data on the basis of the decimal logarithm transformation and the Johnson multivariate transformation, it does not hold for other data. Note that in our case, the poor normalization of multivariate



non-Gaussian data using the Johnson univariate transformation leads to an increase in the widths of the confidence and prediction intervals of non-linear regression for a larger number of data rows compared to both the Johnson multivariate transformation and the decimal logarithm transformation.

CONCLUSIONS

The non-linear regression equation to estimate the software size of open-source PHP-based information systems is improved on the basis of the Johnson multivariate transformation for S_B family. This equation, in comparison with other regression equations (both linear and nonlinear), has a larger multiple coefficient of determination and a smaller value of MMRE.

When building the equations, confidence and prediction intervals of non-linear regressions for multivariate non-Gaussian data to estimate the software size of open-source PHP-based information systems, one should use multivariate transformations.

Usually poor normalization of multivariate non-Gaussian data or application of univariate transformations instead of multivariate ones to normalize such data may lead to increase of width of the confidence and prediction intervals of regressions, both linear and non-linear, to estimate the software size of open-source PHP-based information systems.

In the future, we intend to try other multivariate normalizing transformations and non-Gaussian data sets.

REFERENCES

1. Hee Beng Kuan Tan, Yuan Zhao, and Hongyu Zhang, "Estimating LOC for information systems from their conceptual data models", in Proceedings of the 28th International Conference on Software Engineering (ICSE '06), May 20-28, 2006, Shanghai, China, pp. 321-330.
2. Matinee Kiewkanya, and Suttipong Surak, "Constructing C++ software size estimation model from class diagram", in 13th International Joint Conference on Computer Science and Software Engineering (JCSE), July 13-15, 2016, Khon Kaen, Thailand, pp. 1-6.
3. D.M. Bates and D.G. Watts. Nonlinear Regression Analysis and Its Applications. Wiley, 1988, 384 p.
4. T.P. Ryan. Modern regression methods. Wiley, 1997, 529 p.
5. G.A.F. Seber and C.J. Wild. Nonlinear Regression. John Wiley & Sons, Inc., 2003, 792 p.
6. R.A. Johnson and D.W. Wichern. Applied Multivariate Statistical Analysis. Pearson Prentice Hall, 2007, 800 p.
7. S Chatterjee and J.S. Simonoff. Handbook of Regression Analysis. John Wiley & Sons, Inc., 2013, 236 p.
8. S. B. Prykhodko, "Developing the software defect prediction models using regression analysis based on normalizing transformations", in Abstracts of the Research and Practice Seminar on Modern Problems in Testing of the Applied Software (PTAS-2016), May 25-26, 2016, Poltava, Ukraine, pp. 6-7.
9. S. B. Prykhodko, N. V. Prykhodko, and K. S. Pugachenko, "Building the equations, confidence and prediction intervals of non-linear regressions on the basis of multivariate normalizing transformations", in Materials of the II International Scientific and Practical Conference "Applied Scientific and Technical Research", Ivano-Frankivsk, Ukraine, April 3-5, 2018, p. 16.
10. P.M. Stanfield, J.R. Wilson, G.A. Mirka, N.F. Glasscock, J.P. Psihogios, and J.R. Davis, "Multivariate input modeling with Johnson distributions", in Proceedings of the 28th Winter simulation conference WSC'96, December 8-11, 1996, Coronado, CA, USA, ed. S.Andradyttir, K.J.Healy, D.H.Withers, and B.L.Nelson, IEEE Computer Society Washington, DC, USA, 1996, pp. 1457-1464.
11. S. Prykhodko, N. Prykhodko, L. Makarova, and K. Pugachenko, "Detecting Outliers in Multivariate Non-Gaussian Data on the basis of Normalizing Transformations", in Proceedings of the 2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON) «Celebrating 25 Years of IEEE Ukraine Section», May 29 – June 2, 2017, Kyiv, Ukraine, 2017, pp. 846-849.
12. S. Prykhodko, N. Prykhodko, L. Makarova, and A. Pukhalevych, "Application of the Squared Mahalanobis Distance for Detecting Outliers in Multivariate Non-Gaussian Data", in Proceedings of 14th International Conference on Advanced Trends in Radioelectronics, Telecommunications and Computer Engineering (TCSET), Lviv-Slavske, Ukraine, February 20–24, 2018, pp. 962-965.
13. K. V. Mardia, "Measures of multivariate skewness and kurtosis with applications", *Biometrika*, 57, 1970, pp. 519–530.

*Рецензент: д.т.н., проф. Ходаков В.Є.
Херсонський національний технічний університет*



РОЗРОБКА ПРОГРАМНОГО КОМПЛЕКСУ ДЛЯ РОЗВ'ЯЗАННЯ ПРЯМИХ ЗАДАЧ МОРСЬКОЇ МАГНІТОМЕТРІЇ

УДК 004.942: 621.3.013

РЯБЕНЬКИЙ В.М.

д.т.н., професор кафедри теоретичної електротехніки та електронних систем Національного Університету
Кораблебудування ім. адмірала Макарова, Україна, м. Миколаїв,
e-mail: optron2@gmail.com

ЧУДАЙКІН І.І.

к. т. н., доцент кафедри теоретичної електротехніки та електронних систем Національного Університету
Кораблебудування ім. адмірала Макарова, Україна, м. Миколаїв,
e-mail: igorlu3141509@rambler.ru

ТАРГУНАКОВА Ю. Д.

магістр, аспірант кафедри теоретичної електротехніки та електронних систем Національного Університету
Кораблебудування ім. адмірала Макарова, Україна, м. Миколаїв,
e-mail: golubkajulya@gmail.com

ВСТУП

Одним з пріоритетних напрямків розвитку теорії і практики інтерпретації аномальних фізичних полів, сформульованих В.Н. Страховим [1], є широке застосування апроксимаційного підходу до вирішення прямих і обернених задач і його синтез з методами аналізу структури даних (розпізнавання образів). В інтерпретації гравітаційних і магнітних даних виняткову роль, як і в усіх геофізичних методах, грає пряма задача. Розв'язок прямої задачі морської магнітометрії (ПЗММ) полягає у визначенні аномального поля, створюваного тілами певної форми і густини, за певних умов занурення і накопичення інформації про ці об'єкти. Розв'язування цього складного завдання зазвичай виконується в декілька етапів. Вирішення ПЗММ відіграє особливу роль при розв'язку оберненої задачі морської магнітометрії. Це пояснюється наступними причинами:

1. Складні моделі, як правило, будуються в інтерактивному режимі при взаємодії людини та системи об-

робки інформації (COI). Для цього алгоритми розв'язання прямих задач повинні мати максимально високу швидкодію.

2. Багато методів розв'язання обернених задач базуються, в тому чи іншому вигляді, на методах підбору, що вимагає багаторазового застосування алгоритму розв'язання прямої задачі. Це диктує необхідність того, щоб саме ці алгоритми були стійкими до похибок обчислення та на порядок-два більш швидкодіючими, ніж алгоритми обернених задач.

3. Високий рівень деталізації при постановці обернених задач приводить, як правило, до надмірно високої розмірності і, як наслідок, крайньої нестійкості обчислювального процесу. Це вимагає від алгоритмів розв'язання прямих задач на порядок вищої деталізації подання моделей середовища в порівнянні з алгоритмами обернених задач.

Особливий інтерес серед ПЗММ викликають задачі визначення аномального поля об'єктів, які складаються з тонкостінних конструкцій. Тонкі пластини та оболонки використовуються в різних галузях сучасної техніки

значно ширше, ніж суцільні тіла, тому розв'язок задач про намагнічення тонкостінних конструкцій має особливе значення. Задача про намагнічення тонких оболонок і пластин являє собою специфічний граничний випадок загальної магнітостатичної задачі, що заслуговує самостійного розгляду [2, 3]. Особливості цієї задачі полягають у тому, що мала в порівнянні з іншими геометричними розмірами товщина оболонки обумовлює появу в рішенні задачі різниці близьких величин. Тому необхідно шукати такий шлях розв'язку завдання, при якому умова малості товщини враховувалася б вже в самому формулюванні задачі.

Рішення ж цієї проблеми багато в чому визначається наявністю ефективних інформаційних технологій для розв'язку ПЗММ.

ІСНУЮЧИЙ СТАН ПИТАННЯ

Методи розв'язку польових задач поділяються на аналітичні, чисельно-аналітичні та чисельні методи [4]. Існуючі аналітичні методи розрахунку поля використовуються при вирішенні задач в областях простої геометрії з однорідним або кусково-однорідним середовищем. Але аналітичні методи не завжди можуть забезпечити точне рішення поставленої задачі та для кожного конкретного випадку треба знаходити свій унікальний розв'язок [5]. При цьому рішення більшості практичних задач виходять настільки складними, що виявляються мало придатними для інженерних цілей [6]. Чисельно-аналітичні методи розрахунку мають такі переваги: дають точний результат розв'язку задач; можливість розрахунку поля без штучного його обмеження. До недоліків цих методів можна віднести: матриці СПАР повністю заповнені, мають відносно великі числа обумовленості; потрібне застосування спеціальних прийомів, які забезпечують збіжність ітераційних процесів, що призводить до необхідності додаткових теоретичних і експериментальних досліджень; ці методи дозволяють отримати явний розв'язок тільки для областей, які мають просту геометрію, для якої можливо знайти аналітичний вираз на границі розділу середовищ. Такі вирази знайдені не для всіх можливих областей. Цей метод не може використовуватися для задач, фундаментальний розв'язок яких не відомий чи не визначений. Тому в даний час все більш актуальною стає проблема розробки загальних чисельних алгорит-

мів, що дозволяють з необхідною точністю здійснити розрахунок поля при складних формах поверхонь розділу середовищ. На сьогодні існує ряд чисельних методів для моделювання магнітного поля. Серед них можна виділити три групи класичних методів: метод скінченних різниць (МСР), метод скінчених елементів (МСЕ) і метод граничних елементів (МГЕ) та групу комбінованих чисельних методів, які використовують переваги різних методів і частково позбавлені їх недоліків.

МСЕ та МСР найбільш просто реалізуються на системах обробки інформації (COI). Для того, щоб чисельно відшукати потрібну функцію (найчастіше потенціал), в заданій області будують об'ємну сітку з досить малим кроком, таким чином, щоб границя сіткової області найкраще збігалася з границею заданої області. При цьому задача зводиться до системи алгебраїчних рівнянь щодо значень шуканої функції у вузлах сітки. У цих методах дискретизації піддається вся розрахункова область, тобто, навіть, повітряний простір, який оточує об'єкт, а невідомі величини визначаються в вузлах сітки [7, 8]. Необхідність підвищення точності результатів рішення задач, зокрема використання більш дрібних сіток істотно збільшує час обчислень [9]. Генерація сіток МСЕ і МКР для розв'язання тривимірних задач представляє певні складності, це призводить до ще більшого ускладнення завдання і робить реалізацію на сучасних обчислювальних засобах надзвичайно складною або неможливою [10]. Похибка МКР полягає в самому методі представлення похідних кінцевих різниць. Для збільшення точності застосовують оптимальну побудову сіток, допоміжні сітки, але все це призводить тільки до ускладнення методу [6]. При використанні методу скінчених різниць та методу скінчених елементів доводиться обмежувати область пошуку рішення, та вводити штучні умови для неврахованої частини простору. Навіть враховуючи сучасні роботи, в яких ці методи розвинуті на нескінченний простір, це викликає труднощі [11].

З існуючих методів розрахунку прямих задач магнітометрії найбільш успішно застосовуються метод вторинних джерел (МВД). Основи цього методу були закладені задовго до появи COI в фундаментальній праці Г.А. Грінберга. Внесли великий вклад в розвиток цього методу такі вчені: І.І. Пеккер, В.І. Дмитрієв, О.В. Тозоні,

І.Д. Маєргойз, Ю.М. Гуревич, В.К. Хуторянський, К.М. Ермохін, К.Б. Ізотова, Л.Ф. Московська, Петров А.А., Д.М. Філіппов та інші. Цей метод найбільш поширений серед чисельно-аналітичних методів розв'язку польових задач. Хоча цей метод має багато переваг, але у нього є багато недоліків [11, 4, 18-21]:

— Інтегральні рівняння мають слабку стійкість рішення: розв'язок існує не за будь-якої правої частини рівнянь і нестійкий до малих змін правих частин та мають неоднозначне розв'язання при $\varepsilon = \infty$.

— По причині великої кількості елементів дискретизації і того, що СЛАР виходять не симетричними і не знаковизначеними, матриці СЛАР мають відносно великі числа обумовленості; потрібне застосування спеціальних прийомів, які забезпечують збіжність ітераційних процесів, що призводить до необхідності додаткових теоретичних і експериментальних досліджень.

— МВД дозволяє отримати явний розв'язок тільки для областей, які мають просту геометрію, для якої можливо знайти аналітичний вираз на границі розділу середовищ. Такі вирази знайдені не для всіх можливих

областей. Цей метод не може використовуватися для задач, фундаментальний розв'язок яких не відомий чи не визначений.

На цей час не існує універсальних САП на базі МВД, які за своїми можливостями були б аналогічні програмам на основі МСЕ (наприклад до програмного комплексу ANSYS). МСЕ в останні роки став найбільш популярним. На базі цього методу створені універсальні програмні комплекси такі як FEMM, ANSYS, MagNet та інші. Однак у цьому методі існує цілий ряд недоліків, які були перераховані вище.

Це призвело до пошуків альтернативних методів, з яких найбільш перспективним є метод граничних елементів. Головну роль у формуванні сучасних основ МГЕ зіграли дослідження таких вчених — О.М. Крилова, В.З. Власова, І.А. Біргера, П.К. Бенерджі і Р. Баттерфілд, К. Бреббія, Д. Теллес і Л. Вроубела, С. Уокера, А.Г. Угодчікова, Н.М. Хуторянського, Дж.Т. Кацікаделіса та інших. В таблиці 1 наведено переваги та недоліки цього методу [3, 13, 16, 17]:

Табл. 1

Переваги МГЕ	Недоліки МГЕ
Дискретизації піддається тільки поверхня тіла, що полегшує чисельне моделювання за допомогою МГЕ і на порядок зменшує число невідомих. Таким чином, спрощується і зміна моделі, пов'язана з урахуванням змін проекту.	Чисельна реалізація МГЕ призводить до СЛАР, у яких матриці коефіцієнтів повністю заповнені та несиметричні.
Метод особливо ефективний для обчислення похідних польової функції (наприклад, потоків, деформацій, напружень, моментів). Можна з легкістю оперувати зосередженими силами і моментами як в області, так і на границі.	Застосування МГЕ вимагає наявності фундаментальної матриці розв'язків. Цей метод не може використовуватися для задач, фундаментальний розв'язок яких не відомий чи не визначений, тобто для об'єктів з довільною геометрією.
МГЕ дозволяє знаходити значення розв'язку і його похідних в будь-якій точці розглянутої області та в будь-який момент часу. Це можливо тому, що метод використовує інтегральне представлення розв'язку як аналітичний вираз, який можна диференціювати і використовувати як аналітичну формулу.	Розроблені програми та підпрограми на базі МГЕ носять локальний характер і були створені лише для розв'язку конкретних задач в досліджуваній області, без особливих засобів візуалізації та повної автоматизації.

У зв'язку з вище перерахованими недоліками, застосування МГЕ призводить до необхідності додаткових теоретичних і експериментальних досліджень. Тому авторами статті був розроблений модифікований метод граничних елементів (ММГЕ) для розв'язку магнітних полів тонких оболонок, що базується на застосуванні фундаментальних рішень, які визначаються простими аналітичними виразами. Філіппов Д.М. розробив аналітичні вирази для розв'язування задач плоскопаралель-

ного магнітного поля [4]. На основі цих рівнянь були розроблені аналітичні вирази для розв'язування задач тривимірного магнітного поля складної конфігурації. Розроблений метод об'єднав переваги МГЕ та МВД, та позбавлений недоліків цих методів. Цей метод дозволяє скоротити розмірність дискретних моделей поля при чисельній реалізації методу, розрахувати об'єкти довільної форми та розміру, а також підвищити точність розрахунку.

МЕТА ДОСЛІДЖЕННЯ

На базі розробленого ММГЕ для розрахунку магнітних полів тонкостінних конструкцій (МППК) для розв'язання прямих задач морської магнітометрії розробити програмний комплекс з дружним інтерфейсом, безкоштовною ліцензією використання, відкритими вихідними кодами та навести приклад чисельного розрахунку за допомогою розробленого комплексу.

Розробка програмного комплексу здійснювалася на операційній системі Linux, з використанням gmsh та api3d генераторів, та бібліотеки GNU GSL за допомогою C та C++ мов програмування. Було створено програмний комплекс МВЕМ, який працює в багатонитевому режимі. МВЕМ приймає вхідні параметри, які задає користувач. Сам користувач вибирає за допомогою яких сіткових генераторів (gmsh чи api3D) виконати побудову поверхневої сітки, та куди вивести результати розрахунку: візуалізувати результати програми за допомогою інтерфейсу gmsh чи вивести їх в файл. Також розроблений комплекс програм дозволяє створити довільний об'єкт в інтерактивному режимі чи завантажити вже існуючий об'єкт; побудувати розрахункову сітку та вивести її на монітор (якщо сітка не підійшла, то можна з легкістю поміняти вхідні параметри триангуляції та розрахувати нову сітку); після того, як була побудована сітка, розраховується потенціал простого шару (ППШ). Знаючи значення ППШ на поверхні об'єкта, можна з легкістю розрахувати та змодельувати результат розрахунку магнітного потенціалу та напруженості у вигляді графіків та векторів в будь-якій точці простору. Програмний комплекс МВЕМ складається з наступних програм: mbem_conector, mbem_potencial та mbem_field.

Програма mbem_conector координує роботу між усіма програмами, які входять в даний програмний комплекс та зв'язує роботу МВЕМ з gmsh та mathcad.

Програма mbem_potencial розраховує ППШ тонкостінного об'єкта та записує результат в файл result.txt. Ця програма на вході приймає наступні аргументи:

'-i' ('--input_format') — якщо цьому аргументу передано значення '0', то інформацію із даними про триангуляцію (кількість точок розбиття nV , їх координати, кількість трикутників та номери вершин, які входять в кожен трикутник (масив *faces*)) зчитуємо з

файла, який має розширення «.out», якщо передано значення '1', то з файла, який має розширення «.msh»;

'-t' ('--output_format_txt') — якщо цьому аргументу передано значення '1', то результат розрахунку ППШ та координати точок поверхневої триангуляції зберігаються в файл, який має розширення «.txt»;

'-p' ('--output_format_pos') — якщо цьому аргументу передано значення '1', то результат розрахунку ППШ та координати точок поверхневої триангуляції зберігаються в файл, який має розширення «.pos»;

'-x' ('--Hx') — значення проекції вектору напруженості зовнішнього поля на вісь Ox ;

'-y' ('--Hy') — значення проекції вектору напруженості зовнішнього поля на вісь Oy ;

'-z' ('--Hz') — значення проекції вектору напруженості зовнішнього поля на вісь Oz ;

'-e' ('--permeability') — значення магнітної проникності;

'-h' ('--thickness') — товщина тонкостінного об'єкта;

'-m' ('--ambient_field') — якщо цьому аргументу передано значення '1', то на екрані буде виведено результуюче поле \vec{H} ($\vec{H} = \vec{H}_0 + \vec{H}_i$), інакше — буде виведено індуковане поле \vec{H}_i ;

'-l' ('--size_length') — довжина площини на якій буде виведений результат;

'-w' ('--size_width') — ширина площини;

'-A' ('--Axis') — вісь координат, перпендикулярна до площини спостереження (може приймати значення 'X', 'Y' або 'Z');

'-a' ('--position_x') — початкові координати площини по x ;

'-b' ('--position_y') — початкові координати площини по y ;

'-c' ('--position_z') — початкові координати площини по z ;

'-N' ('--nPoint_l') — кількість точок розрахунку по довжині;

'-n' ('--nPoint_w') — кількість точок розрахунку по ширині;

'-g' ('--position_graph_width') — задаємо координату відрізка по ширині (враховуючи, що по

ширині значення константно, а по довжині — змінюється);

'-r' ('--nPoint_graph') — кількість розрахункових точок на відрізьку;

'-u' ('--flag_result_potencial') — якщо цьому аргументу передано значення 'I', то на графіку буде виведена залежність магнітного потенціалу від відстані, інакше — на графіку буде виведена залежність модуля магнітної напруженості від відстані.

Один із режимів роботи mbem_potencial включає в себе запуск mbem_field для завершення розрахунків. Програма mbem_field формує файли із розрахованими значеннями вектору напруженості та магнітного потенціалу. Ця програма на вході приймає наступні аргументи: '-t' ('--output_format_txt'); '-p' ('--output_format_pos'); '-x' ('--Hx'); '-y' ('--Hy'); '-z' ('--Hz'); '-a' ('--position_x'); '-b' ('--position_y'); '-c' ('--position_z'); '-N' ('--nPoint_l'); '-n' ('--nPoint_w'); '-l' ('--size_length'); '-w' ('--size_width'); '-m' ('--flag_ambient_field');

'--position_graph_width'); '-r' ('--nPoint_graph'); '-u' ('--flag_result_potencial'); '-A' ('--Axis'). Вхідні аргументи програми mbem_field приймають такі ж самі значення, що і однойменні вхідні аргументи програми mbem_potencial.

При виборі api3d генератора інтерфейс користувача програмного комплексу MBEM являє собою командну строку Linux та складається з двох програм: mbem_potencial та mbem_field. Кожна з яких — це консольна утиліта. Так користувач запускає mbem_potencial, передає вхідні аргументи та задає режим роботи.

Інтерфейс MBEM при виборі gmsh складається з інструментів gmsh для створення 3d-моделі та побудови поверхневої сітки; вхідних параметрів задачі, параметрів для візуалізації результатів розрахунку (параметри для виводу напруженості МПТК та графіків залежності модуля напруженості та магнітного потенціалу від відстані), меню вибору режиму роботи програмного комплексу та кнопки "RUN" для запуску MBEM.

Вхідні параметри, необхідні для розрахунку

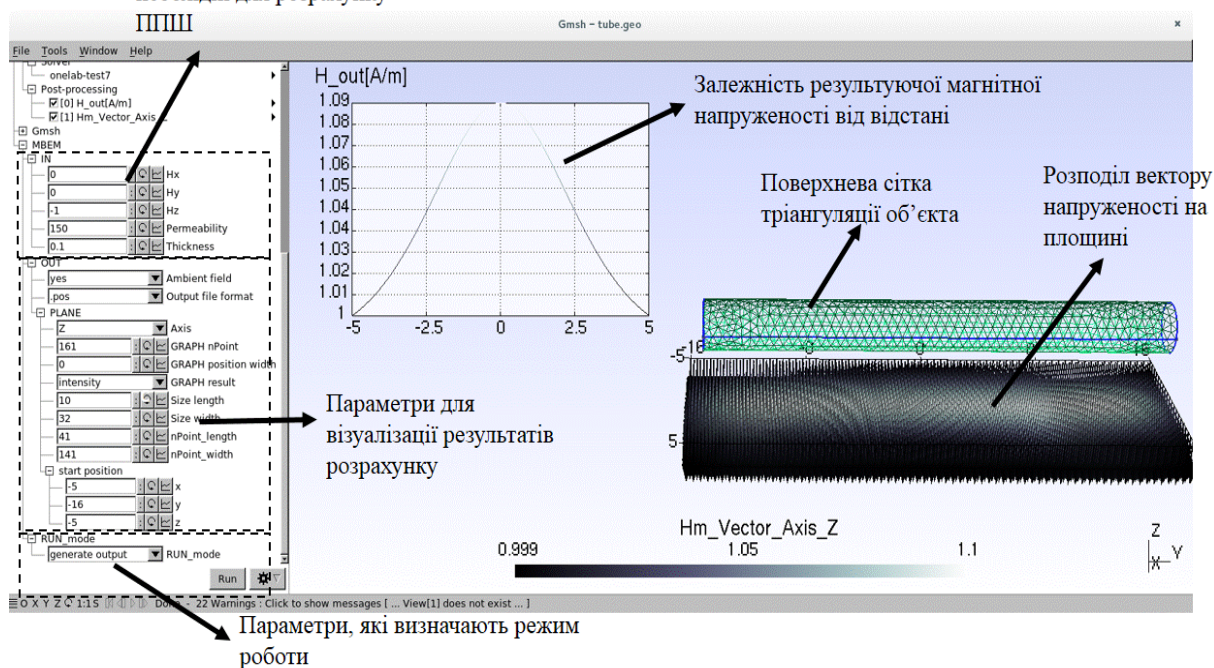


Рис. 1 Інтерфейс програмного комплексу MBEM

Для ілюстрації роботи розробленого вище програмного комплексу було розраховане магнітне поле поблизу підводного човна типу «М» довжиною $l = 39.6$ м, шириною $d = 3.13$ м, товщина оболонки $\delta = 0.1$ м, який знаходиться в зовнішньому полі $H_{0z} = -1$ А/м.

На першому кроці створимо об'єкт. Це можна зробити наступними способами: за допомогою меню графічного інтерфейсу користувача; за допомогою текстових команд через вхідний файл, який має розширення «.geo»; імпортувати готову CAD модель. В даному випадку для створення моделі підводного човна було побудовано CAD модель за допомогою програми OpenSCAD. Код моделі наступний:

```
// Generic parameters
top = 0.6;
// Ship parameters
length = 36.9;
width = 3.13;
height = 3.13;
```

```
// Cabin parameters
cabin_height = 1.5;
cabin_width = 2;
cabin_length = 3;
$fs = 0.1;
difference() {
  scale([length/2, width/2, height/2])
  sphere(r=1);
  translate([-length, -width, top])
  cube([length * 2, width * 2, height]);
}
translate([-1, 0, cabin_height/2 + top])
cube([cabin_length, cabin_width, cabin_height],
center=true);
```

Потім CAD модель була конвертована за допомогою FreeCAD у файл, який має формат «.brep». Модель у форматі «.brep» була відкрита за допомогою gmsh. Після того як об'єкт було створено, будуємо поверхневу сітку за допомогою gmsh (рис. 2).

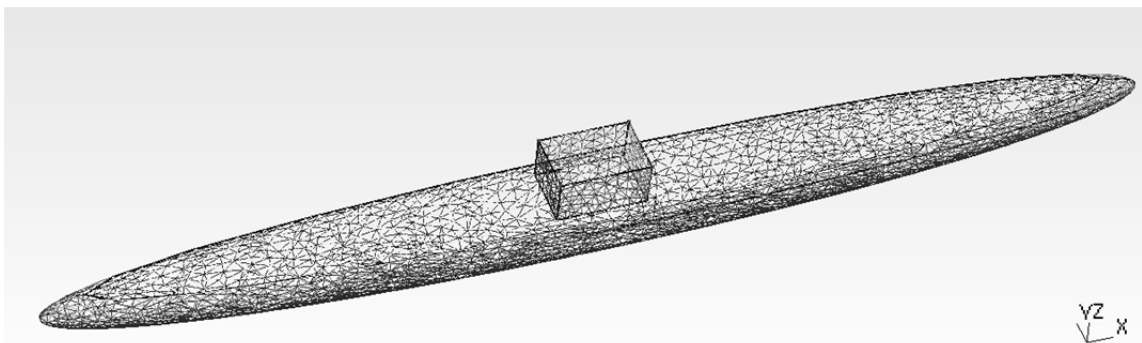


Рис. 2 Розрахункова поверхнева сітка підводного човна

У меню програми вибираємо Solver / onelab-test7 та задаємо необхідні параметри (рис. 1). Вхідні параметри (значення зовнішнього магнітного поля, товщина об'єкта, магнітна проникність матеріалу) задаються в підменю MBEM/IN. Параметри для виводу результатів розрахунку (координати площини, на якій буде виведено вектор магнітної напруженості; координати відрізка, на якому буде виведено значення магнітного потенціалу чи модуля магнітної напруженості від відстані; вибір вихідного поля: вивести індуковане чи результуюче) задаються в підменю MBEM/OUT. Параметри, які

визначають режим роботи (розрахувати потенціал простого шару чи вивести розраховане поле на екран) задаються в підменю RUN_MODE.

Після того як користувач натисне кнопку «RUN» відбуваються необхідні розрахунки та виводиться результат на екран. На рис. 3 та рис.4 представлено результуюче та індуковане магнітні поля підводного човна типу «М», розраховані на площині, яка знаходиться під об'єктом на відстані 2.5 м та має координати $-5 \leq x \leq 5$, $-16 \leq y \leq 16$, $z = -3$.

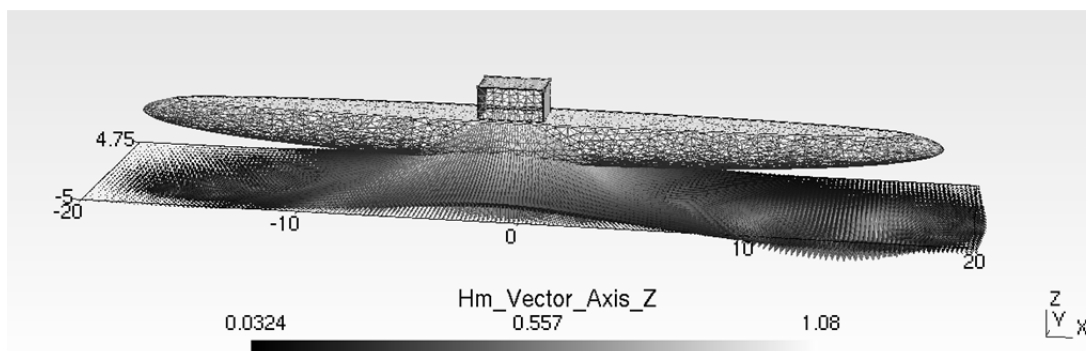
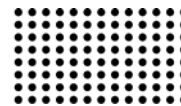


Рис. 3 Розподілення вектору індукованої напруженості \vec{H} на площині, яка знаходиться під підводним човном

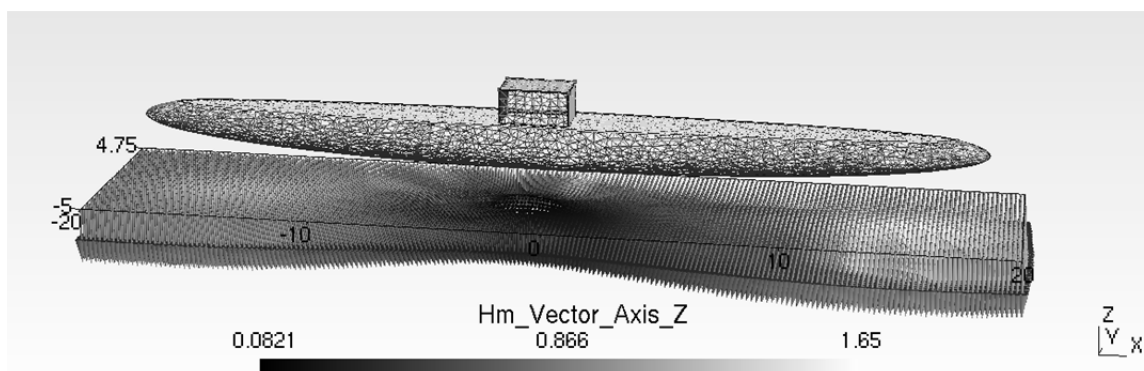


Рис. 4 Розподілення вектору результуючої напруженості \vec{H} на площині, яка знаходиться під підводним човном

ВИСНОВКИ

Наведений вище аналіз польових методів показує доцільність розробки на якісно новому рівні розв'язку прямої задачі магнітометрії затонулих об'єктів в тривимірному просторі та створення програмного комплексу для розрахунку та моделювання магнітних полів цих об'єктів — знаходження полів типових збуджувальних тіл (наборами яких можна апроксимувати реальні об'єкти). Розроблений програмний комплекс МВЕМ має наступні переваги:

1) Цей програмний комплекс має безкоштовну ліцензію використання та відкриті вихідні коди, тобто він доступний для перегляду, вивчення та зміни.

2) Дискретизації піддається тільки поверхня тіла, що полегшує чисельне моделювання і на порядок ско-

рочує число невідомих. Таким чином, спрощується і зміна моделі, пов'язана з урахуванням змін проекту (вхідних умов).

3) МВЕМ дозволяє знаходити значення розв'язку і його похідних в будь-якій точці розглянутої області та в будь-який момент часу.

4) Для розв'язку задач, які мають необмежену область, не потрібне введення штучних границь.

Цей комплекс дозволяє розраховувати та моделювати основні імовірнісні характеристики знаходження затонулих об'єктів, на підставі яких розробляється стратегія проведення експериментальних досліджень.

Наведений вище чисельний розрахунок магнітного поля підводного човна типу «М» засвідчує ефективність роботи розробленого програмного комплексу.

REFERENCES

1. Strahov V.N. Glavneyshaya zadacha v razvitii teorii i praktiki interpretatsii potentsialnykh poley v nachale XXI veka razrushenie gospodstvuyushego stereotipa myishleniya. // Geofizika. — 2001. № 1. — P. 3-18. (Rus)
2. Hayrullin, F. S. Raschet tonkostennykh konstruktsiy slozhnoy formy na osnove approksimiruyuschiy funktsiy s konechnymi nositeliyami. Kazan: KNITU, 2012. 176 p. (Rus)
3. Krasnov I. P. Calculated methods of ship magnetism and electrical engineering. Leningrad: Sudostroenie, 1986. — 216 p. (Rus)

4. Pashkovskiy A.I. Chislennno-analiticheskie metody standartnykh elementov dlya modelirovaniya statsionarnykh fizicheskikh poley v lineynykh kusochno-odnorodnykh i nelineynykh sredakh: extended abstract of dr. tech. sci. diss: 05.13.18 / Pashkovskiy Alexander Vladimirovich. — Novocherkassk, 2014. — 36 p. (Rus)
5. Demirchyan K.S., Chechurin V.L. Mashinnyye raschety elektromagnitnykh poley. — Moscow: Vysshaya shkola, 1986. — 239 p. (Rus)
6. Arhipova E.V. Modelirovanie vtyazhnykh bronevyykh elektromagnitov i razrabotka i usovershenstvovannykh metodik ih proektnogo rascheta: dis. kand. tehn. nauk: 05.09.05 / Arhipova Elena Vladimirovna — Cheboksary: 2014. — 228 p. (Rus)
7. Taflove A. Computational electrodynamics: the finite difference time domain method / A. Taflove, S. Hagness. — Boston-London: Artech House, 2000. — 852 p.
8. Morozov V.A. Bank modeley i metodov dlya rascheta elektrostaticheskikh poley: dissertatsiya na soiskanie uchenoy stepeni kandidata tehnikeskikh nauk. 2003. 137 p. (Rus)
9. Akimova E.N. Parallelnyye algoritmy resheniya zadach gravi-magnitometrii i uprugosti na mnogoprotsessornykh sistemakh s raspredelennoy pamyaty: extended abstract of dr. tech. sci. diss: 05.13.18 / Akimova Elena Nikolaevna — Chelyabinsk: 2009. — 35 p. (Rus)
10. Rezinkina M. M. Raschet trehmernykh elektricheskikh poley v neodnorodnykh sredakh metodom konechnykh ob'emov: dis. dokt. tehn. nauk: 05.09.05 / Rezinkina Marina Mihaylovna — Kharkiv: 2005. — 238 p. (Rus)
11. Sorokin D.S. Rozvitok metodu integralnykh rivnyan dlya rozrahunku nestatsionarnykh elektrodinamichnykh protsessiv v sistemakh z osovoyu simetrii: dis. kand. tehn. nauk: 05.09.05 / Sorokin Dmitro Sergiyovich — Kiyiv, 2017 — 152 p. (Ukr)
12. Nikitenko A.G. Elektromagnitnyye mehanizmy. Analiz i sintez. — Moscow: Vyssh. shkola 1998. — 330 p. (Rus)
13. Filippov D.M. Razvitie metoda vtorichnykh istochnikov dlya modelirovaniya ploskparallelnogo magnitnogo polya elektromagnitnykh sistem: extended abstract of dis. kand. tehn. Nauk: — Kharkiv, 2013. — 24 p. (Rus)
14. Tozoni O.V. Metod vtorichnykh istochnikov v elektrotehniki. Moscow: Energiya, 1975. 296 p. (Rus)
15. Maergoyz I.D. Iteratsionnyye metody rascheta staticheskikh poley v neodnorodnykh, anizotropnykh i nelineynykh sredakh. — Kiev: Naukova Dumka, 1979. — 212 p. (Rus)
16. Ilin V. P. Chislennyye metody resheniya zadach elektrofiziki / V. P. Ilin. — Moskva: Nauka, 1985. — 336 p. (Rus)
17. Bloh Yu. I. (2009) Interpretation of gravitational and magnetic anomalies. Retrieved from <http://sigma3d.com/pdf/books/blokh-interp.pdf>. (Rus)
18. Katsikadelis, John T The Boundary Element Method for Engineers and Scientists and Edition. — Academic Press, 2016, 464 p.
19. Kornienko Yu. V. (2015) Matematicheskoe modelirovanie i analiz ploskopro-stranstvennykh konstruktsionnykh elementov v SAPR: extended abstract of dr. tech. sci. diss: 05.13.12 / Odessa National Polytechnic University. Odessa. 2015. 170 p. (Rus)
20. Arushanyan I. O. Primenenie metoda granichnykh integralnykh uravneniy dlya chislennogo resheniya zadachi Dirihle v oblastiakh s uglovymi tochkami / I. O. Arushanyan // Vyichislitelnyye metody i programmirovaniye. — 2000. — T.1, #1. — P. 1–7.
21. Bazhenov V. G. Metody granichnykh integralnykh uravneniy v reshenii zadach trehmernoy dinamicheskoy teorii uprugosti s sopryazhennyimi polyami / V. G. Bazhenov, L. A. Igumnov. — M.: FIZMATLIT, 2008. — 352 p.

*Рецензент: д.т.н., проф. Коваленко В.Ф.,
Херсонський національний технічний університет*



АНАЛІЗ ЗАДАЧІ ПОБУДОВИ АВТОМАТИЗОВАНОЇ СИСТЕМИ ВІЯВЛЕННЯ ТА ПОПЕРЕДЖЕННЯ НАДЗВИЧАЙНИХ СИТУАЦІЙ НА ПІДПРИЄМСТВАХ ПІДВИЩЕНОЇ НЕБЕЗПЕКИ

УДК 004.386

ЯКОВЕНКО Вадим Олександрович

д.т.н., доцент, професор кафедри кібербезпеки Університету митної справи та фінансів,

Наукові інтереси: математичне моделювання та обчислювальні методи в інформаційних технологіях та кібербезпеці

E-Mail: yakovenko@ua.fm

УЛЬЯНОВСЬКА Юлія Вікторівна

к.т.н., доцент, доцент кафедри кібербезпеки Університету митної справи та фінансів,

Наукові інтереси: інформаційні технології, методи обробки неповних та нечітких даних в інтелектуальних автоматизованих системах

E-mail: yuliyauyv@gmail.com

ВСТУП

Важливу складову економіки держави займає промисловий сектор, до якого входять підприємства хімічної, металургійної, нафтопереробної та іншої промисловості. Важливу роль в промисловості відіграють системи сигналізації, метою яких є повідомлення операторів про аномальні або аварійні ситуації [1]. Особливо актуальним є впровадження таких систем на підприємствах – об'єктах підвищеної небезпеки (ОПН), до яких відноситься значна кількість підприємств України. Згідно закону України «Про об'єкти підвищеної небезпеки» об'єкт підвищеної небезпеки – об'єкт на якому використовуються, виготовляються, переробляються, зберігаються або транспортуються одна або кілька небезпечних речовин чи категорій речовин у кількості, що дорівнює або перевищує нормативно встановлені порогові маси, а також інші об'єкти як такі, що відповідно до закону є реальною загрозою виникнення надзвичайної ситуації техногенного та природного характеру [2].

Більшість міст України є промисловими, а специфіка українських промислових підприємств полягає в тому, що здебільшого їх територія розташована в межах населених пунктів, а відстань від ОПН до жилих кварталів населення може складати усього сотні метрів. На

ОПН існує ризик виникнення техногенних аварій та катастроф. Результатом їхнього прояву є раптовий вихід із ладу машин, механізмів та агрегатів під час експлуатації, що супроводжується серйозними порушеннями виробничого процесу, вибухами, утворенням осередків пожеж, радіоактивним, хімічним або біологічним зараженням великих територій, ураженням та загибеллю людей. Аналіз статистичних даних показав, що найбільш небезпечними в техногенному відношенні є: Донецька, Луганська, Дніпропетровська, Запорізька, Харківська, Одеська, Миколаївська, Київська області. Серед техногенних загроз Донецької, Дніпропетровської, Запорізької, Миколаївської областей найбільшу небезпеку для території та населення становлять радіаційна, гідродинамічна, хімічна, пожежонебезпека та вибухонебезпека, Луганської області – гідродинамічна, хімічна, пожежонебезпека та вибухонебезпека, Харківської області – пожежі та вибухи, в тому числі в будівлях або спорудах житлового призначення, Одеської області – радіаційна, хімічна, пожежонебезпека та вибухонебезпека, Київської області – гідродинамічна, радіаційна та хімічна небезпеки [3].

Саме тому запобігання виникненню надзвичайних ситуацій (НС) техногенного характеру та своєчасне інформування респондентів про загрози їх виникнення є актуальною проблемою. Завчасне виявлення НС, точність та достовірність параметрів систем моніторингу є запорукою запобігання техногенних катастроф різного рівня. Методичне керівництво і координація діяльності системи моніторингу і прогнозування НС на державному рівні має здійснюватися Державною службою України з надзвичайних ситуацій (ДСНС). Прогноз ризиків НС на території країни в цілому здійснює ДСНС у взаємодії з іншими центральними органами виконавчої влади. Разом з цим, на цей час моніторинг і прогнозування НС в Україні здійснюються на рівні регіональних, галузевих або інших самостійних підсистем, не об'єднаних у єдиний інформаційно-аналітичний комплекс. Загальнодержавну систему моніторингу джерел НС та їх прогнозування у державі не створено [4]. Рівень розвитку інформаційних технологій на сьогоднішній день дозволяє значно підвищити інформативність систем моніторингу та достовірність оцінки параметрів фізичних процесів, які використовуються на ОПН для запобігання виникненню НС. Задача розробки автоматизованої інформаційної системи виявлення та попередження надзвичайних ситуацій на основі даних систем моніторингу з використанням сучасних інформаційних технологій займає важливе місце в вирішенні загальнодержавної проблеми запобігання виникненню НС.

Аналіз літературних джерел та постановка задачі дослідження. На підприємствах підвищеної

небезпеки для раннього виявлення надзвичайних ситуацій та, у випадку їх виникнення сповіщення людей, які знаходяться в зонах можливого ураження, впроваджуються автоматизовані системи (АС). У разі виявлення загрози або виникнення надзвичайної ситуації АС повинна автоматично здійснювати інформування про виявлену загрозу відповідальних осіб, на яких покладено виконання певних дій щодо недопущення виникнення НС або мінімізації негативних наслідків у разі її виникнення. Система раннього виявлення НС та оповіщення (СРВНСО) повинна виконувати такі функції [5]:

- Безперервно отримувати данні від джерел первинної інформації,
- Контролювати в реальному вимірі часу відповідність поточних (граничних) значень параметрів проектним режимам технологічного процесу об'єкта та (або) унормованим значенням параметрів джерел НС природного характеру,
- Інформувати працівників, відповідальних за функціонування технологічного обладнання, щодо виявлених фактів досягнення докритичних та критичних значень параметрів, які контролюють,
- Інформувати посадових осіб, які відповідають за стан техногенної безпеки об'єкта, про факти досягнення критичних значень параметрами, які контролюють.

Складовою частиною систем запобігання НС є системи моніторингу (рис. 1) [4]. Системи внутрішнього моніторингу (лівий стовпчик рис. 1) є початком усіх процесів запобігання НС. Складність систем має бути залежною від рівня небезпечності об'єктів.

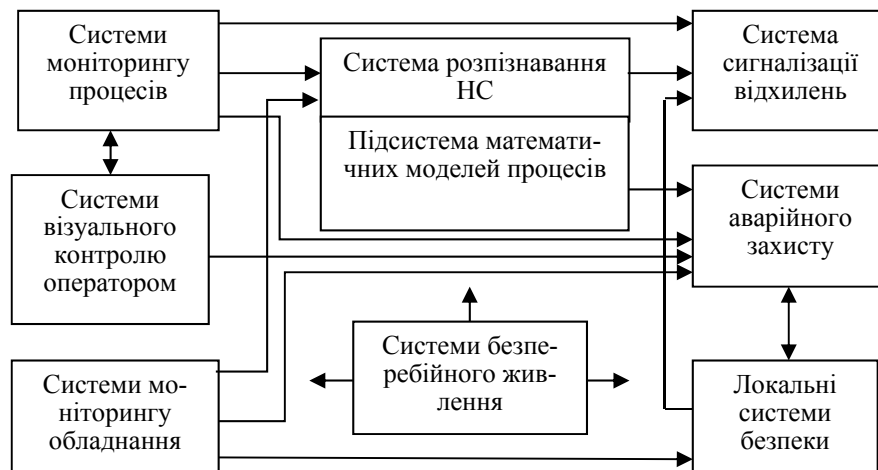


Рис. 1. Системи моніторингу у складі системи запобігання НС

Розробками систем моніторингу та діагностики займаються провідні світові компанії. Так компанією Siemens розроблені системи раннього попередження, які дозволяють контролювати та виявляти відхилення в роботі обладнання [6].

На вітчизняному ринку в галузі розробки систем автоматизації використовуються SKADA системи. На основі технологій таких систем реалізуються на ряду з іншими системами системи сигналізації та документування відхилень від заданих параметрів технологічного процесу та ін. [7].

Для великих підприємств хімічної та металургійної галузі, які відносяться до ОПН, актуальним є впровадження дистанційних систем моніторингу. Зважаючи на великі масштаби виробництва, навіть незначна його оптимізація за рахунок систем дистанційного моніторингу дає змогу значно підвищити продуктивність. Системи дистанційного моніторингу та управління надають можливість покращити конкурентоспроможність підприємства та оптимізувати виробництво [8].

У роботі [9] обґрунтовано необхідність використання додаткових інтелектуальних інформаційних систем для автоматизованої системи управління металургійним підприємством і для попередження надзвичайних ситуацій. У статті продемонстровано структура взаємодії автоматизованих систем управління, систем аварійного оповіщення і систем управління комп'ютерною інформацією для металургійних підприємств. Аналіз дослідження показує спосіб запобігання виникненню аварійних ситуацій, підтверджений позитивними результатами експериментальних досліджень.

Аналіз вітчизняного та закордонного досвіду побудови систем моніторингу та раннього виявлення НС свідчить, що подальший розвиток ефективності науково – методичних принципів побудови повинен виконуватись шляхом підвищення інтелектуального рівня СРВНСО; ступеню автоматизації функцій; забезпечення необхідної надійності інформації, забезпечення відповідного рівня захисту інформації. Вирішення цих питань залежить від розв'язання ряду основних науково – технічних проблем, які можна згрупувати в наступних напрямках: функціонально – алгоритмічний синтез; типізація і уніфікація технічних засобів раннього виявлення НС; системний підхід до створенню систем моніторингу та СРВНСО.

Метою дослідження є аналіз задачі створення автоматизованої системи раннього виявлення загрози виникнення надзвичайних ситуацій техногенного характеру.

Для вирішення поставленого завдання необхідно дослідити інформаційні потоки на виробництві, характеристики технологічних процесів, дослідити структуру та сформулювати функціональні вимоги до СРВНСО, розглянути принципи створення систем раннього виявлення НС на ОПН, дослідити характеристики інформації, що оброблюється а також математичні моделі та методи обробки даних.

Дослідження структури та формулювання функціональних вимог до СРВНСО. Більшість хімічних процесів та виникнення порушень в технологічному процесі мають імовірнісний характер. Але, як правило, підприємства не афішують порушення ТП, тому статистичні дані не завжди є достовірним підґрунтям для розробки необхідних методів та моделей обробки даних. Альтернативою імовірнісних моделей у цьому випадку можуть бути системи нечіткої логіки, які дозволяють оперувати неповними та нечіткими даними а також дозволяють створювати методи та моделі обробки даних на основі знань експертів. Інтеграція інформаційних систем управління надзвичайними ситуаціями в структури системи управління знаннями забезпечить здатність приймати рішення щодо пошуку інформації у надзвичайних ситуаціях та приймати обґрунтовані рішення в умовах невизначеності та ризику. Інтелектуальні системи для управління надзвичайними ситуаціями мають включати функції, які дозволяють змінювати ролі та дозволяють людям отримувати доступ до змін на основі ситуаційних вимог [10].

Система раннього сповіщення повинна забезпечувати:

- підвищення ефективності попереджень НС за рахунок своєчасного виявлення порушень в технологічному процесі;
- підвищення імовірності виявлення НС за рахунок використання якісно нових моделей представлення даних та знань а також методів їх обробки;
- простоту адаптації системи до можливих технологічних змін.

Сформулюємо функціональні задачі, які повинна вирішувати СРВНСО:

- забезпечення оператора інформацією про значення параметрів технологічного процесу, поточний стан технологічних об'єктів регулювання та апаратних засобів системи;
- інтелектуальний аналіз стану ТП та прогнозування можливості виникнення НС;
- забезпечення попереджувальної і аварійної сигналізації процесу автоматичного регулювання, несправності технологічного обладнання (датчики і виконавчі механізми), несправності власних апаратних засобів;
- реєстрація інформації в базі знань: данні про технологічні параметри об'єктів контролю, стан технологічного обладнання, стан апаратних засобів, показники навколишнього середовища та інші данні від джерел первинної інформації;

- ведення журналу подій;
- діагностика технологічного обладнання;
- формування архіву.

З урахуванням сформульованих вимог, задач та інтелектуальної складової системи структурна схема установок виявлення НС може мати вигляд, наведений на рис. 2, де 1- пульт централізованого спостереження, 2 – автоматизована система централізованого сповіщення, 3 – пульт керування, 4 – комунікаційний пристрій, 5 – датчики показників навколишнього середовища, 6 – джерела первинної інформації (ДПІ), 7 – інтелектуальний модуль обробки даних, 8 – пристрій оповіщення, 9 – кінцеві технічні засоби інформування та оповіщення.



Рис. 2. Структурна схема інтелектуальної системи сповіщення

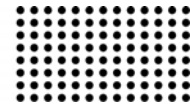
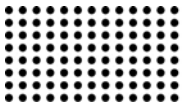
Оповіщення реалізується за допомогою передачі сигналів з вимірювальних датчиків до респондентів сповіщення. Збір даних від джерел первинної інформації повинен охоплювати:

- Прийом даних у вигляді аналогових, дискретних або кодованих сигналів, які містять інформацію про поточне значення контрольованих параметрів та досягнення ними граничних значень, а також інформацію, що стосується спрацювання ручних оповіщувачів;
- Прийом від джерел первинної інформації (ДПІ) відомостей про втрату ними працездатності;

– Визначення працездатності каналів зв'язку з джерелами первинної інформації (датчиками), в залежності від вихідного інтерфейсу контролюванням ліній зв'язку на обрив або замикання, відхилень встановлених параметрів технічного діапазону конкретного ДПІ, наявність підтвердження від ДПІ про обмін даними, достовірність даних, отриманих від ДПІ.

Принципи розробки СВНСО

Проектування, розробка та впровадження систем моніторингу та раннього виявлення НС необхідно розглядати з позицій системного підходу. Підхід до пробле-



ми управління процесам запобіганню виникненню надзвичайних ситуацій як процесу, який враховує взаємозв'язки частин системи або окремих її підсистем є основною рисою системного підходу до розробки СРВНСО.

З урахуванням вищесказаного на основі викладених в роботі [11] принципів особливостей, які мають будь-які автоматизовані системи управління сформулюємо особливості, які властиві системам моніторингу НС:

1) До завершення повної реалізації проекту можливість випробувань функціонуючої системи на об'єкті практично виключена. Всі невдалі рішення і помилки, які не вдалося врахувати і передбачити в ході розробки, можна виявити тільки при випробуваннях системи і введення її в експлуатацію.

2) В процесі розробки системи моніторингу повинна заздалегідь плануватися її еволюція: при побудові системи передбачена можливість її розвитку як по вертикалі, тобто в сенсі інтеграції в її рамках окремих підсистем, які раніше мали самостійне значення, так і по горизонталі, тобто в сенсі розширення кола завдань, які вирішуються окремими підсистемами, та розширення зони моніторингу. Вже на цій стадії необхідно ретельно продумати сполучення різних частин системи на основі взаємозалежності її окремих частин, що вводяться в експлуатацію в різний час.

3) Оскільки частина датчиків первинної інформації знаходяться на відкрито рому просторі, при виникненні НС при прийнятті рішень про можливі наслідки враховуються такі метеорологічні фактори як атмосферний тиск, швидкість та напрям руху вітру, атмосферні опади та ін. можна стверджувати що СРВНСО тісно пов'язана з зовнішнім середовищем та іншими системами, що впливає на алгоритмічне та програмне забезпечення СРВНСО, а часто і на мету і критерії ефективності, змушуючи перебудовувати внутрішній зміст системи. У зв'язку з цим з самого початку розробки системи в неї повинні бути закладені властивості адаптивності, гнучкості в перебудові.

4) Необхідно також враховувати рух меж техногенних аварій та катастроф шляхом побудови нових математичних моделей таких процесів та розробки нових методів їх чисельної реалізації. Під час розробки математичних моделей необхідно враховувати, що, по-перше, матема-

тична модель повинна бути адекватною об'єкту і його розрахунковій схемі, достатньо повно і точно описувати кількісні характеристики, і, по-друге, складність математичних моделей повинна бути обмежена такими умовами, які дозволяють на основі обчислювального експерименту отримати певні результати. Також необхідно використання обчислювального експерименту при проектуванні, виборі та оптимізації режимів роботи, аналізі надійності і прогнозуванні відмов та аварійних ситуацій технічних об'єктів. Слід зазначити, що методи математичного моделювання і обчислювального експерименту є важливою складовою певних підходів, які характерні для сучасних інформаційних технологій. Такі методи дозволяють на основі кількісного аналізу процесів, що відбуваються в системах з розподіленими параметрами, оптимізувати технологічні процеси в технічних установках. Будемо застосовувати в подальшому умовну схему математичного моделювання технічних об'єктів поетапно: технічний об'єкт, технічна модель, математична модель, вибір методу кількісного аналізу і його алгоритму, створення програмного продукту, визначення адекватності моделі.

5) Введення в дію будь-якої нової технічної системи пов'язані, в основному, з монтажем і налагодкою обладнання, при цьому в обов'язки людини входить забезпечення безперебійної роботи цього обладнання. При створенні СРВНСО велику питому вагу займає програмне забезпечення, пов'язане з переробкою великого обсягу інформації. Необхідно не тільки розробити різні алгоритми і процедури переробки інформації, а й передбачити заходи для їх точного дотримання, і алгоритми дій в разі їх порушення. Тому підготовці персоналу до роботи в умовах функціонування СРВНСО повинна бути приділена особлива увага. Тому є доцільним введення в структуру систем моніторингу елементів автоматизованих систем підтримки прийняття рішень.

5) У процесі створення СРВНСО витрати на проведення необхідних досліджень, проектування системи, розробку програмного забезпечення, підготовку персоналу тощо значно перевищують вартість комплексу технічних засобів, їх монтажу та наладки.

ВИСНОВКИ:

1) У роботі проаналізована задача розробки автоматизованої системи виявлення та запобігання надзвичайних ситуацій на підприємствах підвищеної небезпе-

ки. Практичну цінність має запропонована загальна схема передачі даних від джерел первинної інформації до респондентів сповіщення з урахуванням інтелектуальної складової.

2) Визначені напрямки подальшого наукового дослідження, яке полягає у розробці математичних методів та моделей обробки даних від джерел первинної інформації в системах моніторингу із застосуванням систем нечіткої логіки, які дозволяють оперувати неповними та нечіткими даними а також дозволяють створювати методи та моделі обробки даних на основі знань експертів.

3) Сформульовані вимоги та функціональні задачі, якими повинна володіти СРВНСО. На основі сформульованих вимог та функціональних задач запропонована структурна схема інтелектуальної системи сповіщення.

4) Сформульовано принципи особливості, які властиві автоматизованим системам моніторингу НС. Показано, що доцільним є використання системного підходу при побудові систем раннього виявлення надзвичайних ситуацій а розробку автоматизованої системи необхідно розглядати як процес, який враховує взаємозв'язки частин системи або окремих її підсистем.

ЛІТЕРАТУРА

1. Su, J. A multi-setpoint delay-timer alarming strategy for industrial alarm monitoring [Text] / J. Su, D. Wang, Y. Zhang, F. Yang, Y. Zhao, X. Pang // Journal of Loss Prevention in the Process Industries. – 2018.-№. 54.- Pp. 1-9 doi.org/10.1016/j.jlp.2018.02.004/
2. Закон України «Про об'єкти підвищеної небезпеки» від 18.01.2001 №2245-III // Vidomosti verhovnoi rady Ukrainy.- 2001.- №15.- st. 73
3. G.V. Ivanec. Analiz stanu tehnogennoi, pryrodnoi ta socialnoi nebezpeky administratyvno-territorialnyh odynyc Ukrainy na osnovi danyh monitoryngu // Zbirnyk naukovykh prac Kharkivskogo universytetu Povitrianyh syl.- 2016.- vol. 3(48).- p 142-145.
4. Kropotov, P.P. Stvorennia suscasnoi systemy monitoryngu bezpeky– aktualna derzhavna ta naukova zadaca // P.P. Kropotov, V.V. Begun, V.F. Grecaninov / Systemy obrobky informacii.- 2015.- vol.11(136).- p.199-206
5. Avtomatyzovani systemy ranniogo dbzdktyyz zagrozy vynykennia nadzvychaynyh sytuaciy ta opovishennia naseleennia. DBN B.2.5.-76:2014. Kyiv. Minregion Ukrainy. – 2014.- 38 c.
6. Sistema dignostiki oborudovania SPPA-D3000 www.energy.siemens.com
7. Scada Trace mode www.tracemode.ua/news/azot_cherk_askug/
8. Plahotny M. V. Osoblyvosti pobudovy sestem dystanciynogo monitoryngu parametriv obladdnannia / M. V. Plahotny, A. Yu. Grycina // Visnyk Nacionalnogo universytetu "Lvivska politehnika". – 2011. – № 717 : Komputerni systemy ta mrezhzi. – С. 125-133.
9. K.V. Antipov, M.P. Maslakov, K.I. Yurenko Improvement of the *Automated Control Systems* for the Development of the Metallurgy Procedia Engineering, Vol. 129, 2015, Pages 1010-1014
10. Dorasamy, M. Integrated community emergency management and awareness system: A knowledge management system for disaster support [Text] / M. Dorasamy, M. Raman, M. Kaliannan / Technological Forecasting and Social Change. – 2017.- №121,- P. 139-167 doi.org/10.1016/j.techfore.2017.03.017
11. Duel, M.A. Konceptualnye osnovy postroenia ASU slozhnyimi energoobektami // M. A. Duel, G.I. Kaniuk, T. N. Fursova / Vostocno-evropeyskiy zhurnal peredovykh tehnologiy.- 2011.-№ 6/3(54).- c. 4.-11

*Рецензент: д.т.н., проф. Коваленко В.Ф.,
Херсонський національний технічний університет*



ОБЗОР МОДЕЛЕЙ ГЕТЕРОГЕННОГО КАТАЛИЗА

УДК 504.7.064.3:614

КОЗУЛЯ Татьяна Владимировна

доктор технических наук, профессор кафедры программной инженерии
и информационных технологий управления НТУ «ХПИ»

E-Mail: kozula@kpi.kh.ua

ВЕДЬ Елена Валерьевна

старший преподаватель кафедры распределенные информационные системы и облачные технологии НТУ «ХПИ».

Научные интересы: математическое моделирование, прикладная экология,
катализ, компьютерное моделирование, имитационное моделирование.

ВЕДЕНИЕ

Математическое моделирование химических процессов и реакторов в настоящее время сформировалось в систему понятий и методов, применение которых позволило решить ряд важных экологических и научно-технических проблем при разработке технологических процессов и аппаратов. Основой производства большинства химических и нефтехимических продуктов является катализ. Несмотря на многообразие и сложность решаемых задач, методология разработки каталитических процессов и реакторов основана на структурированном, иерархическом подходе, суть которого в последовательном решении задач на различных уровнях: кинетическом, слоя катализатора, реактора, всей химико-технологической схемы в целом. При этом вся полученная информация на предыдущем уровне становится составной частью следующего. Проектирование химических реакторов каталитического типа предусматривает понимание процессов, проходящих в них. В связи с этим можно выделить три уровня описания каталитической реакции. Первый уровень описания относится к каталитической поверхности и описывает химическую кинетику адсорбированных и хемосорбированных частиц. Второй уровень описания относится к гидрогадинамике пограничных слоев – гидродинамического, теплового и диффузионного. Для второго уровня описания для нахождения скорости, температуры, концентраций веществ, образующих газовую смесь используют в качестве гранич-

ных условий, условия на каталитической поверхности условия на границах соответствующих пограничных слоев и ядра потока. Толщины гидродинамического, температурного и диффузионных пограничных слоев не совпадают друг с другом и изменяются по мере продвижения вдоль потока (вдоль каталитической поверхности). Таким образом, изменения скорости, температуры, концентрации веществ в ядре потока доставляет граничные условия для изменений этих величин поперек потока в пограничных слоях. Третий уровень описания относится к описанию процессов проходящих в ядре потока. Также еще одна группа граничных условий сосредоточена на каталитической поверхности и является результатом каталитической реакции, которая происходит на поверхности катализатора.

Для почти всех практических приложений считается, что по температуре в каждый момент времени в каждой точке каталитической среды имеется локальное тепловое равновесие. Это означает, что все вещества, из которых состоит подвергаемая газовой конверсии, газовая смесь, имеют одну и ту же температуру, но разные концентрации.

Существует еще один уровень описания, непосредственно связывающий три уровня с конструкцией реактора. Этот уровень относится к масштабу реактора и описывает гидродинамические, тепловые и концентрационные характеристики ядра потока при течении смеси в реакторе. Для этого уровня описания следует задавать граничные условия по скорости, давлению, темпе-

ратуре, концентрациям на входе в, и выходе из аппарата. Следовательно, все эти уровни описания относятся к открытой системе, которая находится в условиях общего равновесия.

Целью данной статьи является обзор ряда моделей гетерогенного катализа основанных на структурированном, иерархическом подходе, суть которого в последовательном решении задач на различных уровнях описания процессов.

ОСНОВНОЙ МАТЕРИАЛ

Три уровня описания оказывают различное влияние на поведение каталитической реакции. Наибольшее влияние оказывает первый уровень описания – самой каталитической реакции. Второй уровень описания – уровень пограничных слоев, порождает изменения исходя из многокомпонентного характера диффузии веществ в смеси. Третий уровень описания, охватывает случаи течения с локально постоянной, локально увеличивающейся и локально уменьшающейся скоростями. Конкретная степень постоянства, ускорения или замедления потока определяется структурой каталитического носителя. С этой точки зрения все носители можно разбить на две группы: первая – монолитное пористое тело; вторая – неконсолидированное или слабоконсолидированное дисперсное тело. В обоих случаях катализатор покрывает поверхность пор. Движение газовой смеси, подвергаемой конверсии, представляет собой течение смеси переменного состава в пористой среде. Характер течения в порах определяется геометрией пор, гидравлическим сопротивлением носителя в макромасштабе. Распределение температуры в системе: газовая смесь плюс носитель определяется тепловыми характеристиками смеси и носителя. В целом такая система представляет собой композит, характеристики передачи тепла в котором зависят от устройства его элементарной ячейки.

Наиболее сложным и разнообразным является первый уровень описания. Здесь возможно два взаимодополняющих подхода. Первый подход можно охарактеризовать, как такой, в котором изучается изменения характеристик атомов и молекул катализатора и адсорбата и взаимодействие адсорбата и носителя. В подавляющем большинстве случаев этот подход носит экспериментальный характер. Результатом такого подхода

являются такие атомно-молекулярные характеристики, которые определяют химическую активность адсорбата. Второй подход состоит в рассмотрении уравнений поверхности химической кинетики с константами скорости, значения которых определены в рамках первого подхода или из кинетических коэффициентов.

Второй подход позволяет выделить два направления, условно их можно обозначить как: «идеальный» химикокинетический подход и «реальный» химикокинетический подход. Если поверхность зеркально ровная, а все активные центры одинаковы и имеют равную каталитическую активность, то кинетика на такой поверхности называется идеальной. Любое отклонение от этого условия принято относить к реальной кинетике. Разделение на два подхода позволяет выделить в многообразии химико-кинетических явлений составляющую, которая обусловлена только законом действия масс.

Химикокинетический аспект каталитических реакций основан на двух механизмах: Илли-Ридела и Лэнгмюра-Хиншельвуда. По первому механизму поверхностная реакция происходит с реагентами, адсорбированными на поверхности. По второму механизму все реагенты находятся на поверхности в адсорбированном состоянии. Совокупность этих двух механизмов и ровная однородная поверхность составляют принцип «идеальной кинетики».

Для главных факторов реальной кинетики введена следующая классификация: неидеальность адсорбированных и хемосорбированных атомов и молекул; поверхностная диффузия адсорбированных и хемосорбированных атомов и молекул; фазовые переходы (газ – жидкость, жидкость – твердое тело) в адсорбате и хемосорбате; неоднородность поверхности катализатора в отношении его химической активности; влияние локального тепловыделения на скорости поверхностных реакций; перестройка структуры поверхности катализатора под влиянием адсорбата и хемосорбата (без фазовых переходов); объемная диффузия адсорбата и хемосорбата в приповерхностном слое катализатора.

В «идеальной кинетике» считается, что молекулы по поверхности не взаимодействуют между собой, то есть образуют смесь идеальных газов. На самом деле между ними имеются латеральные взаимодействия. Реальные адсорбированные и хемосорбированные атомы или

молекулы приобретают в результате взаимодействия поляризацию. Такие атомы и молекулы взаимодействуют электрически как диполи, квадруполь и т.д. Такое взаимодействие дальше, поэтому может простираться на дальних соседей и влечет за собой изменение констант скоростей реакции на поверхности. В Аррениусовском множителе констант скорости появляется зависимость от поверхностных концентраций адсорбированных, хемосорбированных частиц:

$$k_i = k_{i0} e^{-E_i/RT}, \quad (1)$$

$$k_i = k_{i0} e^{-\frac{E_i \pm f(K)}{RT}}, \quad (2)$$

где θ_A, θ_B доли покрытия поверхности частицами A, B

Поверхностная диффузия частиц (A, B, ...) на поверхности происходит со скоростью, на много порядков превышающей скорость объемной диффузии. Различия в величинах коэффициентов поверхностной диффузии приводит к зависимости долей покрытия поверхности частицами (A, B, ...) от положения на поверхности. В результате на каталитической поверхности возникают рельефы (рис 1) соответствует состоянию «идеальности», (рис 2) неидеальность.

По закону действия масс неоднородность заполнения порождает неоднородность выхода продукта реакции, который следует на конечном участке площади

катализатора находить интегрированием по этой площади.

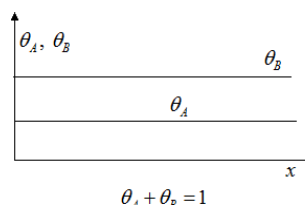


Рисунок 1 – «Идеальное» покрытие поверхности частицами

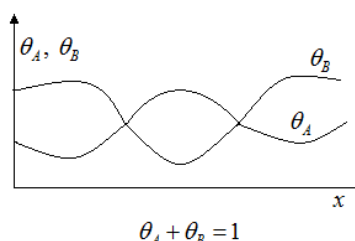


Рисунок 2 – «Неидеальное» покрытие поверхности частицами

Фазовые переходы адсорбированных и хемосорбированных частиц на поверхности, могут приводить к образованию сплошных фаз. В этом случае размерность поверхностной реакции с двух уменьшается до одного. Скорость реакции изменяется во столько раз, во сколько раз число мест сплошной фазы превосходит число мест ее границы (рис.3).

A				B		B	A			
A	A	A	A	A	A	B	A	B		
		B	B	A	B	B	A	B		
				A	B	A	B	B	A	
				B						

(a)

		A	A	A	A					
		A	A	A	A			B	B	
		A	A	A	A			B	B	B
		A	A	A				B	B	B
			A							

(б)

Рисунок 3 – (a) – смесь фаз; (б) – чистые фазы

Неоднородность поверхности катализатора связана с представлением реальной поверхности, которую схематически можно изобразить (рис.4). Цифры означают места на поверхности с различной каталитической активностью. Активности разных мест могут отличаться на порядки. Поэтому следует проводить усреднение по

всем местам. Возможна ситуация, когда слабоактивные места имеют большую долю поверхности, а сильно активные – малую долю поверхности, но вклады в их общую активность будут соизмеримыми. Фактически неоднородность приводит к изменению констант скоростей поверхностных реакций.

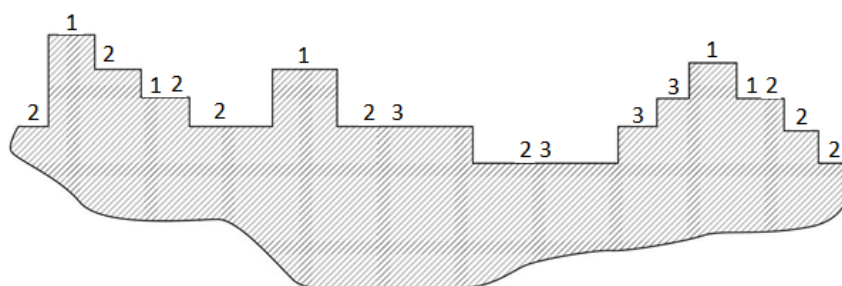


Рисунок 4 – Схематическое изображение неоднородности поверхности катализатора

Влияние локального тепловыделения можно рассматривать с точки зрения значительного теплового эффекта от каталитической реакции. Теплообмен на каталитической поверхности имеет большие значения тепловыделения в местах высокой каталитической активности и меньшие значения в местах низкой активности. Тепловое сопротивление газовой смеси в пограничном слое велико, а в слое катализатора (если он металлической природы) – мало. Часть потока тепла идет в приповерхностный слой катализатора, а заем и в газовую смесь. В этом случае возможны: проплавление наноскопических участков катализатора, повышение свободного объема его атомов и объемная диффузия части адсорбата и хемосорбата в объеме катализатора (приповерхностный слой и носитель). Как следствие – изменение свойств катализатора, появление объемных соединений атомов катализатора и поверхностных частиц. На уровне закона действия масс, в следствии этого изменяются предэкспоненциальные множители и экспоненциальные множители: пороговая энергия и температура поверхностных реакций в Аррениусовской форме коэффициентов скоростей поверхностных реакций.

Суть перестройки структуры поверхности катализатора под влиянием адсорбата и хемосорбата заключается в том, что если поверхностных частиц достаточно много, то они способны изменить характер поверхностного распределения атомов катализатора. Без поверхностных частиц грани катализатора, выходящие на его свободную поверхность имеют поверхностное расположение одной структуры и плотности, а с присутствием поверхностных частиц – другие. При этом может измениться число мест и их активность. Это явление приводит к изменению предэкспоненциальных и экспоненциальных величин в константах скоростей поверхностных реакций, а именно, к зависимости предэк-

споненциальных множителей от долей заполнения поверхности, а также зависимости энергии реакции от долей заполнения по аналогии с неидеальностью поверхностных частиц.

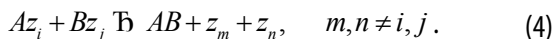
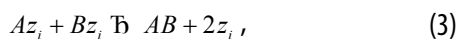
Объемная диффузия адсорбата и хемосорбата в приповерхностном слое катализатора происходит по механизмам твердотельной диффузии-внедрения, если атомы адсорбата и катализатора сильно отличаются по размерам; замещения, если размеры атомов адсорбата и катализатора близки. В последнем случае важную роль играют вакантные места на поверхности. Диффузия в объеме происходит, в основном, по границам зерен (поликристаллитов), где она на порядки превосходит объемную. Попадая в глубину слоя катализатора часть диффузионного потока возвращается в приповерхностный слой из глубины. Причины объемной диффузии могут быть различны: высокоэнергетические молекулы газовой смеси; локальный перегрев каталитической поверхности; выход границ зерен на каталитическую поверхность.

Различные модели в рамках формально-кинетического подхода можно разделить на изотермические модели и неизотермические [1]. В работах изотермического катализа основные исследования посвящены критическому поведению. При этом в основном, изучаются сосредоточенные каталитические системы. Уравнения кинетики формируются с использованием механизмов Илли-Ридела и Лэнгмюра –Хиншельвуда [2]. Эти уравнения бывают дополнены формальными уравнениями предассорбционной стадии (диссоциация), стадий адсорбции и десорбции. Для таких моделей характерным является очень низкое давление газовой смеси. Целью этих работ является определение множественности состояний равновесия и их устойчивости и, по возможности, построения фазового портрета

уравнений кинетики как конкретной динамической системы [2].

Все модели можно разделить на сосредоточенные и распределенные (т.е. зависящие от координат точки на каталитической поверхности). Распределенные модели представляют интерес в разрезе моделирования диффузионной подвижности адсорбированных и хемосорбированных атомов и молекул на каталитической поверхности, а так же с точки зрения описания каталитического процесса на уровне реактора идеального вытеснения. То есть причиной появления пространственной дисперсии есть моделирование на разных уровнях описания, при этом касательно сосредоточенных моделей, они могут включать первый и четвертый уровни описания. Касательно распределенных моделей, то они могут включать в себя все как все четыре уровня описания, так и только первый, и четвертый; первый и второй уровни описания.

Наиболее активно в настоящее время исследуются модели с автокатализом и модели с реакциями или стадиями окисления [1]. В последние два десятилетия сосредоточенные модели подвергались усложнению. Моделирование каталитических реакций на многоцентровых поверхностях, каждый из которых отличается от других величиной своей активности [1]. При этом допускается взаимное превращение различных центров друг в друга. Моделирование такого взаимного превращения можно осуществлять двумя путями: первый – прямое превращение одного центра в другой по схеме $z_i \rightleftharpoons z_j$, а второй – в результате поверхностной реакции:



С точки зрения промежуточных продуктов модели можно разделить на такие, в которых цепочка реакций с данным продуктом сосредоточена только на центрах одного вида, и на такие, в которых промежуточные продукты, получены на разных центрах взаимодействуют друг с другом.

Модели химической каталитической кинетики, которые учитывают неидеальность сорбированных атомов и молекул, основаны на использовании химическо-

го потенциала μ , общий вид которого представляется следующим образом [3]:

$$\mu_i = \mu_{i0}(P, T) + \ln x_{si} + f(x_{sj}), \quad (5)$$

где первые два слагаемых описывают химический потенциал i -го вещества как идеального газа; индекс « s » означает принадлежность к каталитической поверхности; последнее слагаемое описывает латеральные взаимодействия. Это можно осуществить двумя способами. Первый способ заключается в том, что величину $f(x_{sj})$ представляют в виде полинома следующего вида:

$$f(x_{sj}) = \sum a_{f_{mn}} x_{ms} x_{ns} + \sum a_{f_{pqr}} x_{ps} x_{qs} x_{rs} + K, \quad (6)$$

где каждая сумма описывает двойные, тройные, и т.д. взаимодействия между молекулами. Все взаимодействие сосредоточено в коэффициентах $a_{f(K)}$, значения которых определяют по экспериментальным данным. Второй способ заключается в том, что величину $f(x_{sj})$ строят на основании теории межмолекулярных взаимодействий. Здесь требуется привлечение данных о строении сорбированных атомов и молекул.

Распределенные модели, основанные на учете поверхностной диффузии, представляют собой системы уравнений Тьюринга-Пискунова-Петровского и содержат помимо кинетических слагаемых диффузионный оператор, который имеет следующий вид [5]:

$$D \frac{d^2(K)}{dx^2}, \quad (6)$$

или его двухмерный аналог:

$$D \left(z \frac{d^2(K)}{dx^2} - (K) \frac{d^2 z}{dx^2} \right), \quad (7)$$

где D – коэффициент поверхностной диффузии; (K) – означает концентрацию поверхностного вещества [6]. Одномерное выражение формально не

учитывает наличие или отсутствие свободных мест для диффузионных скачков и относится к случаям очень малых степеней покрытия каталитической поверхности. Двухчленное выражение относится к большим степеням покрытия [7].

К распределенным моделям, основанным на диффузии на поверхности примыкают так называемые «островковые модели» [8]. В этих моделях сорбированные атомы и молекулы, все или некоторая их часть, образует сплошные фазы так, что их можно рассматривать как результат фазового перехода сорбированного газа в жидкое или твердое состояния [9]. Химическое взаимодействие между островками осуществляется диффузионным путем с использованием одного или двух механизмов: диффузионного перераспределения островков как целого и взаимодействия островков и сорбированных атомов и молекул в состоянии поверхностного газа. Основной задачей островкового моделирования является не столько исследование свойств моделей, сколько построение убедительного механизма их образования из газовой фазы, которая находится в состоянии идеального перемешивания.

Неизотермические модели представляют собой расширение моделей изотермической кинетики путем добавления уравнения для теплообмена каталитической поверхности с окружающей средой. В качестве окружающей среды используется газовая фаза, примыкающая к каталитической поверхности, и бислойная система, состоящая из слоя катализатора и носителя иной некаталитической природы. Носитель также другим своим краем контактирует с газовой фазой. Если система катализатор плюс носитель имеет металлическую природу, то ее тепловое сопротивление можно считать малым. В системе катализатор – носитель – газовая фаза, поток тепла, выделяющийся на каталитической поверхности, распределяется на две части, примерно, поровну. Если же носитель является теплоизолятором, то тепловые потоки в газовую фазу и носитель неодинаковы. В этом случае используют соответствующий коэффициент распределения. В целом исследование моделей неизотермической кинетики направлено на изучение влияния неизотермичности на число, характер и устойчивость состояний равновесия, которые получаются в рамках сосредоточенных изотермических моделей [8].

Большое число моделей, которые называются макрокинетическими, устанавливают связь между реальным поведением каталитических реакторов и моделями поверхностной кинетики. Эти модели получаются из моделей изотермической и неизотермической кинетик добавлением слагаемых, из которых состоит субстанциональная производная [9]. Объектами субстанциональной производной являются макрокинетические скорость, температура, концентрации газовой смеси. Модели, связанные с поверхностью, формируют для описания с помощью субстанциональной производной источники массы, температуры и концентраций. Такие модели способны учитывать реальную структуру потока. В силу большой вычислительной сложности эти модели фактически исследуются в приближении идеального перемешивания и идеального вытеснения.

К последним достижениям в области математического моделирования каталитических процессов следует отнести: модели, учитывающие корреляции при поверхностной диффузии сорбированных атомов и молекул [10]; модели, учитывающие все отклонения от реальной кинетики с помощью использования моделей присоединения и отражения молекул на каталитической поверхности степени их аккомодации [11]. Последнее время начинают развиваться исследования стохастических моделей, использующие представления о «шуме». В таких моделях возможен учет и вычисление флуктуаций. К самым новым исследованиям стоит отнести и модели, в которых учитывается взаимодействие между сорбированными на поверхности молекулами и атомами и приповерхностным слоем катализатора [12].

Анализ моделей каталитических превращений показывает наличие большого числа моделей, которые можно классифицировать как сосредоточенные и распределенные; изотермические и неизотермические макрокинетические и макрокинетические. Наблюдаемая тенденция к усложнению касается, в подавляющем большинстве случаев, макрокинетических моделей. Вызвано это стремлением приблизить макрокинетические модели к реальности. Сложившаяся на сегодняшний день ситуация свидетельствует о том, что одним из недостатков формально-кинетических моделей является использование при моделировании данных глубокого вакуума и слишком приближенное описание ситуа-



ции при нормальном или высоком давлении. Другим недостатком является существующих моделей можно отнести использование на макроскопическом уровне описания бесструктурных моделей идеального перемешивания вытеснения, а также не учет взаимодействия: сорбат – поверхностный или приповерхностный слой катализатора.

Современные катализаторы, используемые в промышленности, представляют собой или пористое, пронизанное сплошное тело, или совокупность неконсолидированных или слабо консолидированных частиц с межчастичными порами [12]. Способы расположения катализатора отличаются большим разнообразием, которое можно упорядочить следующим образом: это или пленка на поверхности носителя, или нанометрические кристаллические включения, стабилизированные носителем [12]. Исходя из организации, каталитические среды можно различать по тому, состоят ли они из одинаковых элементов или различных. С этой точки зрения каталитические среды, сформированные одинаковым образом из одинаковых элементов можно назвать регулярными. Такие среды, как правило, являются слабо консолидированными. Каталитические среды, состоящие из одинаковых или различных элементов, расположенных в случайном порядке, можно назвать нерегулярными. Нерегулярные среды обычно – неконсолидированные. Модели каталитических реакций третьего и четвертого уровней описания чувствительны к форме и размерам элементов каталитической среды и формируемому или поровому пространству. Имеется очень большое число моделей порового пространства [12]. Все эти модели можно разделить на два больших класса: детерминированные и стохастические модели. Детерминированные модели можно разделить на поровые и капиллярные. Поровые модели обычно используют для описания фильтрующих свойств гранулированных тел с разной степенью консолидации [12]. Капиллярные модели используют для описания сплошных тел [12]. При этом поперечное сечение и форма капилляров могут быть любыми. С целью приблизить капиллярные модели к реальности, вводят понятие извилистости, сужений и расширений. Четких границ между поровыми и капиллярными моделями нет, так как одни могут быть получены из других путем непрерывной деформации элементов, из

которых состоит каталитическая среда. К основному топологическому отличию между поровым описанием каталитической среды и капиллярным описанием каталитической среды следует отнести то, что поры сообщаются между собой, а капилляры в большей степени изолированы друг от друга. Учет анизотропии проникновения пористых сред существенен только для сред, состоящих из элементов с сильно различающимися характерными размерами. При этом эти элементы должны быть расположены регулярным образом. В противном случае внутреннее пространство все равно будет изотропным.

В детерминированных моделях внутренняя поверхность может считаться гладкой или шероховатой. В последнем случае влияние шероховатости учитывается специальными поправками, как это принято в гидравлике [12]. При моделировании течения в порах и капиллярах используют следующие приемы: внутреннее пространство считают неоднородным; или в каждой точке внутреннего пространства пора или капилляр являются результатом усреднения пор и капилляров [12]. В последнем случае внутреннее пространство является однородным, но усредненным в каждой точке.

Стохастические модели предполагают источником стохастичности внутренние поверхности пористой среды [12]. Из-за этого газовая смесь может течь через поры и капилляры турбулентно. Такой режим течения называется псевдо турбулентным. Такое движение можно моделировать, таким образом, будто бы газ течет в случайно изменяющихся от точки к точке границах. При описании используется аппарат и понятия теории турбулентности [12]. Другой модельный подход состоит в том, что течение газа можно рассматривать как поведение легкого газа в тяжелом. В роли тяжелого газа выступают внутренние поверхности пористой среды [12]. Считается, что масса частиц тяжелого газа настолько велика, что при соударении с легким газом он практически неподвижен.

ЗАКЛЮЧЕНИЕ

В настоящей работе сделан обзор моделей гетерогенного катализа по состоянию на сегодняшний день. Гетерогенный катализ представляет собой очень обширную область, лежащую на пересечении таких обла-

стей науки, как химическая кинетика, физическая и химическая адсорбции, теплофизика, гидрогазодинамика, механика сплошной среды и прочее. При выполнении анализа публикаций был дан обзор совокупности физических механизмов, присутствующих при реальном осуществлении каталитических реакций и способы моделирования этих механизмов. Наибольшее число работ посвящено сосредоточенным моделям, в которых изучается структура пространства параметров на предмет определения границ устойчивости и множе-

ственности стационарных состояний, а так же изучаются характеристики колебательных режимов. Значительно число работ отведено распределенным моделям стационарного характера, меньшее – нестационарным распределенным моделям. За последние десять лет растет число работ, основанных на многомерных моделях на по поверхности. Также последнее время появились исследования малоразмерных моделей в подповерхностном объеме.

ЛИТЕРАТУРА:

1. Bikov V. I. Nelineynye modeli khimicheskoy kinetiki. – М.: Krasand, 2011. – 400 s.
2. Yablonsky G. S., Bikov V. I., Elokhin V. I. Kinetika modelnikh reaktsiy geterogenno kataliza. – М.: Nayka, 1986. – 224s.
3. Gorban A. N., Bikov V. I., Yablonsky G. S. Ocherki j khimicheskoy relaksatsii.–H.: Nayka, 1986. – 320s.
4. Kaplan I. G. Vvedenie v teoriu megmolekulyarnih vzaimodeystviy. – М.: Nayka, 1982. – 312s.
5. Bikov V. I. Modelirovanie kriticheskikh yavleniy v khimicheskoy kinetiki. – М.: Nayka, 1988. – 263s.
6. Chebotin V. N. Fizicheskaya khimiya tverdogo tela. – М.: Khimiya, 1982 – 320s.
7. Chebotin V. N. Khimicheskaya diffuziya v tverdiy telah. – М.: Khimiya, 1982 – 315s.
8. Kirillov V. A. Gas-Liquid-Solid reactors with Cocurrent Up- and Down Flow Operations. – Novosibirsk: Publishing House of Siberian Branch of the Russian Academy of Science, 1997 – 485s.
9. Grigorios C. K., Anastasios M. S. Catalytic automotive exhaust aftertreatment // Energy Combustion Science.–1997. – Vol. 23. – PP.1–40
10. Deutschmann O. Modeling and Simulation of Heterogeneous Catalytic Reactions: From the Molecular Process to the Technical System. – Wiley, 2013. – 370p.
11. Dixon A., Deutschmann O. Spatially Resolved Operando Measurements in Heterogeneous Catalytic Reactors, Volume 50 (Advances in Chemical Engineering) 1st Edition. – Academic Press, 2017 – 302p.
12. James J. S., Mayank G. Catalysis: Volume 24 (Specialist Periodical Reports). – Royal Society of Chemistry, – 2012. – 357p.

*Рецензент: д.т.н., проф. Марсанов В. В.
Херсонський національний технічний університет*



ВИКОРИСТАННЯ ПРЕДМЕТНО – ОРІЄНТОВАНОЇ МОВИ І ВІЗУАЛЬНИХ ПІДХОДІВ ДЛЯ ПРОЕКТУВАННЯ СИСТЕМИ АКТОРІВ(АККА)

УДК 004.42+004.436

ГАЛКІН Олександр Володимирович

доцент кафедри інформаційних систем, факультет комп'ютерних наук та кібернетики Київський національний університет імені Тараса Шевченка, кандидат фізико-математичних наук, доцент

Наукові інтереси: інформаційні системи, супералгебри Лі, парасупералгебри, рівняння для частинок з вищими спінами, квантові алгебри, квантова теорія поля.

E-Mail: galkin@unicyb.kiev.ua

ВЕРЕС Максим Миколайович

доцент кафедри інформаційних систем, факультет комп'ютерних наук та кібернетики Київський національний університет імені Тараса Шевченка, кандидат фізико-математичних наук, доцент

Наукові інтереси: паралельні та розподілені обчислення, методи об'єктно орієнтованого програмування, інформаційні системи, управління інформацією.

E-Mail: veres@unicyb.kiev.ua

ЛАРІН Владислав Олегович

аспірант 2-го року навчання (денна форма навчання) кафедри інформаційних систем, факультет комп'ютерних наук та кібернетики Київський національний університет імені Тараса Шевченка

Наукові інтереси: інформаційні системи.

E-Mail: bantysh.oleg@ukr.net

БАНТИШ Олег Віталійович

аспірант 2-го року навчання (денна форма навчання) кафедри інформаційних систем, факультет комп'ютерних наук та кібернетики Київський національний університет імені Тараса Шевченка

Наукові інтереси: інформаційні системи.

E-Mail: vlarinmain@gmail.com

ВСТУП

Сучасний процес проектування, розробки і впровадження великих розподілених інформаційних систем є досить ресурсномістким і складним. Для спрощення цих процесів створюються різні концептуальні підходи, інструментальні ланцюги, рамки та інші. Одним з таких підходів є модель актора, яка базується на використанні окремих функціональних сутностей, що дозволяє досягти спрощення складності проектування та управління складної розподіленої системи великою командою. Інструментарій Акка є найповнішим засобом для

розгортання систем, заснованих на моделі актора. Але він має певні недоліки такі як низькорівневий опис поведінки акторів та їхньої взаємодії, крім того опис та конфігурація переважно знаходяться в одному і тому ж місці. Отже, текст опису систем з багатьма учасниками та сценарії їхньої взаємодії є громіздким і неочевидним для сприйняття. Розгортання таких систем є ресурсномістким процесом, крім того актори не мають вбудованих механізмів компонування, а також мають обмежені можливості статичної типізації повідомлень. Концепція Visual Akka, яка потенційно спрощує

роботу програмістів і системних аналітиків при створенні систем акторів, була створена для подолання перерахованих вище недоліків.

Модель актора і його реалізація в Akka. Модель актора в інформатиці є математичною моделлю паралельних обчислень, яка розглядає «актори» як універсальні примітиви паралельних обчислень. Отримуючи повідомлення, актор може: виконувати локальні дії, створювати додаткових учасників, відправляти повідомлення і визначати, як відповідати на наступне повідомлення. Модель актора виникла в 1973 році. Вона використовувалася як основа для теоретичного розуміння обчислень і теоретична основа для практичних реалізацій паралельних систем. [1]

Актор є основним суб'єктом взаємодії в моделі актора. Всі процеси взаємодії між акторами відбуваються асинхронно, на відміну від об'єктів в об'єктно-орієнтованій парадигми, де взаємодія зазвичай послідовна.

Актор - обчислювальна сутність, яка у відповідь на отримане повідомлення може одночасно:

1. Відправляти визначену кількість повідомлень іншим учасникам;
2. Створити визначену кількість нових учасників;
3. Визначити поведінку, яка буде використовуватися для обробки наступного повідомлення.

Розмежування асинхронного зв'язку та керування передачею повідомлень дає можливість розглядати модель акторів, як паттерн передачі повідомлень. [2,3]

Актори - дуже прості, одночасно функціональні сутності, за їх допомогою збільшуються рівень абстракцій, що спрощує процес написання, тестування, розуміння і підтримки паралельних і розподілених систем. Розробники можуть зосередитися на розробці конкретної задачі, а не на тому як будуть відправлятися повідомлення, на відміну від використання об'єктів більш примітивних та низьких рівнів реалізації, таких як потоки, сокети та інше. [4]

Актори в Akka виконані на так званих «легких» потоках, що дає наступні переваги:

- актори захищені від впливу інших акторів;
- немає необхідності використовувати різні механізми блокування, коли реалізують логіку акторів;
- низьке використання ресурсів дозволяє одночасно використовувати велику кількість акторів.

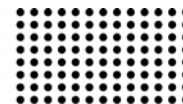
Ще однією важливою особливістю Akka є реалізація так званої контролюючої системи. Коли виникають збої, контролююча система втручається і відповідно до призначеної стратегії може або перезапустити актора, або взагалі зупинити його. При бажанні стан актора може автоматично відновлюватися після перезапуску, при цьому отримані повідомлення зберігаються і оновлюються після перезапуску. Це забезпечує можливість самовідновлення системи. Крім того, кожен актор може контролювати інших акторів, утворюючи ієрархію супервізора. Батьківський актор отримує спеціальне повідомлення, яке включає в себе інформацію про актора, і може самостійно вибрати подальшу стратегію дій.

Мета акторів - обробляти повідомлення. Канал, що з'єднує відправника і одержувача, є поштовою скринькою актора: кожен учасник має тільки один поштовий ящик, на який відправляються всі повідомлення. Передача повідомлень від різних учасників, може мати довільний порядок відповідно до випадкового розподілу потоків між учасниками. Тим не менше, повідомлення відправлені від одного актора іншому, будуть оброблені в порядку їх відправки.

Всі елементи в Akka призначені для роботи в розподілених системах, де всі учасники використовують тільки асинхронні повідомлення. Мета такого дизайну полягає в тому, щоб розроблені функції можна було однаково використовувати як в межах локальної JVM, так і на кластері з сотнями машин, а зміна взаємодії виконувалась через зміну конфігурації проекту. Це виявляється досить зручним - досить написати код за правилами встановленими Akka, і його виконання на одній машині або в межах розподіленого проекту буде залежати лише від конфігурації. Таким чином, додаток можна масштабувати без необхідності зміни коду.

Однак у моделі актора є і деякі недоліки. Модель актора не має вбудованих механізмів компонування декількох акторів. Akka вирішує цю проблему лише частково: опис поведінки актора є тільки в функціональному стилі. Але таке рішення громіздким і важким для модифікації і супроводження системи.

Іншим недоліком системи Akka є її обмежені можливості статичної (на етапі компіляції) типізації повідомлень. Розробник не може сказати, синхронізовані дії між учасниками чи ні, без попереднього тестування системи. Цей недолік не є суттєвою проблемою



для розробки систем з простою конфігурацією. Однак, коли конфігурація стає складнішою це може стати великою проблемою, а статична перевірка типів дозволила б відразу виявити значну кількість помилок.

ВІЗУАЛЬНИЙ АККА

Модель актора була обмежена в своєму використанні до створення Akka Framework. Akka - це повна реалізація моделі актора в JVM, що відрізняється відмінною продуктивністю і надійністю (50 мільйонів повідомлень в секунду, ~ 2,7 мільйона акторів на 1 ГБ RAM) завдяки використанню «легких» потоків і відмовостійкості механізму обробки повідомлень. Крім того, конфігурація системи виносить з коду в конфігураційний файл, також Akka з коробки підтримує кластеризацію.[5]

За короткий період часу Akka отримав величезну популярність і в даний час широко використовується у фінансовій сфері, в засобах масової інформації, соціальних і розважальних послугах.

Як згадувалося вище, у Акки є певні недоліки. Використання візуального підходу - це можливий варіант подолання проблеми складності опису акторів та їх взаємодії. Візуальний підхід забезпечує високу прозорість, але створення діаграм - більш складне завдання, ніж написання коду програмування. Більш того, візуальне середовище розробки підвищує вимоги щодо програмного і апаратного забезпечення. Таким чином, для ефективного вирішення цієї проблеми було запропоновано використовувати предметно-орієнтовану мову і її візуальне представлення для редагування і опису актора.

Предметно-орієнтована мова(DSL) є мовою програмування, що спеціалізуються на конкретній предметній області. Склад такої мови або структури даних охоплює специфіку предметної області, в якій завдання вирішується.

Приклади мов:

1. TeX \ LaTeX - мова розмітки для текстових документів;
2. SQL – мова опису запитів в реляційних базах даних;
3. HTML – мова опису розмітки для веб сторінок;
4. Пролог – мова опису предикатів математичної логіки;

5. ML, Haskell – функціональна мова програмування.

Можна виділити деякі специфічні особливості предметно-орієнтованих мов у порівнянні з мовами загального призначення:

- абстракції DSL забезпечують визначення абстрактних понять в предметній області;
- синтаксис DSL може забезпечити природний опис суб'єктних понять і запобігає синтаксичній неузгодженості, що відбувається, коли використовується мова загального призначення;
- при перевірці описів в DSL потрібні статичні аналізатори, які можуть виявити більше помилок, ніж аналізатори загального призначення, і надавати повідомлення про них на тій же мові, яка більш зрозуміла для експертів в предметній області;
- оптимізація коду за описом в DSL заснована на знанні, яке недоступно для компілятора загального призначення;[6,7]

Візуальне програмування є спосіб створення програм шляхом маніпулювання графічними об'єктами замість текстового написання програмного коду.

Візуальне програмування дозволяє програмувати з використанням графічних або символічних елементів, якими можна маніпулювати в інтерактивному режимі відповідно до певних правил. Крім того, він дозволяє використовувати візуальне представлення графічних елементів в якості елементів синтаксису програми. Значна кількість мов візуального програмування заснована на використанні «боксів і стрілок», прямокутників, овалів і т. д. Розглядаються як сутності, пов'язані стрілками (лініями, дугами і т. д.), так і представляють відносини, наприклад - UML.

Основні принципи реалізації Akka. Основна ідея системи - додати додатковий рівень за допомогою ключових слів предметної області Flow DSL і їх біктивне відображення на візуальну основу. Користувач може вільно використовувати як візуальні, так і текстові підходи. Зв'язки між акторами і їх взаємодія фіксується на цьому рівні, тим самим формуючи їх інтерфейс взаємодії. Після цього визначаються аргументи операцій і їх типи. Це дозволяє перевірити сумісність типів, поєднуючи різних учасників. Елементи конфігурації Akka винесені в параметри візуального актора (або в анотації і ключові слова Flow DSL), реалізована підтримка

маршрутизації, контролю та інших функцій Akka. Кожна діаграма представляє певний стан актора. Вся ця інформація стала підґрунтям для створення високорівнево-

го формату актора, на основі якого генератор коду формує прототипи в Java з можливістю розширення логіки користувача.

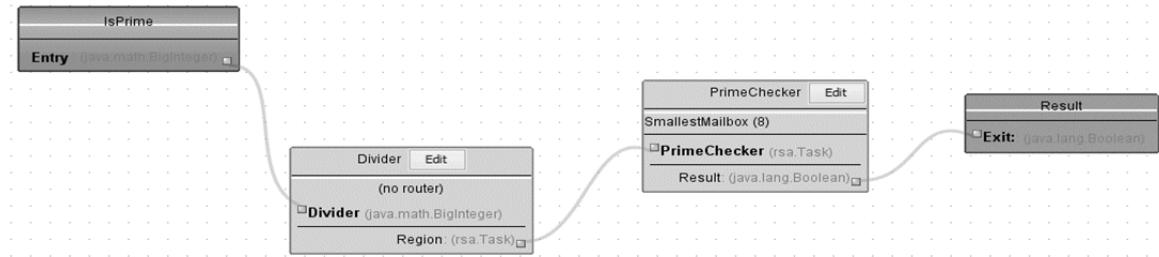


Рис. 1. Приклад візуальної одиниці

Akka Visual Extension (AVE) - це розширення для Netbeans, а також інструментарій для предметно-орієнтованого візуального опису прототипів і програмування акторів Akka, який реалізує вищевказану концепцію. Через додатковий рівень абстракції створені актори можуть використовуватися в інших модулях без необхідності зміни коду. Щоб вирішити проблему типізації взаємодії акторів для кожного входу і виходу в Visual Akka, повинен бути визначений тип, який є основою для генерації класів в Java з динамічною перевіркою передачі повідомлень заданого типу.

Основна частина написана у вигляді проекту Java NetBeans Plugin. Плагін реалізує розширення середовища розробки NetBeans IDE, додаючи підтримку предметної мови для роботи з акторами (включаючи підказку синтаксису), візуалізацію елементів акторів, а також

генераторів коду для перетворення програм DSL Flow і візуальних структур в акторів Akka (на мові Java).

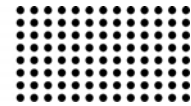
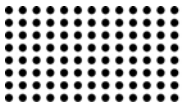
Предметно-орієнтована мова Flow. DSL Flow служить проміжною ланкою між Java-кодом і візуальним представленням. Через специфіку предметної мови можна відобразити і редагувати в візуальних одиницях і навпаки без втрати інформації. DSL Flow забезпечує гнучкість системи і дозволяє як програмістам, так і системним аналітикам працювати з ним. Актори, повідомлення і потоки повинні бути описані за допомогою основних елементів мови. Актор символізує основний елемент системи, яка в свою чергу, містить інтерфейс обробки повідомлень (**message**), є контейнером інших учасників (**routed, group**), можуть мати посилання на інших учасників (**ref**), і містить логіку обробки похибок (**catch**).

```
catch IllegalNumberFormat invoke strategy on st stop
                                on rs restart;

[VisualPosition(x: 10,y: 30)]
[VisualInvocation(id:"generateTasks", x:15, y:20)]
message compute String path : String {
    flow void broadcast workers.init();
    flow path invoke generateTasks() send workers.sum;
    broadcast workers.finish() invoke storeResults() return;
}

message test void : void {
    flow 1 invoke test when isOk send self;
    flow "test" invoke t2 on success send self
                                on fail alter null send self.kill;
```

Рис. 2. Фрагмент коду актора в Flow DSL



Повідомлення містить алгоритм для обробки певного типу повідомлень, які можуть бути отримані за допомогою даного актора. Кожен обробник повідомлень складається з множини потоків (**flow**). Кожен потік, в свою чергу, складається з набору послідовних команд для обробки і передачі інформації. Кожен потік запускається після отримання вхідної інформації. Зазвичай кожен потік запускається асинхронно, однак якщо потік використовує дані, отримані іншим потоком, він буде синхронно очікувати доступність даних. Дані в кожному потоці можна маніпулювати такими способами:

- Почати потік (**flow**) з певними початковими даними. Вхідні дані можуть бути представлені літералом, аргументом обробника повідомлення або посиланням на стороннього суб'єкта;

- Надіслати (**send**) дані для обробки до іншого актора або до цілої групи акторів одночасно (**broadcast**), результати їх роботи повернути в потік;

- Надіслати дані для обробки в метод на мові Java (**invoke**), результат їх обробки повернути в потік;

- Повернути (**return**) результат до потоку як результат роботи обробника повідомлень.

Контейнер актора / посилання на актора (**routed, group, ref**) означає, що кожен актор може містити посилання і безпосередньо створювати інші актори. Повна підтримка акторських груп і пулів в Akka була реалізована за допомогою додаткових атрибутів або стандартного конфігураційного файлу Akka. Кожен запис має своє ім'я, яке можна використовувати в обробниках повідомлень або обробників помилок.

Обробник помилок (**catch**) означає, що Flow DSL реалізує підтримку обробки помилок і системи контролю Akka. У відповідь на певний клас помилок, обробник помилок може виконати певний призначений для користувача код, і він повинен повернути назад одну із стратегій реагування на помилку (**stop, restart, resume, escalate**).

ВІЗУАЛЬНИЙ КОМПОНЕНТ СИСТЕМИ

Ще одним важливим компонентом є модуль візуального відображення акторів. Цей модуль використовує візуальний підхід до програмування для маніпуляції користувача з фреймворком взаємодії у формі діаграми. Таке представлення взаємодії акторів більш

сприйнятливим до аналітики, а в деяких випадках зручніше для роботи. Ключовим аспектом є досягнення бієктивного відображення діаграми в текст мови програмування.

Редактор блоку, який забезпечує візуальну роботу з акторами, реалізований на основі Visual Library, що входить до складу Netbeans RCP [8]. Найважливішим робочим простором програмного рішення є граф, де функціональні об'єкти є вузлами, а їхня взаємодія - гранями. Згідно з принципами Visual Akka, кожен актор може мати кілька обробників різних типів повідомлень і обробників помилок. Домен на панелі інструментів призначений для роботи з обробниками повідомлень (помилки). У робочій області відображається конкретний актор, описаний в даний момент.

Вхід - це об'єкт, який відповідає за аргумент обробника повідомлень. Кожен обробник має рівно один вхід.

Вихід - це елемент, який відповідає за повернення результату обробника повідомлень. На відміну від методів Java, методи модулів Visual Akka можуть мати кілька виходів. Залежно від способу виконання повідомлення можуть відправлятися на один вихід та не відправлятися іншим.

Екземпляр актора - це функціональний елемент, який втілює посилання на актора. У середовищі Akka, в залежності від конкретних налаштувань елемента, екземпляр є одним актором, пулом акторів або прототипом актора. Прикладом є базові композиційні елементи Visual Akka.

Особливості розроблених модулів. Вони майже не відрізняються від екземплярів акторських одиниць зовні, проте мають різні сутності. Розроблений модуль (модуль) є посиланням на певний метод на мові програмування Java. Тобто, якщо входи, виходи і екземпляри блоків використовуються для створення структурної і транспортної логіки актора, то модулі реалізують безпосередньо бізнес-логіку системи в процесі розробки. Для створення модуля надаються візуальні «помічники» (майстри).

Ще однією важливою особливістю є поле «**extract**». Цей оператор застосовується до відношень між елементами. Метод Java, визначений користувачем, буде викликатися для отриманих даних, і результат обробки буде доставлятися далі. За допомогою даної функції

може бути виконано гнучке перетворення даних, що особливо корисно при суворих умовах типізації. IDE повністю підтримує інтеграцію файлів візуальних акторів, надає конструктору підтримку базових операцій (Undo / Redo і т. д.) І інших компонентів для користувача інтерфейсу (вікна навігації, панелі, смуги властивостей і т. д.),

Створені блоки зберігаються у вигляді текстового файлу візуального актора (text / x-vfa). Бібліотека Oracle

Codemodel використовується для генерації коду [9]. При генерації коду Java з коду DSL створюється Java-клас актора, похідний від базового класу Akka UntypedActor. Методи користувачів (native java code calls) реалізуються в окремий клас, який задається ключовим словом **invoke**. Один актор в Flow DSL відповідає одному класу розробленої логіки користувача в Java.

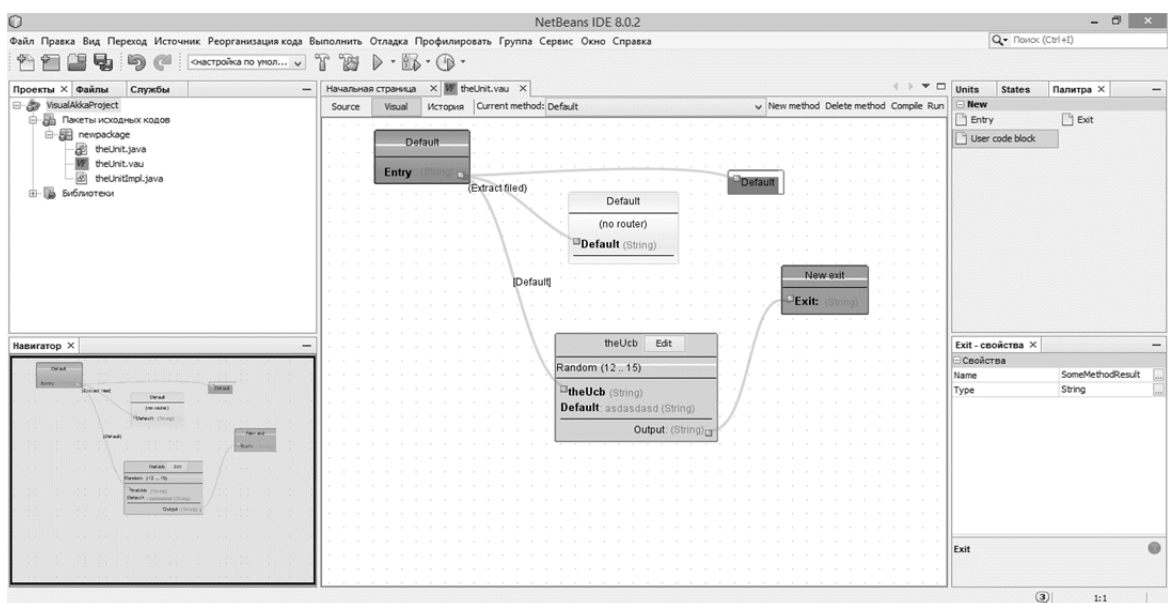


Рис. 3. Основні компоненти Akka Visual

ВИСНОВКИ

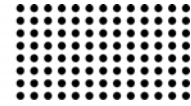
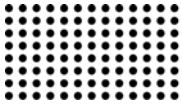
Була досліджена і ретельно проаналізована основа Akka для масштабованої обробки транзакцій в реальному часі, а також розроблено концептуальну модель системи Visual Akka, яка поєднує використання предме-

тної мови та візуального підходу для створення програмних систем в рамках Akka. Запропоноване рішення дозволяє вирішити проблему складності, компонування актора і можливості статичної типізації.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. John C. Mitchell, Concepts in programming languages, Cambridge University Press, 2002
2. Reactive Messaging Patterns with the Actor Model, by Vaughn Vernon, Addison-Wesley Professional, 2015
3. C. Hewitt, «Viewing Control Structures as Patterns of Passing Messages.» Journal of Artificial Intelligence, June 1977.
4. «Akka,» [Online]. Available: <http://akka.io/>.
5. Mastering Akka, by Christian Baxter, PACKT Publishing, 2016
6. Fowler M. Domain-Specific Languages. Addison-Wesley Professional, 1 edition, 2010.
7. Eric Evans. Domain-driven Design: Tackling Complexity in the Heart of Software Addison-Wesley Professional, 2004
8. H. Bock, The Definitive Guide to NetBeans Platform 7, apress.
9. Naman, "Use CodeModel to generate Java Source Code," 2014. [Online]. Available: <http://namanmehta.blogspot.com/2010/01/use-codemodel-to-generate-java-source.html>.
10. T. Inc, Akka Java Documentation, 2014.

Рецензент: д. ф-м. н., проф. О.О. Марченко
Київський національний університет імені Тараса Шевченка



ХАОС ФРАКТАЛ БИРЖА

УДК 004.94

ВЕЗУМСЬКИЙ О.К.

старший викладач, ХНТУ кафедра ІТ

Научные интересы: моделювання, прикладна математика, системний аналіз і структури.

E-Mail: vizir@meta.ua

Уже несколько лет ряд ученых в качестве альтернативы гипотезе эффективного рынка поддерживает гипотезу фрактального рынка. Современная задача по исследованию финансовых рынков в рамках данной гипотезы заключается в модернизации фрактального анализа рынков акций, облигаций и валют, методов нелинейного стохастического процесса с целью выяснения их влияния на инвестиционную политику.

Методами фрактальной геометрии - одного из направлений науки о хаосе - можно за 10 секунд проанализировать любой рынок и узнать, что именно следует предпринимать на нем. [1, 7]

Хаотические (естественные) системы совсем не привлекательны для классической математики: они подчиняются законам нелинейной динамики. Эти явления (феномены) подтверждены наличием сложных петель обратной связи в циклических системах, не периодичность возникновения которых прогнозированию не поддается. Например, если столкнуть один колеблющийся маятник с другим, колебания обоих могут стать "дико" беспорядочными.

К изменению тренда на финансовых рынках приводят неожиданные правительственные заявления, погодные явления, сообщения о видах на урожай, политические или экономические события, происходящие в странах, влияющих на мировую экономику. Такие "толчки" способны запустить любую очевидно хаотическую систему в новом направлении, которое невозможно качественно осмыслить с использованием линейных Ньютоновских инструментов.

Кажется, что такие отклонения создают полностью хаотичный мир. Но, несмотря на чрезвычайную тревогу

ученых, этот неуправляемый, всех нас путающий беспорядок не увеличивается. То, что начинается как безумство беспорядочных импульсов (движение цен на рынках), в конечном счете, принимает форму призрачной геометрии: того, что мы называем странными аттракторами.

(Аттрактор (англ. attract — привлекать, притягивать) — компактное подмножество фазового пространства динамической системы, все траектории из некоторой окрестности которого стремятся к нему при времени, стремящемся к бесконечности.)

Что же такое "странный аттрактор" и как предсказуемая структура может появляться из этого кажущегося беспорядка? Представьте его как идеализированное состояние, к которому непредсказуемым образом тяготеет система. Такая структура образуется вследствие того, что поведение системы (рынка) - лишь отчасти случайная функция. Вернее сказать, система беспорядочно колеблется в пределах специфического диапазона или нормы.

Этот факт меняет сложившиеся представления о хаосе: так называемый "ужасающий беспорядок", когда-то "устраненный" классической физикой, в действительности представляет собой высшую форму порядка.

Рынки - природное явление, и их деятельность не подчиняется законам классической физики, параметрической статистики или линейной математики.

Фрактальная геометрия предлагает абсолютно новый подход к обработке информации. Фрактальная геометрия, один из инструментов теории хаоса, используется для изучения феноменов, которые являются

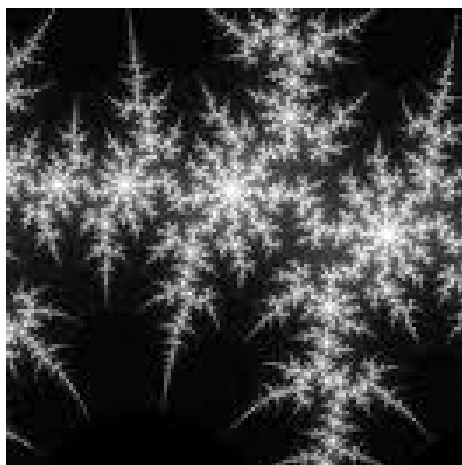
хаотическими только с точки зрения евклидовой геометрии и линейной математики.

Мандельброт и другие ученые обнаружили, что на границе между конфликтами противоположных сил стоит не рождение хаотических, беспорядочных структур, как считалось ранее, а происходит спонтанное возникновение самоорганизации порядка более высокого уровня. [1, 9]

Поскольку рынки - это нелинейные, турбулентные системы, созданные взаимодействием людей, цен и времени действия, то они представляют собой идеальное место, где нужно искать наличие фрактальных структур.

Что ж такое **Фрактал**?

Фрактал (лат. fractus — дробленный, сломанный, разбитый) — сложная геометрическая фигура, обладающая свойством самоподобия, то есть составленная из нескольких частей, каждая из которых подобна всей фигуре целиком (рис.1)



Фракталы являются объектами, с одной стороны сложными (содержащие бесконечно много элементов), с другой стороны — построенными по очень простым законам.

Само слово фрактал может употребляться, когда рассматриваемая фигура обладает какими — либо из перечисленных ниже свойств:

•Обладает нетривиальной структурой . В этом отличие от регулярных фигур (таких, как окружность, эллипс, график гладкой функции): если мы рассмотрим небольшой фрагмент регулярной фигуры в очень крупном масштабе, он будет похож на фрагмент пря-

мой. Для фрактала увеличение масштаба не ведёт к упрощению структуры, на всех шкалах мы увидим одинаково сложную картину.

•Является самоподобной или приближённо самоподобной.

•Обладает дробной метрической размерностью.

Надо теперь определиться с понятием дробной(не целой) размерности.

Для начала приведем некоторые термины из теории измерений:

- Размер

Размер, позволяет сравнивать объекты с однородной структурой,

(например содержимое кошельков), в другом случае он мало информативен,(например при сравнении количества руды в терриконах, один террикон высокий - другой широкий).

Размер можно измерить рулеткой, спичечной коробкой и т.д.

- Мера

Мера тоже служит для измерения объектов, но она измеряется не рулеткой, её главное свойство — мера аддитивна.

Для одномерных объектов мера пропорциональна размеру. Если вы возьмёте отрезки длиной 1см и 3см, «сложите» их вместе, то «суммарный» отрезок будет иметь длину 4см ($1+3=4$ см).

Для не одномерных тел, мера вычисляется по некоторым правилам, которые подбираются так, чтобы мера сохраняла аддитивность. Например, если вы возьмёте квадраты со сторонами 3см и 4см и «сложите» их (сольёте их вместе), то сложатся площади ($9+16=25$ см²), то есть сторона (размер) результата будет 5см.

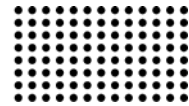
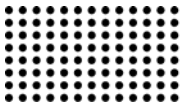
И слагаемые, и сумма являются квадратами. Они подобны друг другу и мы можем сравнивать их размеры. Оказывается, что размер суммы не равен сумме размеров слагаемых ($5 \neq 4+3$).

Как же связаны мера и размер?

- Размерность

Давайте обозначим размерность — D , меру — M , размер — L . Тогда формула, связывающая эти три величины будет иметь вид:

$$M = L^D$$



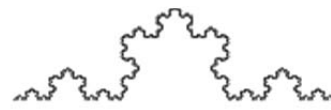
Вывод: если фигуру уменьшить в N раз (отмасштабировать), то она будет укладываться в исходной N⁰ раз.

При уменьшении размера фигуры в N раз, оказалось, что она укладывается в исходной n раз (то есть мера её уменьшилась в n раз), т.е. размерность можно вычислить по формуле:

$$D = \ln(n)/\ln(N) [2]$$

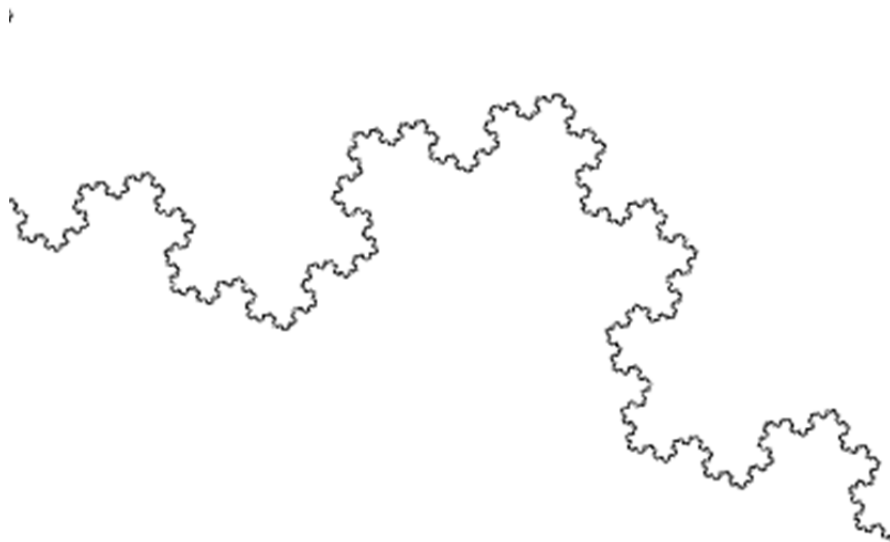
Теперь о собственной дробной размерности.

Построим кривую Коха, которая является типичным геометрическим фракталом. Процесс её построения выглядит следующим образом: берём единичный отрезок, разделяем на три равные части и заменяем средний интервал равносторонним треугольником без этого сегмента. В результате образуется ломаная, состоящая из четырёх звеньев длины 1/3. На следующем шаге повторяем операцию для каждого из четырёх получившихся звеньев и т. д... Предельная кривая и есть кривая Коха. (рис.2), построение снизу вверх.



Эти построения повторяются бесконечное число раз и в конце концов у нас получается ломаная, состоящая из бесконечного числа отрезков.

Сколько бы мы её не масштабировали, мы всё равно будем получать одно и то же.

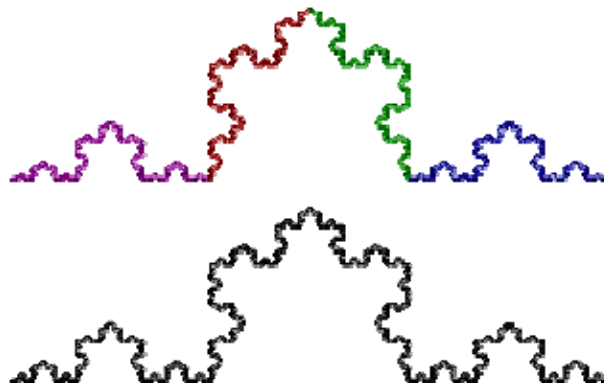
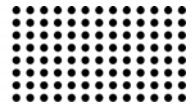
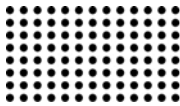


Это и есть звезда Коха

Давайте теперь воспользуемся приёмом, чтобы определить её размерность.

Из построения и рисунка видно, что звезду можно разбить на четыре равные части, при этом размер

(скажем, длина исходного отрезка) каждой части будет равен трети размера исходной фигуры. То есть будучи уменьшена в три раза, она уложится в себе четыре раза: [3, 8]



По ранее представленной формуле получаем, что размерность равна

$$D = \ln(4)/\ln(3) \approx 1.26185950714291487419$$

То есть это уже не просто отрезок или ломаная (длина звезды Коха бесконечна), но и не плоская фигура, полностью покрывающая некоторую площадь.

Мера, как мы помним, аддитивна, то есть мера полного фрактала, равна сумме мер его частей:

$$M_0 = M_1 + M_2$$

И сам фрактал, и его части имеют одинаковую размерность (D) и мы можем выразить меры, через размеры:

$$L_0^D = L_1^D + L_2^D$$

Если принять, что размер полного фрактала 1, то размер зелёной части (полученной из большего отрезка) будет 0.88, а размер красной (полученной из меньшего) — 0.41.

Та формула, которой мы располагаем, уже не годится, так как мы имеем не один, а два коэффициента масштабирования. Но мы можем воспользоваться нашими знаниями о свойствах меры, размера и размерности. Мера, как мы помним, аддитивна, то есть мера полного фрактала, равна сумме мер его частей:

$$M_0 = M_1 + M_2$$

И сам фрактал, и его части имеют одинаковую размерность (D) и можно выразить меры, через размеры:

$$L_0^D = L_1^D + L_2^D$$

Размеры известны. То есть для размерности нашего фрактала можно написать уравнение:

$$1^D = 0.88^D + 0.41^D$$

или просто

$$1 = 0.88^D + 0.41^D$$

Таким образом, если фрактал образован из N подобных элементов, с коэффициентами подобия k_1, k_2, \dots, k_N , то его размерность можно найти из уравнения:

$$1 = k_1^D + k_2^D + \dots + k_N^D$$

По этой формуле уже можно рассчитать размерность многих итерационных систем.

Если все коэффициенты равны, то эта формула превращается в уже известную простую формулу:

$$1 = k^D + k^D + \dots + k^D = N \cdot k^D$$

$$1/N = k^D$$

$$D = \ln(1/N)/\ln(k)$$

или

$$D = \ln(N)/\ln(1/k)$$

Последнее выражение есть первая простая формула для вычисления размерности простейших самоподобных фракталов.

Мандельбротом было замечено, что график цены акций имеет дробную размерность, такую как имеют фрактальные ряды. Отсюда была выдвинута гипотеза о том, что ценовые ряды тоже являются фрактальными и обладают свойствами фрактальных рядов. Анализ ценовых рядов с помощью фрактальной геометрии позволяет по-другому взглянуть на фондовый рынок.

Для определения уровня стохастичности ценовых рядов используют так называемый показатель Хер-

ста. Показатель Херста (Hurst) дает трейдеру два важные характеристики временного ряда. Во – первых, «память рынка» для оценки инертности движения. Память рынка представляется собой глубину ретроспективных данных оказывающих влияние на формирование текущей цены. Следует заметить, что для анализа памяти рынка, по классике статистики, используют автокорреляционную функцию. Во-вторых, показатель Херста является устойчивым, содержит минимальное предположение об изучаемой системе, а главное может идентифицировать вид временного ряда.

В настоящее время наиболее интенсивно изучаются фрактальные множества на комплексной плоскости. Это связано с их использованием при создании математических моделей в разнообразных науках. Комплексные фракталы переплетаются здесь с разработкой алгоритмов, реализуемых с помощью информационных технологий, включая параллельное программирование.

Примером фрактала для использования в биржевой игре является классика фракталов – фрактал Мальденброта

Множество Мандельброта — это множество таких точек z на комплексной плоскости, для которых рекуррентное соотношение
$$z_{n+1} = z_n^2 + c$$

$$z_{n+1} = z_n^2 + c$$
 при $z_0 = 0$ $z_0 = 0$ задаёт ограниченную последовательность. То есть, это множество таких c , для которых существует такое действительное R , что неравенство $|z_n| < R$ выполняется при всех натуральных n . [4]

Вышеуказанная последовательность может быть раскрыта для каждой точки $c = x + iy$ на комплексной плоскости следующим образом:

$$Z_{i+1} = Z_i^2 + c, \text{ число итераций при этом стремится к бесконечности.}$$

Задача биржевого прогноза в общем случае состоит в определении каких-либо количественных или хотя бы качественных характеристик будущего поведения временного ряда на базе имеющихся исторических данных. Локальный фрактальный анализ может быть эффективно использован при решении самых различных задач подобного рода, в частности, при построении

индикатора сильных изменений цены. Такое поведение рынков обычно связывают с обвалами или с корнерами («пузырями»). Соответствующий индикатор основан на доказанном [5] эффекте увеличения крупномасштабных флуктуаций при подавлении мелкомасштабных. Этот эффект является следствием следующих двух фактов. Во-первых, степенной закон для функции выполняется с удивительной точностью на огромном интервале масштабов (от нескольких минут до нескольких лет). Во-вторых, степенная функция обладает особым свойством: чем выше ее рост (по сравнению с функцией с другим степенным показателем) при малом значении аргумента, тем он выше при большом значении последнего. Агрегирование индексов здесь используется для того, чтобы уменьшить число экзогенных факторов, исключив, предварительно влияние фондовых рынков различных стран друг на друга. [6]

Многие считают, что рынки случайны. Тем не менее многие другие утверждают, что, хотя цены могут показаться случайными, они фактически следуют схеме в виде тенденций. А одним из самых действенных способов определения тенденций трейдерами является использование фракталов. Фракталы разбивают большие тенденции на простые и предсказуемые шаблоны разворота. Поэтому каждый, кто научится как торговать на фракталах, имеет шансы значительно увеличить свою прибыль.

Фрактал (от латинского – «fractus») – это индикатор ценовых показателей, демонстрирующий на графике минимальные и максимальные показатели цены. Другими словами фракталы на Форекс – это некий уровень, точка на графике, достигнув которого, показатель цены меняет направление на противоположное.

Фракталы помогают определить наиболее безопасные точки для вступления в сделку.

Цель фрактального анализа – вовремя заметить и правильно интерпретировать фрактальные показатели в сочетании с другими рыночными данными независимо от того, являются ли они фундаментальными или техническими, объёмными или временными. Несмотря на расхожие мнения об их «медлительности», на самом деле реальные фракталы очень динамичны. Их формат может варьироваться в зависимости от движения цен, поэтому появление max/min для фрактала необязательно должно быть последовательным.

Вопрос о принципиальной предсказуемости фондовых инструментов не решен до сих пор. Согласно теории эффективного рынка котировки фондовых инструментов являются случайными величинами, их динамика подобна броуновскому случайному процессу и поэтому получение сколько-нибудь точного прогноза

невозможно. С другой стороны, в литературе приводятся многочисленные свидетельства того, что поведение биржевых котировок не случайно. А потому фрактальный анализ временных рядов, а изменение котировок представлено именно таким рядом, сулит биржевикам хорошие перспективы.

ЛИТЕРАТУРА

1. Б.Вильямс, Торговый хаос – экспертные методики максимизации прибыли. www.koob.ru
2. Кроновер Ричард М. Фракталы и хаос в динамических системах М.: Постмаркет 2000
3. Шредер М. Фракталы хаос степенные законы М.: R&C 2005
4. Везумський О.К., Яровий Ф.В. Дослідження алгоритмів фрактального аналізу у мережі трейдерів фондової та валютної біржі. 2017р
5. X Всеукраїнська науково-практична WEB конференція.
6. Збірник матеріалів 22-34 березня 2017р. Кр.Ріг ст-ка 141
7. О фрактальном анализе хаотических временных рядов.
8. М.М. Дубовиков, Н.В. Старченко http://www.mirkin.ru/_docs/dubov.pdf
9. М.М.Бутовский Технический анализ и фрактальные методы в исследовании финансовых рынков. <https://cyberleninka.ru/article/v/>

*Рецензент: д.т.н., проф. Марасанов В. В.
Херсонський національний технічний університет*



ПІДХОДИ ДО ОБРОБЛЕННЯ «СИНТАКСИЧНОГО ЦУКРУ» ПРИ ПОШУКУ ПЛАГІАТУ В ПРОГРАМНОМУ КОДІ

УДК 004.91

ШЕВЧУК Михайло Михайлович

магістрант кафедри ПЗКС ФПМ НТУУ «КПІ ім.І.Сікорського»
e-mail: myte@ukr.net.

ЮСИН Яків Олексійович

магістрант кафедри ПЗКС ФПМ НТУУ «КПІ ім.І.Сікорського»
e-mail: yusin.yakiv@gmail.com.

ЗАБОЛОТНЯ Тетяна Миколаївна

к.т.н., доцент кафедри ПЗКС ФПМ НТУУ «КПІ ім.І.Сікорського»
e-mail: tetiana.zabolotnia@gmail.com.

РИБАЧОК Наталія Антонівна

к.т.н., ст.викладач кафедри ПЗКС ФПМ НТУУ «КПІ ім.І.Сікорського»
e-mail: rybachoknata@gmail.com.

ДИЧКА Андрій Іванович

магістрант кафедри ПЗКС ФПМ НТУУ «КПІ ім.І.Сікорського»
e-mail: andriydychka@gmail.com.

ВСТУП

Задача пошуку плагіату в програмному коді стає все більш актуальною для різних галузей людської діяльності. В першу чергу, це стосується галузі комерційного розроблення програмного забезпечення і освітньої галузі. Наприклад, одним з найбільш відомих прецедентів щодо копіювання програмного коду в комерційній галузі є судовий процес компанії Oracle проти Google [1]. В ході цього процесу Google була визнана винною в копіюванні 9 рядків коду мовою Java (функції rangeCheck, яка надалі отримала поширену назву «найбільш відомі 9 рядків коду» [2]).

Через поширення на ринку ПЗ явища так званих «open source» проектів підтримка пошуку плагіату в програмному коді стала вкрай необхідною для ефективного вирішення питань щодо захисту інтелектуальної власності та ліцензування. В мережі Інтернет явище

розроблення через копіювання коду навіть отримало назву «Stack Overflow Driven Development» [3] (від назви сайту stackoverflow.com – найбільш популярної веб-системи питань та відповідей про програмування).

Існуючі методи пошуку плагіату в програмному коді або не враховують можливість наявності в ньому «синтаксичного цукру», або є досить обчислювально складними і потребують великого обсягу додаткової пам'яті для роботи.

Таким чином, можна зробити висновок, що розроблення нових підходів, методів та алгоритмів пошуку плагіату у вихідному коді програмного забезпечення є актуальною задачею.

ПОСТАНОВКА ЗАДАЧІ

Задача пошуку плагіату в програмному коді має свої особливості, які роблять неможливим використан-

ня для її вирішення класичних методів пошуку плагіату в текстових даних. Це пов'язано з тим, що в кодї цілком можливі перестановка блоків операторів, перейменування змінних та функцій, додавання коментарів тощо, і ці дії не впливають на коректність програми та можуть слугувати для маскування факту плагіату.

Зазвичай виділяють такі чотири типи схожості фрагментів коду [4], коли:

- два фрагменти коду повністю ідентичні, а змінюватись можуть лише коментарі, пробільні символи та відступи;
- два фрагменти коду ідентичні за структурою та синтаксисом за винятком імен ідентифікаторів, типів, а також відмінностей, що потрапляють до першого типу;
- один фрагмент коду отримано шляхом копіювання другого та додаванням та/або видаленням з нього операторів мови програмування;
- два фрагменти коду повністю синтаксично відрізняються один від одного, але при цьому виконують однакові дії.

Недоліком даної класифікації є те, що вона не розглядає наявність в кодї так званого «синтаксичного цукру», що на сьогодні присутній майже в будь-якій мові програмування високого рівня.

«Синтаксичний цукор» - це загальноприйнятий термін для позначення доповнень синтаксису мови програмування, які не додають їй нових функціональних можливостей, а є більш зручними замінами інших конструкцій, вже наявних в мові [5]. «Синтаксичний цукор» може бути використаний для маскування плагіату програмного коду без особливих зусиль (наприклад, популярні інструменти для розроблення програм автоматично пропонують заміни конструкцій мови на «синтаксичний цукор»), проте відповідно до розглянутої класифікації такі фрагменти коду потрапляють тільки до четвертого типу – найменш схожого на плагіат.

Таким чином, **метою** даної роботи стало підвищення ефективності пошуку плагіату в програмному кодї за критерієм точності завдяки врахуванню можливої наявності в кодї «синтаксичного цукру».

Відповідно до вказаної мети в роботі поставлені і розв'язані такі **задачі**:

- вивчення існуючих методів пошуку плагіату в програмному кодї;

- розроблення підходів до оброблення «синтаксичного цукру» при пошуку плагіату в програмному кодї;
- формування переліку «синтаксичного цукру» для використання при реалізації запропонованих підходів при пошуку плагіату в програмному кодї на мові C#.

«СИНТАКСИЧНИЙ ЦУКОР» ТА ІСНЮЧІ МЕТОДИ ПОШУКУ ПЛАГІАТУ В ПРОГРАМНОМУ КОДІ

Існуючі методи пошуку плагіату в програмному кодї прийнято поділяти на три великі групи [6]:

- текстові методи;
- структурні методи;
- семантичні методи.

Даний перелік не включає в себе методи, які розглядають програмний код в первинному вигляді, без будь-яких його додаткових перетворень. Це пов'язано з тим, що такі методи працюють з програмним кодом як зі звичайним текстом, через це забезпечуючи занадто низьку точність оброблення вхідних даних. У зв'язку з цим автори статті вважають їх розгляд недоцільним.

Текстові методи пошуку плагіату в програмному кодї базуються на розгляданні коду як послідовності tokenів – символів, що представляють собою оператор або групу операторів мови програмування [7]. При цьому параметри цих операторів повністю ігноруються.

Токенізація (процес перетворення програмного коду на послідовність tokenів) методами цієї групи виконується наступним чином:

1. Кожному оператору мови програмування, який не є операндом, приписується цифровий код, що раніше був обраний для відповідного класу операторів. Також коди можна приписувати блоковим операторам (в мові програмування C#, наприклад, це фігурні дужки) та підключенням бібліотек (для C# – просторів імен).

2. Будується рядок з отриманих кодів, зберігаючи порядок їх слідування відповідно до порядку в програмному кодї.

На основі результатів аналізу цієї процедури можна зробити висновок, що дані методи пошуку плагіату дозволяють ефективно працювати з фрагментами коду, що належать до перших двох типів розглянутої класифікації. Разом з тим слід зазначити, що текстові методи були першими методами пошуку плагіату в програм-



ному коді і на сьогодні до них належать найбільш сучасні та ефективні методи [7].

Методи, що базуються на дослідженні структури програми (яка часто представляється ними у вигляді графу потоку керування або абстрактного синтаксичного дерева), відповідно отримали назву *структурних методів* [8]. Методи даної групи також є ефективними для оброблення фрагментів коду перших двох типів розглянутої класифікації і непогано працюють з кодом, що відноситься до третього типу. Головним недоліком методів пошуку плагіату в програмному коді цієї групи є їхня обчислювальна складність і потреба в досить великій кількості додаткової пам'яті для зберігання відповідного подання програмного коду (графу потоку керування або абстрактного синтаксичного дерева).

Семантичні методи є подібними до структурних методів, проте в їх основі лежить знання семантики операторів [9]. Наприклад, методами цієї групи може використовуватись подання програмного коду у вигляді особливого семантичного графу, що має вершини двох типів. Вершини одного типу будуються на основі послідовностей операторів, що характеризуються певною семантикою (наприклад, це можуть бути арифметичні операції, умовні оператори, оператори циклу тощо). Вершини іншого типу задають відношення, в якому перебувають сусідні з ними вершини – таким чином з'єднуючи пари вершин першого типу.

Як вже було зазначено в попередньому розділі, під «синтаксичним цукром» розуміється будь-який елемент синтаксису мови програмування, що дублює існуючий аналог, але є більш зручним у використанні, або більш коротким при записуванні, або виглядає більш природнім чи більш зрозумілим при читанні програмного коду. Таким чином, «синтаксичний цукор» призначений лише для того, щоб зробити мову програмування більш зручною для програміста.

За своїм визначенням найбільш стійкими до заміни операторів на «синтаксичний цукор» для маскування плагіату є структурні та семантичні методи. Проте ці методи, як зазначено, є обчислювально складними і часто потребують залучення великої кількості додаткової пам'яті. Текстові методи є більш привабливими з точки зору обчислювальної складності та необхідної пам'яті, проте є зовсім нестійкими до маскування плагіату за допомогою «синтаксичного цукру».

На думку авторів, вирішенням проблеми неможливості виявлення плагіату в програмному коді, що містить «синтаксичний цукор», текстовими методами може стати розроблення та практичне впровадження нових підходів до попереднього аналізу програмного коду на наявність в ньому «синтаксичного цукру» та подальшої нейтралізації впливу останнього на результати процедури пошуку плагіату.

ПІДХОДИ ДО ОБРОБЛЕННЯ «СИНТАКСИЧНОГО ЦУКРУ» ПРИ ПОШУКУ ПЛАГІАТУ В ПРОГРАМНОМУ КОДІ

Принциповими ознаками «синтаксичного цукру» є те, що [5]:

- 1) все, що може бути написаним за допомогою використання «синтаксичного цукру», може бути написаним без нього цією ж мовою програмування;
- 2) заміна частини програмного коду на «синтаксичний цукор» не змінює хід виконання програми.

Також слід відмітити, що конструкції «синтаксичного цукру» є попередньо відомими і вони чітко визначаються у коді.

На цих ознаках базуються два наступні запропоновані підходи до оброблення «синтаксичного цукру». Обидва підходи передбачають певні дії на етапі попереднього оброблення програмного коду перед виконанням будь-якого методу пошуку плагіату у програмному коді і розглядають останній як звичайний текст.

Reduce. Основна ідея цього підходу полягає в заміні базових конструкцій мови програмування, що мають еквіваленти у вигляді «синтаксичного цукру», на «синтаксичний цукор».

Map. Основна ідея цього підходу є протилежною до *Reduce*-підходу, і полягає в заміні операторів «синтаксичного цукру» на їх еквіваленти, що використовують базові конструкції мови програмування.

Свої назви дані підходи отримали відповідно до результату їх застосування:

- у зв'язку з тим, що в більшості випадків код з використанням «синтаксичного цукру» є більш коротким, підхід, що передбачає заміну базових конструкцій мови програмування на «синтаксичний цукор», начебто «стискає» програмний код – звідси походить назва **Reduce**;

– протилежний підхід в більшості випадків збільшує обсяг програмного коду, «відображаючи» «синтаксичний цукор» на базові синтаксичні конструкції мови програмування – звідси походить назва *Map*.

Заміна базових конструкцій мови програмування на «синтаксичний цукор» та навпаки може бути реалізована по-різному. Наприклад, можливе використання регулярних виразів для пошуку та заміни конструкцій. Також очевидним є те, що дані підходи потребують попередньо створеного списку «синтаксичного цукру» конкретної мови програмування та його еквівалентів, записаних базовими конструкціями цієї ж мови.

Результатами реалізації даних підходів є те, що всі оператори програми приводяться до одного виду – або з використанням «синтаксичного цукру», або без. Отже, у випадку, якщо плагіат програмного коду було замасковано за допомогою «синтаксичного цукру», після оброблення програмного коду буде отримано його початкову форму і плагіат буде знайдено при подальшому виконанні будь-якого методу пошуку плагіату.

Загальний алгоритм використання запропонованих підходів оброблення «синтаксичного цукру» при пошуку плагіату програмного коду буде мати такий вигляд:

1. Читання сирцевого коду, який потрібно перевірити на наявність в ньому плагіату.

2. Читання коду, який є потенційним джерелом плагіату.

3. Попереднє перетворення обох кодів відповідно до одного з запропонованих підходів (Map або Reduce) до оброблення «синтаксичного цукру» в кодї програмного забезпечення.

4. Виконання основного класичного методу пошуку плагіату в програмному кодї.

ПЕРЕЛІК КОНСТРУКЦІЙ

«СИНТАКСИЧНОГО ЦУКРУ» МОВИ C#

Поняття «синтаксичного цукру» в деякій мірі є умовним – різні конструкції можуть бути як віднесені до «синтаксичного цукру» мови програмування, так і не бути віднесеними до нього. Наприклад, більшість сучасних мов програмування як високого, так і низького рівня, мають декілька конструкцій для запису циклів (з умовою, без умови, з лічильником, з післяумовою) або декілька умовних конструкцій. Віднесення однієї з них до базових конструкцій, а інших варіантів до «синтакси-

чного цукру» є помилковим і часто не доцільним, особливо при пошуку плагіату. До таких же конструкцій можна віднести різноманіття арифметичних операторів: наприклад, оператор `+=` та подібні до нього оператори формально підпадають під визначення «синтаксичного цукру», проте вони є досить низькорівневими. В той час, як програмний код стає все більш складним та високорівневим, доцільним є віднесення до «синтаксичного цукру» саме конструкцій високого рівня абстракції для пошуку плагіату. Розгляд «синтаксичного цукру» лише високого рівня також дозволяє зменшити кількість хибно-позитивних помилок, таким чином підвищуючи якість отримуваних результатів за різними оцінками (наприклад, за такими, як точність та F-міра).

До «синтаксичного цукру» мови програмування C# версії 7.0 та вище [10], за допомогою якого можливе маскування плагіату програмного коду, автори пропонують відносити:

- 1) оператор `null`-об'єднання;
- 2) неявну типізацію локальних змінних;
- 3) ініціалізатори колекцій;
- 4) ініціалізатори об'єктів;
- 5) лямбда-вирази;
- 6) методи розширення;
- 7) автоматичні властивості;
- 8) іменовані параметри;
- 9) `null`-умовні оператори;
- 10) лямбда-визначення функцій;
- 11) інтерполяція рядків;
- 12) `out` змінні;
- 13) опціональні параметри.

Перелік конструкцій «синтаксичного цукру» мови C# та відповідні йому базові конструкції наведено в табл. 1.

До окремого переліку слід включити «цукор», що відноситься до LINQ (Language Integrated Query). Цей механізм платформи .NET додає до мов програмування, що її підтримують (в тому числі і до мов програмування C#), синтаксис мови запитів (подібну до SQL), яка дозволяє оперувати даними з різних джерел – бази даних, пам'яті, XML і т.д.

LINQ в мові програмування C# можна використовувати двома шляхами:

- за допомогою операторів мови;
- за допомогою методів розширення.

Таблиця 1.

«Синтаксичний цукор» мови C#

№	«Синтаксичний цукор»	Еквівалент
1	<code>return obj1 ?? obj2;</code>	<code>return obj1 != null ? obj1 : obj2;</code>
2	<code>var obj = 1</code>	<code>int obj = 1</code>
3	<code>string[] arr = {"A", "B"}</code>	<code>string[] arr = new string[2]; arr[0] = "A"; arr[1] = "B";</code>
4	<code>Customer c = new Customer { Name = "James", Age = 30 };</code>	<code>Customer c = new Customer(); c.Name = "James"; c.Age = 30;</code>
5	<code>obj.Act(x => x)</code>	<code>obj.Act(delegate(obj x) { return x; })</code>
6	<code>x.Do()</code>	<code>StaticClass.Do(x)</code>
7	<code>public string A { get; set; }</code>	<code>private string a; public string A { get { return a; } set { a = value; } }</code>
8	<code>Method(a: 10, b: 25)</code>	<code>Method(10, 25)</code>
9	<code>var obj1 = a?.Do(); var obj2 = arr?[0];</code>	<code>var obj1 = a != null ? a.Do() : null; var obj2 = arr != null ? arr[0] : null;</code>
10	<code>public object Do() => new object();</code>	<code>public object Do() { return new object(); }</code>
11	<code>var s = \$"{obj.ToString()}"</code>	<code>var s = string.Format("A{0}", obj)</code>
12	<code>int.TryParse("1", out int value);</code>	<code>int value; int.TryParse("1", out value);</code>
13	<code>public void Method(int a = 5); Method();</code>	<code>public void Method(int a); Method(5);</code>

Методи розширення виступають в ролі «синтаксичного цукру» для операторів LINQ, таким чином вони також відносяться до області застосування запропонованих підходів до пошуку плагіату у вихідному програмному коді. Причиною їх виділення в окремий перелік (див. табл. 2) авторами цієї роботи є те, що перетворення методів розширення LINQ на оператори та навпаки потребує розроблення та застосування більш складних та комплексних правил для того, щоб зберегти правильний порядок оброблення даних та викликів.

REFERENCES

1. Google vs. Oracle: sud dlinoj v sem' let [Electronic Resource]. – Access mode: <https://ain.ua/special/google-vs-oracle/>. – Title from the screen. – (application date: 23.03.2018).
2. Google's 9 lines [Electronic Resource]. – Access mode: <https://majadhondt.wordpress.com/2012/05/16/googles-9-lines/>. – Title from the screen. – (application date: 23.03.2018).
3. Stack Overflow Driven Development [Electronic Resource]. – Access mode: <https://dzone.com/articles/stack-overflow-driven-development-sodd-its-really>. – Title from the screen. – (application date: 23.03.2018).

Таблиця 2.

Оператори та методи розширення LINQ

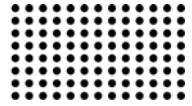
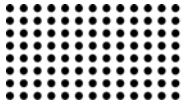
№	Метод розширення	Відповідний оператор
1	GroupBy	group ... by
2	Join	join ... in ... on ... equals ...
3	OrderBy	orderby ...
4	OrderByDescending	orderby ... descending
5	Select	select
6	SelectMany	декілька операторів from
7	ThenBy	orderby ..., ...
8	ThenByDescending	orderby ..., ... descending
9	Where	where

Перспективним напрямом подальшої роботи над цією задачею автори вважають актуалізацію переліку конструкцій «синтаксичного цукру» у відповідності до нових версій мови C# та змін, що вони несуть, а також розгляд можливості застосування запропонованих підходів до оброблення «синтаксичного цукру» при пошуку плагіату в програмному коді, написаному іншими мовами програмування.

ВИСНОВКИ

Таким чином, в рамках даного дослідження показана необхідність оброблення «синтаксичного цукру» при пошуку плагіату в програмному коді і запропоновано два підходи до його визначення на етапі попереднього оброблення коду. Запропоновані підходи орієнтовані на текстові методи пошуку плагіату та не потребують структурного подання програми або її скопійованого вигляду в машинних кодах. Реалізація цих підходів при пошуку плагіату у вихідному програмному коді дозволить підвищити точність автоматизованого визначення наявності запозичень в коді.

Також наведено перелік «синтаксичного цукру» мови програмування C# та його еквіваленти в базових конструкціях, що є необхідним для практичного застосування запропонованих підходів.



4. Roy C. K. and Cordy J. R.. A survey on software clone detection research, Tech.Rep. 2007-541, School of Computing, Queen's University, Kingston, Ontario, Canada, 2007., pages 43-59.
5. Syntaksynnyi tsukor [Electronic Resource]. – Access mode: [https:// goo.gl/uaP469/](https://goo.gl/uaP469/). – Title from the screen. – (application date: 23.03.2018).
6. Prechelt L., Malpohl G., Philippsen M. JPlag: Finding plagiarisms among a set of programs. // Technical Report No. 1/00, University of Karlsruhe, Department of Informatics. March 2000.
7. Huang X., Hardison R.C., Miller W. A space-efficient algorithm for local similarities. // Computer Applications in the Biosciences 6. 1990. P. 373–381.
8. Baxter I., Yahin A., Moura L., Anna M.S., Bier L. Clone Detection Using Abstract Syntax Trees. // Proceedings of ICSM. IEEE. 1998.
9. Moussiades L.M., Vakali A.P Detect: A Clustering Approach for Detecting Plagiarism in Source Code Datasets. // The Computer Journal Advance Access. June 24, 2005.
10. C# Reference [Electronic Resource]. – Access mode: <https://docs.microsoft.com/en-gb/dotnet/csharp/language-reference/>. – Title from the screen. – (application date: 23.03.2018).

*Рецензент: д.т.н, проф. І.А. Дичка
НТУУ «КПІ ім. І.Сікорського»*



POWER MANAGEMENT FOR SERVER CLUSTERS HARDWARE

UDC 681.3.093:044.3

ZUEV D. O.

International IT Expert Independent Consultant USA, Colorado
e-mail: root@dzuev.pro

DOS E.V.

Lead DevOps Architect EPAM Belarus, Minsk
e-mail: bokarevdos@gmail.com

KROPACHEV A. V.

Automation Solution Department Manager Bell Integrator, USA, Colorado
e-mail: beckett@protonmail.ch

BABKIN O. V.

Bell Integrator Cloud Infrastructure Department Manager Moscow, Russia
e-mail: o.v.babkin@gmail.com

VARLAMOV A. A.

SharxDC LLC CTO Moscow, Russia
e-mail: varlamov.it@gmail.com

INTRODUCTION

Low performance of data center infrastructure work is usually associated with disproportion of servers' utilization. Statistically less than 30% of data servers are under-utilized more than 90 % of the total time while other servers normally cause idle power consumption which leads to the 50% of power loss and inefficient work of data center during peak period [1-5]. Overconsumption problem could be solved by minimization of the active servers' number within the bounds of the server consolidation procedure. Server consolidation is server virtualization technique, which allows workloads encapsulating as virtual machines (VMs) and, thus, run multiple VMs at single server with the aid of hypervisor block (Figure 1).

Main task of data center server consolidation implementation is maintenance of the prior performance level of the servers room infrastructure work. It leads to necessity of

the data center peak utilization regime analysis, usually at the 90%, 95% and 99% of maximal recorded value threshold level. Threshold level should be based on the recorded sample in order to get a compromise between stable data center work and opportunity for power savings which is associated with skipping of rare cases of servers' peak load [6, 7]. Thereby, server consolidation may cause performance degradation due to the conflict of using shared resources by VMs [8, 9], specifically last level cache (LLC). The results of cache co-located VMs usage analysis show that sharing LLC between two copies of VMs leads to 20%-30% performance decreasing. The amount of interference could be characterized with a set of parameters, such as effective number of used sets [10]. Allocation of VMs, thus, can be realized by accounting for the amount of the interference and its minimization through the estimation of the required performance requirement.

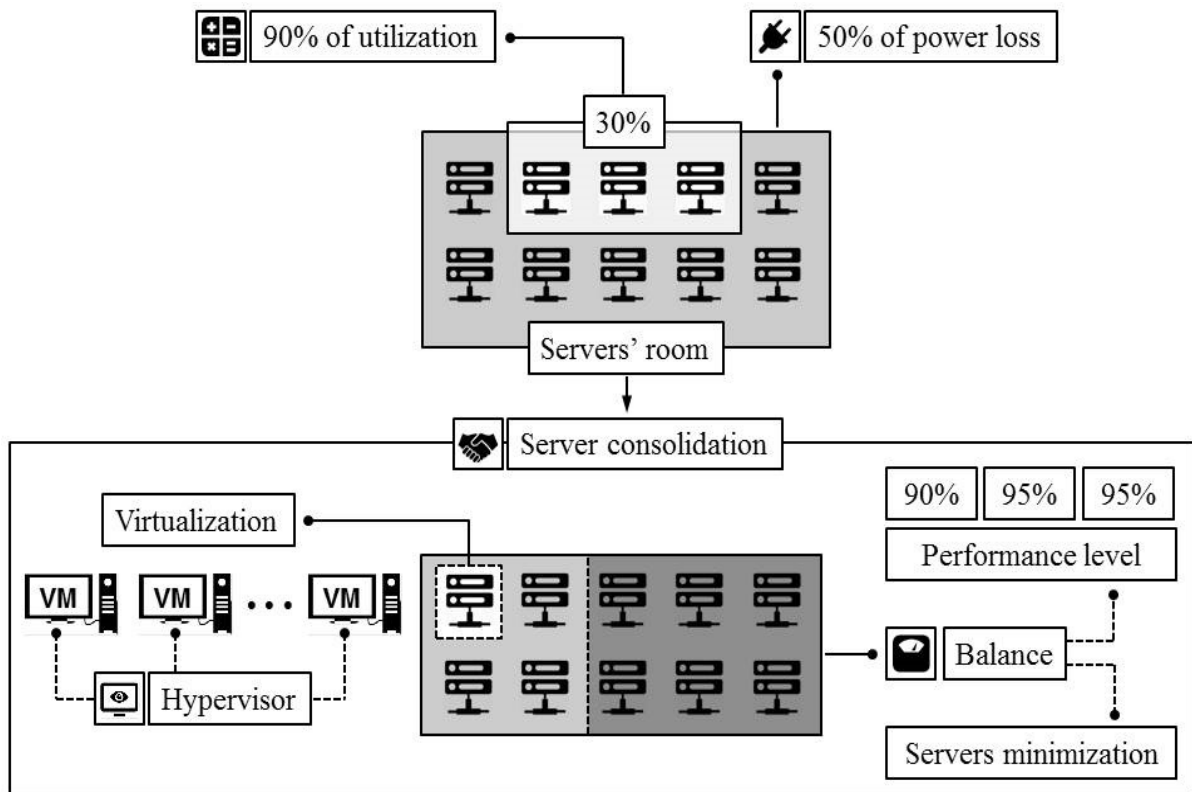


Fig. 1. Data center server consolidation scheme

To develop data center server consolidation methodology there were analyzed recent studies and publications. Statistics of modern data servers' physical resources utilization rate and proportions was considered [1-5]. In order to overcome restrictions and optimize power consumption models that based on the resources utilization threshold value rather than the peak value were discussed [6-7]. Sharing of the servers' resources among co-located VMs, especially LLC problem were studied [8, 9]. To develop the methodology co-located VMs interference with a set of parameters, particularly effective number of used sets was analyzed [10], as well as correlations among VMs' workload [11-14]. Finally it was studied power management solution for data centers scale-out application and targeting distinctive workload characteristics of scale-out applications [15, 16].

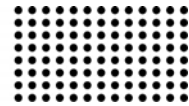
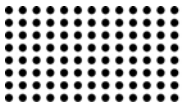
KEY ASPECTS OF THE CORRELATION-AWARE POWER MANAGEMENT

As it was mentioned above, server consolidation could be achieved by considering correlation among workload

variation. Basic scheme of clustering-based correlation-aware VM development [11, 16] solution includes:

- trace data center servers' physical resources utilization level;
- transform utilization traces into binary sequence up to the utilization threshold value;
- clustering of VMs up to the binary sequence in order to maintain not overlapping of different clusters;
- VMs allocation at physical servers in order to minimize the possibility of the service performance degradation at peak period.

Typical engineering solution is pairing of two uncorrelated VMs into super-VM. Maintaining of the super-VMs can be done by predicting of the aggregated workloads. Once two uncorrelated VMs are paired correlations of them within a same super-VM have not be considered, and possibility of further power consumption decrease will be lost. If servers' utilization is perfectly known this scheme could be extended by utilization of multiple VMs workload of such that for VM placement. To overcome those drawbacks it was proposed [15, 16] to develop power management procedure for cloud services that includes:



- user-interactive and fast changing service;
- maintaining of the minimal performance degradation caused by physical resources sharing conflict;
- high correlation level of VMs.

While the scale-out applications usually operate as highly parallel processes, it is advisable to assign the right number of CPU cores for each VM. At Figure 2 are demonstrated generalized results of recent studies [6, 17] of response time of a websearch cluster with respect to the number of queries for 90% threshold value. The number of allocated cores varied from 4 to 16. It should be noticed

that resource utilization level depends on time and usually is lower than the available amount of resources, though dynamic power gating cannot be applicable to this type of applications due to the performance degradation caused by the unapropriable transition delay of power modes switching. Thereby it should be noted necessity of allocating the right number of cores for each VM according to its peak and off-peak resources utilization demands. This procedure has to be implemented at the stage of scaling voltage/frequency level (V/F level).

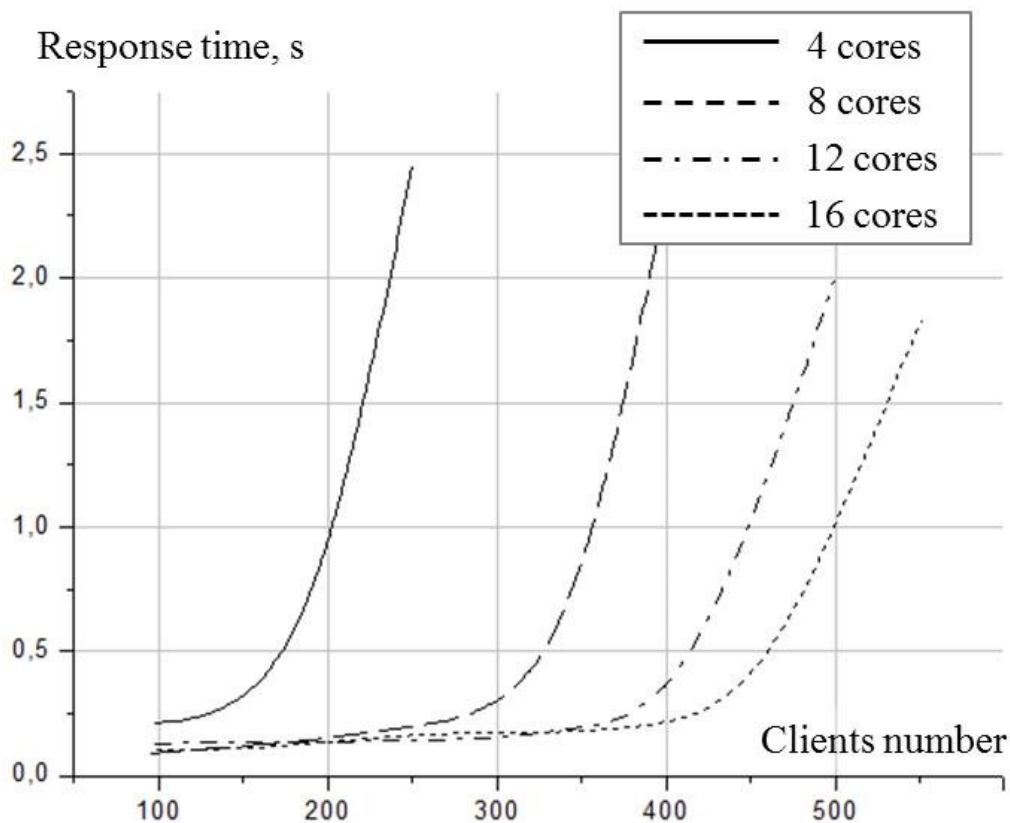


Fig. 2. Response time of cluster with respect to the number of queries for 90% threshold value

Dynamics of the websearch cluster's CPU utilization level is shown at Figure 3. CPU utilization level was traced for 2 VMs with respects to the number of clients' queries. CPU utilizations of both VMs are synchronized with the variation of the number of queries and it could be seen that loads

between VMs are not perfectly balanced. Therefore it should be mentioned that resource utilization efficiency has to be improved by sharing cores among multiple VM.

This procedure will provide more flexible use of the core cores up to the resource demands as a real-time function.

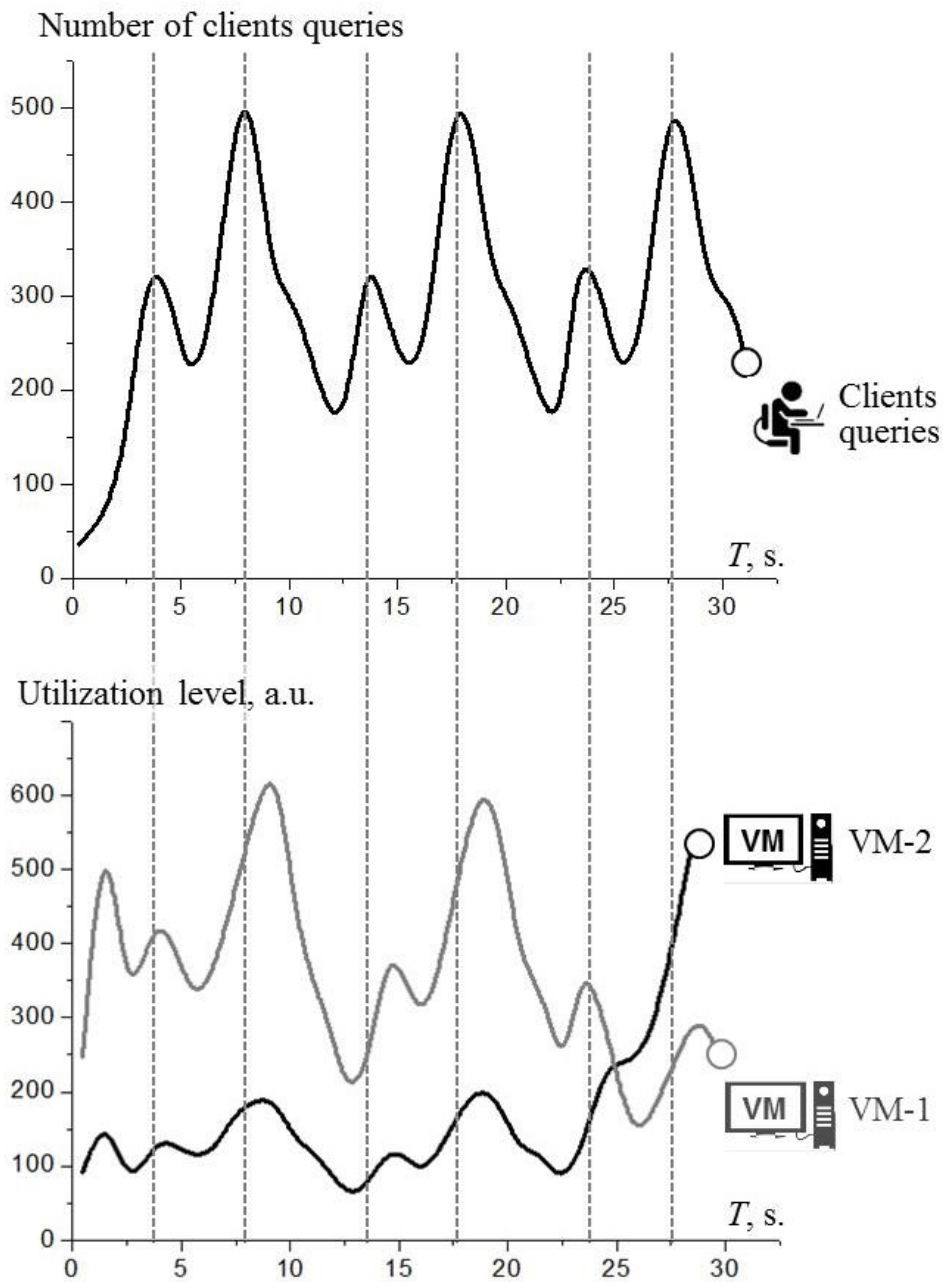


Fig. 3. CPU utilization level of two VMs with respect the number of clients queries

PROPOSED SERVER CONSOLIDATION METHOD

As it was mentioned before clients queries are distributed between multiple VMs and of every cluster and workloads of VM within a same cluster are highly correlated in comparison of correlation of different clusters VMs. In Figures 4-6 is shown intra-cluster correlation of 2 VMs. It can be seen that VMs resources utilization are strongly synchro-

nized. Proposed method includes analysis of VMs pervasive correlation within a cluster and among clusters. The Figures demonstrates the effectiveness of the correlation-aware VM maintaining of 2 servers which possess 2x8 cores. Servers virtualization produces 4 VMs: VM-1, VM-2, VM-3, and VM-4 where VM-1 is co-located with VM-2 and VM-3 is co-located with VM-4 (Figure 4).

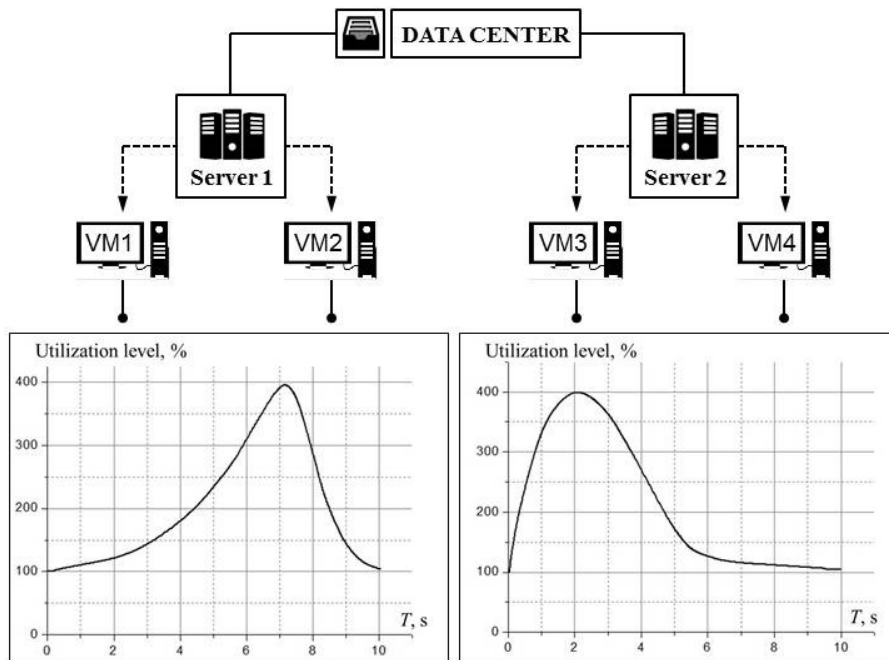


Fig. 4. Data center servers' virtualization procedure simulation

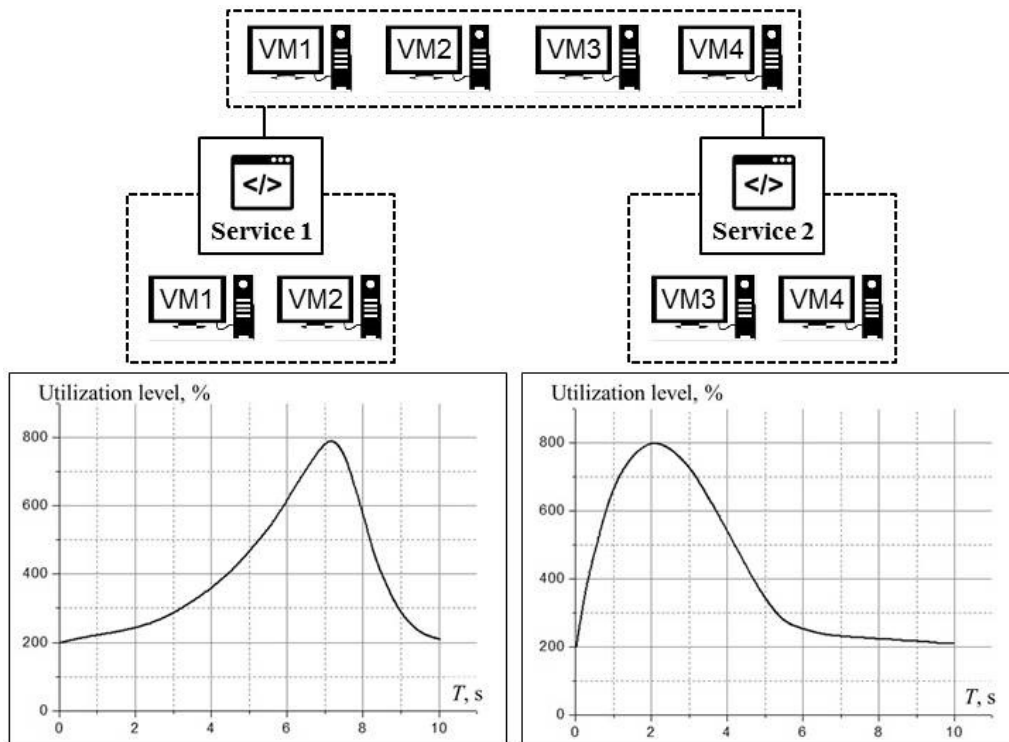


Fig. 5. Drawbacks of VMs allocations without considering correlation value

VMs have the same tail distribution of CPU computational resource utilization and co-located ones are highly correlated. If one will not take into account the correlation (Figure 5), services 1 will allocate sets of VM-1 and VM-2,

while service 2 will allocate sets of VM-3 and VM-4. In this case, extremum value of CPU utilization will attain 8x100% of core of each server (active state of all cores). In other hand, if one will pair services [VM-1; VM-3] and [VM-2; VM-

4], extremum value of CPU utilization for each server cores may be lowered down to 6x100% (Figure 6), which allow

to lower v/f level without services performance degradation.

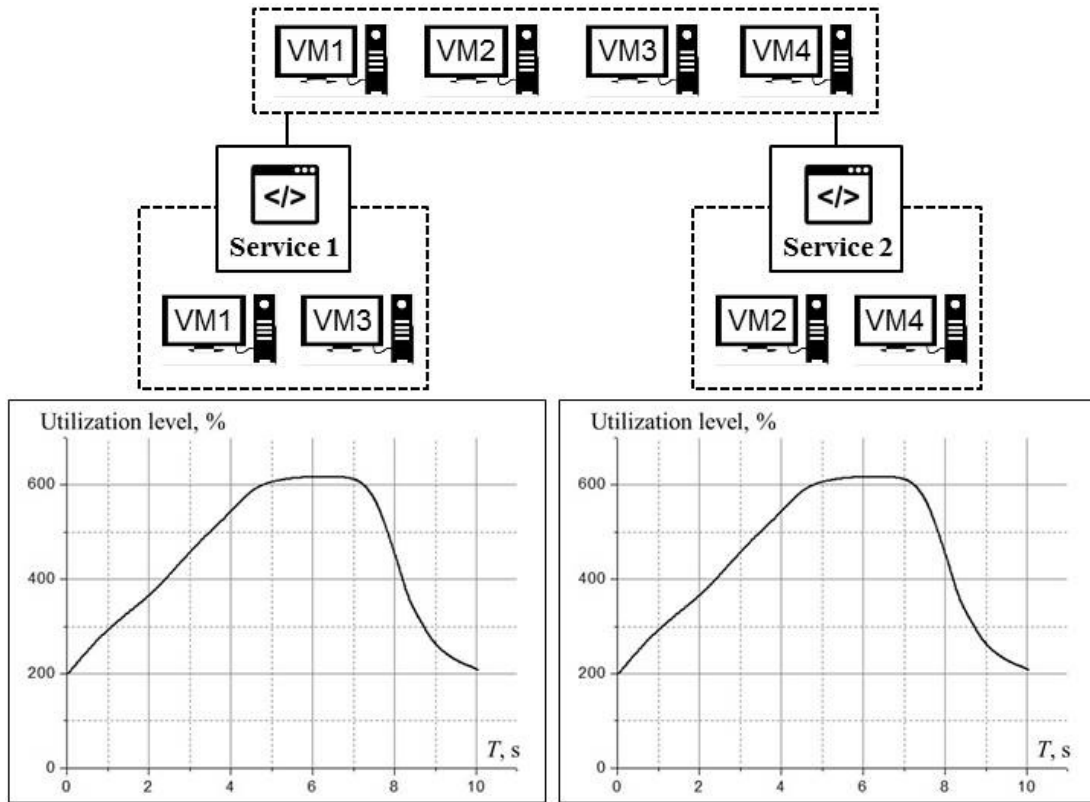


Fig. 6. Correlation-aware VMs allocations procedure

To develop an efficient mathematical model it is proposed to use Pearson's correlation [13] to quantify the correlation coefficient of used data center VMs CPU utilization. It can be calculated as the ratio of covariance of the two variables to the product of their standard deviations. However, Pearson's correlation could be inefficient for the task because this value refers to correlation throughout the corresponding time interval while only correlation at peak or threshold VMs utilization is required. Thereby, it was important to estimate proper measure to quantify the correlation coefficient between VMs that is able to overcome the inefficiency of the conventional correlation metric:

$$C_{i,j} = \left(1 - \frac{U_i^{CPU} + U_j^{CPU}}{U_{i,j}^{CPU}}\right) \cdot 100\%, \quad (1)$$

where $C_{i,j}$ is correlation measure of VM_i and VM_j , U_i^{CPU} is CPU utilization level of VM_i , of VM_j is CPU

utilization level of VM_j and $U_{i,j}^{CPU}$ is aggregated actual peak utilization of co-located VM_i and VM_j . $C_{i,j} = 100\%$ refers to complete correlation, while $C_{i,j} = 0\%$ refers to no correlation.

It is important to note that values of each recorded period of utilization have to be updated. Correlation coefficients between all VMs have to be modeled by $C[i, j]$ matrix where each element corresponds to the $C_{i,j}$ measuring function. This model will allow storing all samples and evenly distributing computational utilization as well as correlation between the events in the bounds of certain time period.

CONCLUSIONS

It was shown that low performance of data center infrastructure work refers to disproportion of servers' utilization. Overconsumption problem could be solved by minimization of the active servers' number. In order to provide



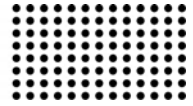
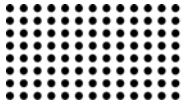
server consolidation implementation it is necessary to maintain acceptable performance level of the servers room infrastructure work. Server consolidation may cause performance degradation due to the conflict of using shared resources by virtual machines. Basic scheme of correlation-aware virtualization includes: tracing data center servers' physical resources utilization level, transforming of utilization traces into binary sequence up to the utilization threshold value, clustering of virtual machines up to the binary sequence and virtual machines allocation at physical servers. Power management procedure which consists from

user-interactive and fast changing service, maintaining of the minimal performance degradation caused by physical resources sharing conflict, high correlation level of virtual machines was developed. Pearson's correlation coefficient was proved to be optimal instrument of the correlation of used data center virtual machines physical resources utilization quantifying. Developed model model allows storing all samples and evenly distributing computational utilization as well as correlation between the events in the bounds of certain time period.

REFERENCES:

1. Harris, M., & Geng, H. (2015). Data center infrastructure management. *Data center handbook, 1st edn.* Wiley, Hoboken, NJ, 601-618.
2. Barroso, L. A., Clidaras, J., & Hölzle, U. (2013). The datacenter as a computer: An introduction to the design of warehouse-scale machines. *Synthesis lectures on computer architecture*, 8(3), 1-154.
3. Ferdman, M., Adileh, A., Kocberber, O., Volos, S., Alisafae, M., Jevdjic, D., ... & Falsafi, B. (2012, March). Clearing the clouds: a study of emerging scale-out workloads on modern hardware. In *ACM SIGPLAN Notices* (Vol. 47, No. 4, pp. 37-48). ACM.
4. Harris, M. (2014). Data Center Infrastructure Management. *Data Center Handbook*, 601-618. doi:10.1002/9781118937563.ch33.
5. Meisner, D., Sadler, C. M., Barroso, L. A., Weber, W. D., & Wenisch, T. F. (2011, June). Power management of online data-intensive services. In *ACM SIGARCH Computer Architecture News* (Vol. 39, No. 3, pp. 319-330). ACM.
6. Bobroff, N., Kochut, A., & Beaty, K. (2007, May). Dynamic placement of virtual machines for managing sla violations. In *Integrated Network Management, 2007. IM'07. 10th IFIP/IEEE International Symposium on* (pp. 119-128). IEEE.
7. Jaramillo, D., Furht, B., & Agarwal, A. (2014). Mobile virtualization technologies. In *Virtualization Techniques for Mobile Systems* (pp. 5-20). Springer, Cham. doi:10.1007/978-3-319-05741-5_2.
8. Tickoo, O., Iyer, R., Illikkal, R., & Newell, D. (2010). Modeling virtual machine performance: challenges and approaches. *ACM SIGMETRICS Performance Evaluation Review*, 37(3), 55-60.
9. Govindan, S., Liu, J., Kansal, A., & Sivasubramaniam, A. (2011, October). Cuanta: quantifying effects of shared on-chip resource interference for consolidated virtual machines. In *Proceedings of the 2nd ACM Symposium on Cloud Computing* (p. 22). ACM.
10. Wang, L., & Lu, Y. (2010, December). Power-efficient workload distribution for virtualized server clusters. In *High Performance Computing (HiPC), 2010 International Conference on* (pp. 1-10). IEEE. doi: 10.1109/hipc.2010.5713178.
11. Meng, X., Isci, C., Kephart, J., Zhang, L., Bouillet, E., & Pendarakis, D. (2010, June). Efficient resource provisioning in compute clouds via vm multiplexing. In *Proceedings of the 7th international conference on Autonomic computing* (pp. 11-20). ACM.
12. Chen, M., Zhang, H., Su, Y. Y., Wang, X., Jiang, G., & Yoshihira, K. (2011, May). Effective VM sizing in virtualized data centers. In *Integrated Network Management (IM), 2011 IFIP/IEEE International Symposium on* (pp. 594-601). IEEE.
13. Halder, K., Bellur, U., & Kulkarni, P. (2012, June). Risk aware provisioning and resource aggregation based consolidation of virtual machines. In *Cloud Computing (CLOUD), 2012 IEEE 5th International Conference on* (pp. 598-605). IEEE.
14. Santos, J. R., Turner, Y. (2011). Virtual Machine Management. *Mastering*, 255-326. doi:10.1002/9781118257432.ch7.
15. Kim, J., Ruggiero, M., Atienza, D., & Lederberger, M. (2013, March). Correlation-aware virtual machine allocation for energy-efficient datacenters. In *Proceedings of the Conference on Design, Automation and Test in Europe* (pp. 1345-1350). EDA Consortium.
16. Khan, S. U. (2015). Handbook on data centers. Place of publication not identified: Springer.

Рецензент: д.т.н., проф. Марасанов В. В.
Херсонський національний технічний університет



РОЗРОБКА СИСТЕМИ УПРАВЛІННЯ ЗНАННЯМИ ОРГАНІЗАЦІЇ НА БАЗІ CMS WORDPRESS

УДК 004.658: 652.3

ДВОРЕЦЬКИЙ М. Л.

ст. викладач кафедри інженерії програмного забезпечення,
Чорноморський національний університет імені Петра Могили, м. Миколаїв, Україна.

Наукові інтереси: СКБД, OLAP, OLTP, web-розробка.

e-mail: m.dvoretzkiy@gmail.com

ДВОРЕЦЬКА С. В.

викладач кафедри інтелектуальних інформаційних систем,
Чорноморський національний університет імені Петра Могили, м. Миколаїв, Україна.

Наукові інтереси: інтелектуальні інформаційні системи та CASE-засоби їх створення,
системи автоматизованого проектування, бази даних та бази знань.

e-mail: svetag603@gmail.com

ДАВИДЕНКО Євген Олександрович

кандидат технічних наук, доцент кафедри інженерії програмного забезпечення,
Чорноморський національний університет імені Петра Могили, м. Миколаїв, Україна.

Наукові інтереси: програмні продукти та технології Microsoft, бази даних та бази знань,
системи автоматизованого проектування, web-технології, системи підтримки прийняття рішень, системний аналіз.

e-mail: genik.davydenko@gmail.com

ПОСТАНОВКА ПРОБЛЕМИ

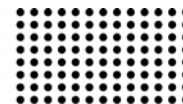
Знання визначаються як інтелект, що використовується у роботі [1]. Знання, що набуті фактичним досвідом, продуктивні тільки у тому випадку, якщо вони і надалі використовуються при виконанні робіт або інтегруються у процес виконання інших робіт. Це необхідна інформація, що використовується згідно певних правил, у відповідності із певними процедурами, і обов'язково із урахуванням відношення до цієї інформації особи, яка їх використовує [2].

Під поняттям управління знаннями мають на увазі пошук та розповсюдження досвіду людей або актуальної інформації в середовищі пов'язаних між собою осіб. Управління знаннями (Knowledge Management – KM) – процес створення умов для виявлення, збереження та ефективного використання інформації та знань у спільноті [3]. Це стратегія, що направлена на вчасне надання необхідних знань тим членам спільноти, яким ці знання

необхідні для того, щоб підвищити ефективність діяльності цієї спільноти [4]. Ішне важливе поняття - обмін знаннями (Knowledge Sharing) – процес постійної циркуляції знань у спільноті, що здійснюється різними способами, по відповідним процедурам згідно певних правил за допомогою застосування організаційних методів із використанням технологічних рішень [5].

Ключем до управління знаннями є доставка в потрібний час необхідних знань тим особам, групам чи організаціям, які їх потребують. Метою управління знаннями є допомога виконавцям робіт краще працювати разом, використовуючи та керуючи інформацією, об'єми якої безперервно збільшуються [6]. Результатом успішно працюючої системи має стати організація, що має знання, самонавчається та самовдосконалюється.

За останні роки в багатьох компаніях створювались технічні рішення «інформаційних порталів» або інших сховищ, що використовувались із словом «знання».



Вони хаотично наповнювались інформацією і формалізованими знаннями (звітами, листами та ін.). Проблема пов'язана з тим, що наявність великої кількості ресурсів не завжди корисна для компанії. В реальності вона свідчить тільки про інформаційну перевантаженість співробітників компанії. Нестача структури, а не кількість інформації є причиною зростаючої неможливості обробити інформацію. Управління знаннями відбувається для того, щоб ними скористатись, а не заради процесу як такого.

Для ефективного видобутку знань необхідно знайти мотивацію до обміну знаннями і професійним досвідом. Наприклад, важливо, якщо до досвіду прислуховуються, визнають авторитет в певній області, або пропозиція стала основою для нової стратегії компанії на ринку [7]. До того ж, співробітники не будуть постійно інвестувати в КМ свій час, якщо це не відображається позитивно на їх професійній діяльності. Такі витрати мають бути компенсовані деякою матеріальною винагородою.

Багато хто вважає, що у процесах КМ мають приймати участь лише найбільш компетентні співробітники (експерти) компанії і КМ-професіонали. Однак, якщо співробітники не приймають участь у процесах КМ, вони відчують, що їх знання залишаються непотрібними [8]. Часто трапляється, що саме ці співробітники полишають компанію.

Отже для будь-якої організації необхідна інтелектуальна та проста у використанні система управління запасами знань, що має механізми ефективного пошуку та набуття нових знань. Слід зазначити, що при розробці системи управління знаннями має бути врахована мотивація співробітників та потенційна можливість будь якого з співробітників приймати участь в наповненні системи знаннями та удосконаленні уже наявних у системі.

АНАЛІЗ ОСТАННІХ ДОСЛІДЖЕНЬ І ПУБЛІКАЦІЙ

Серед існуючих реалізацій систем управління знаннями можна виділити пакет Logik компанії Coreedge [9], пакет Discovery Server фірми Lotus [10], пакет Smartdiscovery компанії Inxight [11] та Livelink компанії Open Text [12]. Підсистеми управління знаннями стають невід'ємною частиною великих корпоративних систем. Так, відома методологія управління бізнес-процесами

ARIS і створена на її основі корпоративна інформаційна система SAP\R3 [13] має у своєму складі підсистему управління знаннями. Розуміння бізнес-процесів є однієї з важливих передумов цілеспрямованого управління знаннями.

Проаналізувавши еволюцію архітектури програмних застосунків [14] можемо звернути увагу, що на зміну класичній трьох ланковій архітектурі приходять web-застосунки, що орієнтовані передусім на використання web-браузера із підтримкою JavaScript та AJAX на клієнтському пристрої. Це дає перевагу, у порівнянні із іншими архітектурами у повній незалежності від платформи та операційної системи клієнтського пристрою. Використання ж Local storage може зробити застосунок менш залежним від постійної наявності он-лайн web-серверу та серверу застосунків.

Поняття веб-застосунку тісно пов'язане із каркасом веб-застосунку, або веб-фреймворком (англ. Web framework). Це програмний каркас, призначений для створення веб-застосунків, служб або ресурсів. Він спрощує розробку, частково за рахунок автоматизації, і позбавляє від необхідності написання рутинного коду. Більшість каркасів спрощують доступ до баз даних а також зменшують дублювання коду [15]. Існують спеціалізовані типи каркасів веб-застосунків, що орієнтовані на створення систем управління контентом.

Система управління вмістом (контентом) (Content management system, CMS) [16] - інформаційна система або комп'ютерна програма, яка використовується для забезпечення і організації спільного процесу створення, редагування і управління вмістом, інакше – контентом. В системі управління вмістом можуть перебувати найрізноманітніші дані: документи, фільми, фотографії, номери телефонів, наукові дані і так далі. Отже система може бути використана для зберігання, управління, перегляду і публікації знань організації.

CMS є одним з ключових компонентів практично будь-якого сайту. Розглянемо три найбільш популярні на даний момент open source системи [17,18]. Joomla, Drupal та WordPress є безкоштовним програмним забезпеченням з відкритим вихідним кодом. Всі три написані на PHP, використовують MySQL в якості СУБД, використовують теми і шаблони для візуального оформлення сайтів і підключають плагіни, модулі для розширення функціоналу. Як програмне забезпечення з

відкритим вихідним кодом, всі вони є проектами, що розвиваються і підтримуються спільнотою.

Але в ряді аспектів ці системи відрізняються. У них різна концепція щодо того, що включати в ядро платформи, як обробляти модулі та шаблони, як забезпечувати безпеку і т.д. Проаналізувавши ряд публікацій [17-19], можна зробити висновок, що Drupal та Joomla за замовчуванням поставляються з більшою кількістю вбудованих функцій, ніж WordPress. Проте, WordPress переважає за рахунок простоти у використанні, великою кількістю плагінів і тем, а також більш розвиненим спільнотою.

Отже, можна зробити висновок, що розробка системи управління знаннями організації у вигляді web-застосунку є актуальною задачею, для реалізації якої авторами статті вважається за доцільне обрання cms wordpress.

ПОСТАНОВКА ЗАВДАННЯ

Метою дослідження є систематизація накопичених співробітниками організації знань за рахунок створення та впровадження системи управління знаннями у вигляді крос-платформного веб-застосунку на базі CMS Wordpress. Вимоги, що висуваються до системи управління знаннями у більшості збігаються із загальними вимогами до інформаційних систем із деякими доповненнями, а саме: можливість вводу та збереження інформації у довільному форматі; віднесення даних до категорій та вказання ключових; пошук та вивід необхідної для вирішення задачі інформації по категоріях та ключових словах; управління користувачами із контролем прав доступу до операцій та матеріалів на рівні користувацьких ролей; можливість розміщення відгуків на матеріал для оцінки його якості; ведення статистики користування матеріалом для оцінки ступеня його корисності.

Для досягнення загальної мети потребують вирішення наступні задачі: інсталяція та налаштування веб-серверу із підтримкою PHP та серверу баз даних MySQL; встановлення та налаштування CMS Wordpress; проведення аналізу можливостей платформи та виділення переліку нереалізованих функцій; обрання та доробка під власні потреби теми для сайту; підключення до системи необхідних плагінів, для реалізації відсутнього за замовченням у системі функціоналу (користувацькі

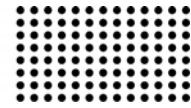
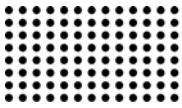
ролі, контроль доступу до матеріалів, історія перегляду та ін.); реалізація частини функціоналу шляхом розробки власних плагінів, що мають відношення до аналізу статистики користувацької активності для підвищення ефективності пошукових механізмів.

ВИКЛАД ОСНОВНОГО МАТЕРІАЛУ

WordPress добре відомий своєю легкою установкою. У більшості ситуацій установка WordPress - дуже простий процес і займає менше ніж п'ять хвилин від початку до кінця. Однак для того, щоб установка та запуск WordPress став можливим, є декілька вимог, що мають бути попередньо виконані. На момент виконання дослідження до них входять наявність веб-сервера із підтримкою HTTP (та HTTPS) та скриптової мови програмування PHP (бажано 7+ версія) та системи управління базами даних MySQL версії 5.6+ (або MariaDB 10+).

Пропонується дуже багато різноманітних готових зборок під ОС Windows, Linux та MacOS (WAMP, LAMP та XAMP відповідно), в які входять всі необхідні для функціонування WP компоненти. Для середовища Windows можна згадати такі пакети, як OpenServer, XAMPP, WampServer та багато інших, використання яких значно спрощує етап установки та налаштування веб-серверу та серверу БД. Однак всі подібні реалізації орієнтовані передусім на підтримку середовища розробника ПЗ (developer environment), а не середовища використання програмного продукту (production environment). У більшості випадків для production environment досвідчені веб-розробники пропонують встановлювати окремо Apache, PHP, MySQL та виконувати відповідні налаштування. Особливості установки та налаштування веб-сервера, сервера БД та самого CMS WordPress не є темою даного дослідження, тому можуть бути опущені.

Після установки та налаштування Apache, PHP, MySQL та WordPress маємо у своєму розпорядженні інформаційну систему, що працює на базі клієнт-серверної СУБД та має у своєму розпорядженні механізми щодо вводу, збереження, представлення даних та підтримки їх цілісності. Крім того, за замовченням WordPress уже підтримує прив'язку матеріалів по категоріям та ключовим словам, та публікацію матеріалів у текстовому форматі із можливістю включення до них у разі необхідності медіа файлів довільного формату. Також зареєстровані користувачі та гості ресурсу мають



можливість залишати коментарі до опублікованих матеріалів. Отже, отримана ІС може бути використана у якості системи управління знаннями організації. Знання щодо вирішення поточних задач представляються у вигляді публікацій (постів). Структурувати дані пропонується за рахунок використання механізму категорій та тегів (що відіграватимуть роль ключових слів).

Для досягнення мети в повному обсязі залишаються невирішеними наступні задачі. По перше, стандартний інтерфейс (тема за замовченням) має бути змінений відповідно вимог до системи управління знаннями (автентифікація, пошук, перегляд, коментар, оцінка, створення/редагування матеріалу). По друге, необхідна додаткова підсистема контролю прав доступу користувачів, передусім на рівні матеріалів, оскільки стандартного розбиття на ролі administrator, creator, editor, author, contributor, subscriber вважається недостатньо. Крім того, можливо впровадження деяких додаткових зручностей, наприклад історія перегляду останніх матеріалів, що може бути корисною для користувачів, які багаторазово використовують обмежений набір публікацій.

Перш за все система вимагає виконання деяких налаштувань інтерфейсу. Виконані вони можуть бути шляхом зміни теми за замовченням. Тема може бути завантажена або через WordPress Dashboard, або у вигляді файлових архівів з офіційного сайту розробників із подальшим їх розгортанням до директорії `wp-content\themes` нашого проекту. Насправді, тема не має бути прив'язаною до якогось певного плагіну, отже вимоги до неї є суто естетичні. На етапі розробки була обрана тема «bani» (офіційний сайт - <http://bani.themesease.com/>), що є однією із класичних тем блогу, не перенасичена додатковими ефектами, що дозволяє користувачам зосередитись на вмісті та медіа контенті. Тема легка (style.css близько 40 кб) та побудована на популярних бібліотеках jquery та bootstrap, що робить можливим підлаштування інтерфейсу під вимоги пристрою, з якого відбувається робота із системою. Простота та універсальність робить можливим подальше використання будь яких необхідних плагінів, та у разі необхідності, спрощує процес розробки власних.

Стандартна система контролю прав доступу в більшості випадків цілком достатньо для типового блогу або сайту-візитівки. Але у випадку системи управління

знаннями доступ має бути обмежено не лише на рівні операцій, а і на рівні матеріалів та публікацій. Для вирішення даної задачі необхідно використати додатковий плагін. Є велика кількість реалізацій, серед яких можна згадати User Access Manager, Restrict User Access – Membership Plugin with Force, Advanced Access Manager та багато інших. Головний недолік вказаних продуктів – велика кількість налаштувань та відповідно складність адміністрування. Так, наприклад, User Access Manager дає весь перелік необхідних функцій, але вимагає детальних налаштувань доступу по кожному матеріалу (публікації). Враховуючи вірогідність великої кількості користувачів та ролей, даних процес стає досить дорогим з точки зору використаного людено-часу. Слід зазначити, що існують платні версії, які дозволяють вирішити вищенаведену проблему через використання замовчень, але це збільшує загальну вартість продукту, що не завжди припустимо для багатьох організацій.

Після тестування вищенаведених та деяких інших продуктів, було прийнято рішення зупинитись на плагіні «groups». Groups створено як ефективне, потужне та гнучке рішення для групового орієнтованого членства та контролю доступу до контенту. Це забезпечує групове керування членством користувачів та можливість контролю доступу до вмісту на рівні користувацьких груп. Groups є невеликим за обсягом та невимогливим до ресурсів, пропонує простий користувацький інтерфейс, та виступає в якості фреймворка, що інтегрує стандартні можливості WordPress та спеціальні можливості застосунків.

Серед переваг додатково можна відмітити підтримку ієрархії користувацьких груп, із можливістю наслідування та обмеження доступу до постів та сторінок, включаючи підтримку custom post types. Особливої уваги заслуговує механізм коротких кодів (shortcodes), що дозволяє обмежувати доступ до фрагментів даних одного матеріалу. Основною перевагою, що зіграла вирішальну роль при виборі підсистеми контролю прав доступу є простота адміністрування. Якщо не виконано жодних дій, матеріал знаходиться в загальному доступі. У разі необхідності його обмеження, необхідно лише додати групу, якій дозволено перегляд цих даних. Враховуючи наслідування групами доступу до матеріалів по ієрархії, навіть наявність великої кількості користу-

вацьких груп при правильній їх ієрархічній організації не робить процес адміністрування дуже дорогим з точки зору людино-часових втрат. Так, наприклад, на даний момент маємо ієрархію з 3-х користувачьких ролей – стандартна роль «Registered» та ще дві: «Readers» та «IT Readers». Кожна з наступних у нащадком попередньої в ієрархії ролей. Тобто матеріали, по яких обмежено доступ через додавання права «read» для групи «Readers», то щоб матеріал було відображено, має бути виконано вхід до системи, та користувач має входити до групи «Readers» або «IT Readers». Підключення плагіну, так само як і теми, може бути виконано двома шляхами: або через WordPress Dashboard або

завантаживши пакет з офіційного сайту та розвернувши їх до директорії wp-content\plugins.

Після інтеграції у систему плагіну groups постає необхідність у додаванні до інтерфейсу логін-форми. Враховуючи особливості інтерфейсу обраної теми bani, вбачається за доцільне розміщення форми у верхній частині бічної панелі. За вивід бічної панелі у WordPress відповідає шаблон sidebar.php, що знаходиться в корні папки wp-content\themes\bani. Також для авторизованих користувачів додамо посилання, за яким можна опублікувати новий матеріал. Фрагмент коду, до виконує вищенаведені операції, наводиться на рис.1.

```

15 <aside id="secondary" class="widget-area col-lg-3 bani-sidebar" role="complementary">
16 <section class = "widget bani-widget card widget_search">
17 <div class="card-block loginform">
18 <?php
19     if (!is_user_logged_in()) {
20         wp_login_form();
21     } else {
22         $current_user = wp_get_current_user();
23         echo "<a href=" . wp_logout_url(home_url()) . ">Logout (" . $current_user->user_login . ")</a>";
24     }
25 >?>
26 </div>
27 <?php
28     if (is_user_logged_in() && current_user_can( 'capability: edit_posts' )) {
29         echo "<div class='card-block loginform'><a href=" . get_site_url() . "/wp-admin/post-new.php'>Додати нову статтю</a></div>";
30     }
31 >?>
32 </section>

```

Рис.1. Зміни шаблону sidebar.php

Крім того, в wp-content\themes\bani створимо login-styles.css, в якому визначимо особливості відображення вищенаведених секцій. І, нарешті, у functions.php підключимо нові стилі і відключимо виведення панелі ін-

струментів WordPress у верхній частині екрану, якщо користувач, який виконав вхід до системи не є членом групи адміністраторів (рис.2).

```

440 //added to insert some our css styles
441 add_action( 'wp_enqueue_scripts', 'my_child_theme_scripts' );
442 function my_child_theme_scripts() {
443     wp_enqueue_style( handle: 'parent-theme-css', src: get_stylesheet_directory_uri() . "/login/login-styles.css" );
444 }
445
446 //added to remove admin menu from site
447 add_action('after_setup_theme', 'remove_admin_bar');
448 function remove_admin_bar() {
449     if (!current_user_can( 'capability: administrator' ) && !is_admin()) {
450         show_admin_bar( show: false);
451     }
452 }

```

Рис.2. Зміни у functions.php

Для додаткової зручності користувачів до системи також включений модуль, що дозволяє відображати на бічній панелі у вигляді віджиту список матеріалів, що останнім часом переглядав користувач, який виконав вхід до системи.

Починаючи використання веб-застосунку дуже важливо мати можливість проаналізувати отриманий веб-сайт. Потрібно з'ясувати, хто є справжніми користувачами проекту і що саме, коли і в якій послідовності вони переглядають. Це питання особливо актуальне в контексті систем управління знаннями, оскільки дозволяє

об'єктивно оцінити корисність та актуальність матеріалів (знань), виявити приховані зв'язки між матеріалами, та організувати зворотній зв'язок із користувачами системи. Це в свою чергу дозволить представити знання кінцевому користувачу у більш структурованому вигляді, відсіяти неактуальні дані та виділити найбільш корисну інформацію. Даний підхід дозволить скоротити час пошуку необхідних інструкцій та пояснень при виконанні поточних задач.

На жаль, WordPress не має вбудованої функції статистики для моніторингу трафіку та відвідувачів, але, звичайно, є кілька плагінів, які можна використати для моніторингу обсягів даних та відстеження користувачів і відвідувачів. Далі проаналізуємо базові можливості найбільш популярних реалізацій.

Загальновідомий JetPack від WordPress.com по праву вважається одним із найкращих доповнень під WordPress та дає їм змогу дізнатись про щоденних користувачів, відвідувачів, налаштування, відвідувачів мобільних пристроїв, інструменти, трафік тощо. Крім того, до складу JetPack входить близько 25 модулів для керування майже всіма основними функціями вашого веб-застосунку. Серед них інтеграція до соціальних мереж, оптимізація для пошукових машин (google, twitter та ін), спам фільтри, управління темами, безпека сайту та багато іншого. Отже плагін є дуже корисним, але у випадку відсутності необхідності у іншому функціоналі, перевантаженим великою кількістю додаткових функцій.

Використовуючи плагін Google Analyticator, за допомогою фрагменту коду JavaScript можливо підключити до сайту блок Google Analytics. У цьому плагіні доступні різні віджети, які відображають дані на сторінці адміністратора. Також доступні деякі додаткові можливості, а саме: відстеження швидкості відгуку сайту, забезпечення перекладу на локальну мову, відслідковування завантажень та ін. Основний недолік – необхідність включення до проекту сторонніх пакетів у вигляді Google Analytics та подальша залежність від нього. Дана особливість також робить неможливою використання проекту у локальній мережі без доступу до мережі інтернет.

Останнім по порядку, але не по значенню, розглянемо пакет WP Statistics. Цей плагін використовує підхід щодо збирання, генерації, та представлення статистики веб-застосунку та надає наступні можливості: виділення сторінок, які мають найбільшу кількість відвідувачів; передача статистики за допомогою електронної пошти; виведення он-лайн користувачів. Також серед переваг можна відзначити наявність актуальної бази даних GeoIP, що дозволяє прив'язати користувачів ресурсу до мапи при візуалізації статистичних даних. Статистичні дані на рівні користувачького інтерфейсу представлені як у табличному вигляді, так і у вигляді графіків. На рис.3. наведена секція pages, що відображає найпопулярніші за відвідуваністю матеріали (знання). Даний інформація використовується для подальшого виявлення корисності матеріалів для кінцевих користувачів системи.

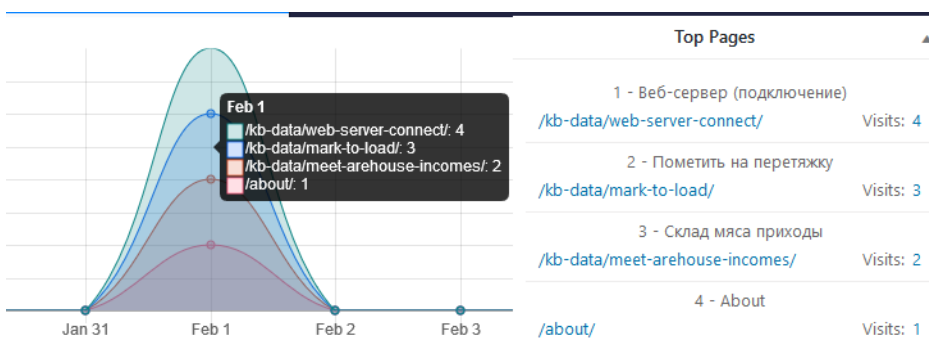


Рис. 3. Секція Pages WP Statistics

Основна перевага, що відіграла вирішальну роль на користь вибору саме пакету WP Statistics – відносна прозорість представлення даних на рівні БД MySQL, що

дозволяє у разі необхідності використовувати накопичену статистику для автоматичної корекції структури матеріалів або виявлення ланцюжків взаємопов'язаних

даних. Так, наприклад, на рис.4. наведено фрагмент використання окремих ресурсів системи. даних таблиці `kb_statistics_pages`, що описує історію

		uri	date	count	id
<input type="checkbox"/>		/about/	2018-01-31	1	2
<input type="checkbox"/>		/kb-data/mark-to-load/	2018-01-31	3	26
<input type="checkbox"/>		/kb-data/meet-arehouse-incomes/	2018-01-31	2	9
<input type="checkbox"/>		/kb-data/web-server-connect/	2018-01-31	4	33

Рис. 4. Фрагмент даних таблиці `kb_statistics_pages`

ВИСНОВКИ ТА ПЕРСПЕКТИВИ ПОДАЛЬШИХ ДОСЛІДЖЕНЬ

Під час виконання дослідження було наведено перелік вимог щодо ПЗ (Apache, PHP, MySQL), встановлено і налаштовано CMS Wordpress та проведено аналіз можливостей платформи щодо використання у якості системи управління знаннями, який виявив наявність у механізмів для збереження матеріалів, представлення їх в ієрархічному (структурованому) вигляді, можливості використання тегів для представлення ключових слів пошуку, та використання коментарів для визначення якості знань та корекції їх змісту (організації зворотного зв'язку із користувачами знань).

Також визначено перелік функцій системи, що не реалізовані «із коробки», та потребують задіяння додаткових модулів, або модифікації програмного коду системи. Проведено аналіз пропозицій відносно контролю доступу на рівні ролей та матеріалів, та згідно результатів обрано та інтегровано до системи плагін `groups`. Виконано додаткові налаштування для оптимізації роботи системи, що мають відношення до корис-

тувального інтерфейсу та аналізу статистики користувацької активності.

В результаті виконання вищенаведених задач на базі CMS Wordpress знаннями у вигляді крос-платформного веб-застосунку розроблено та впроваджено у відділі інформаційних технологій торгівельної організації систему управління знаннями, що дозволяє систематизувати накопичені співробітниками знання.

Система управління знаннями знаходиться у тестовій експлуатації та триває активне наповнення її матеріалами, але уже на цьому етапі отримала схвальні відгуки користувачів – співробітників-виконавців поточних задач організації. Також, згідно до даних системи обліку виконання задач відділу, що використовується на даному підприємстві, час виконання типових завдань після впровадження системи управління знаннями скоротився в середньому на 10%, але результати дуже різняться залежно від предметної області (програмування, адміністрування, технічне обслуговування та ін.). Отже система позитивно зарекомендувала себе на рівні відділу IT та планується для подальшого використання на рівні організації в цілому.

ЛІТЕРАТУРА:

1. Christee Gabour Atwood. Knowledge Management Basics. American Society for Training and Development, 2009. – 144 s.
2. Aleksandr Gaponenko, Tamara Orlova. Upravlenie znaniyami. Kak prevratit znaniya v kapital. Eksmo, 2008. – 400 s.
3. Menedzhment znaniy. [Elektroniy resurs]. Rezhim dostupu: URL: https://ru.wikipedia.org/wiki/Menedzhment_znaniy. – Zagol. z ekranu.
4. Dzhanelto Karen, Uiler Enn. Upravlenie znaniyami. Rukovodstvo po vnedreniyu korporativnoy sistemiy upravleniya znaniyamiyu. Per. s angl. E. M. Pesterevoy. – M.: Dobraya kniga, 2005. – 192 s.
5. Upravlenie znaniyami: printsipy, metody, effektivnost. [Elektroniy resurs]. Rezhim dostupu: URL: <http://magazine.hrm.ru/db/hrm/AC09CFE8A92DD7E9C3257718004F1C82/print.html>. – Zagol. z ekranu.
6. Kevin C. Desouza, Scott Paquette. Knowledge Management: An Introduction. Neal-Schuman Publishers, 2011. – 351 s.
7. Mariya Marinicheva. Upravlenie znaniyami na 100%. Putevoditel dlya praktikov. Alpina Biznes Buks, 2008. – 320 s.
8. Ermoshina N.P. Upravlenie znaniyami kak element innovatsionnoy deyatelnosti proektnoy organizatsii. [Elektroniy resurs]. Rezhim dostupu: URL: <http://www.sworld.com.ua/index.php/uk/economy-311/innovation-economy-311/7692-knowledge-management-as-an-element-of-design-innovation>. – Zagol. z ekranu.



9. CoreEdge Software launches version 2.0 of powerful knowledge preview tool [Elektroniy resurs]. Rezhim dostupu: URL: <http://www.prnewswire.com/news-releases/coreedge-software-launches-version-20-of-powerful-knowledge-preview-tool-76286797.html>. – Zagol. z ekranu.
10. Inside the Lotus Discovery Server [Elektroniy resurs]. Rezhim dostupu: URL: <http://www.redbooks.ibm.com/redbooks/pdfs/sg246252.pdf>. – Zagol. z ekranu.
11. Inxight SmartDiscovery Sets New Standard for Enterprise Information Retrieval [Elektroniy resurs]. Rezhim dostupu: URL: <http://www.prnewswire.com/news-releases/inxight-smartdiscovery-sets-new-standard-for-enterprise-information-retrieval-75972852.html>. – Zagol. z ekranu.
12. Livelink and Livelink Server [Elektroniy resurs]. Rezhim dostupu: URL: <http://www.opentext.com/what-we-do/products/opentext-product-offerings-catalog/rebranded-products/livelink-is-now-part-of-the-opentext-ecm-suite>. – Zagol. z ekranu.
13. Sheer Avgust-Vilgelm. ARIS- modelirovanie biznes-protsesov. Vilyams, 2009. 224 s.
14. Markus Aurelius. Evolyutsiya prilozheniy ili kuda myi idem. [Elektroniy resurs]. Rezhim dostupu: URL: <https://habrahabr.ru/post/326016/>. – Zagol. z ekranu.
15. Karkas veb-zastosunkiv. [Elektroniy resurs]. Rezhim dostupu: URL: https://uk.wikipedia.org/wiki/Karkas_veb-zastosunkiv. – Zahol. z ekranu.
16. Sistema upravleniya soderzhimiyim. [Elektroniy resurs]. Rezhim dostupu: URL: https://ru.wikipedia.org/wiki/Sistema_upravleniya_soderzhimiyim. – Zagol. z ekranu.
17. Kakuyu CMS vybrat dlya sayta. [Elektroniy resurs]. Rezhim dostupu: URL: <https://lifehacker.ru/2014/06/16/kakuyu-cms-vybrat-dlya-sajta-sravnenie-wordpress-joomla-i-drupal/>. – Zagol. z ekranu.
18. WordPress vs Joomla vs Drupal – Chto luchshe? [Elektroniy resurs]. Rezhim dostupu: URL: http://www.internet-technologies.ru/articles/article_2808.html. – Zagol. z ekranu.
19. Brad Williams, David Damstra, Hal Stern. Professional WordPress: Design and Development, 3rd Edition. John Wiley & Sons, 2012. 456 s.

*Рецензент: д.т.н., проф. Фісун М. Т.
Чорноморський національний університет
імені Петра Могили*

ГЕНЕРАЦІЯ СУБ-ОПТИМАЛЬНИХ МАРШРУТІВ БЕЗПІЛОТНОГО ЛІТАЛЬНОГО АПАРАТА З ВИКОРИСТАННЯМ НЕЙРОННОЇ МЕРЕЖІ ХОПФІЛДА

УДК 004.021:004.85

ЖУРАВСЬКА Ірина Миколаївна

кандидат технічних наук, доцент, докторант кафедри комп'ютерної інженерії,
Чорноморський національний університет імені Петра Могили, м. Миколаїв, Україна

Наукові інтереси: архітектура, впровадження та інформаційна безпека гетерогенних комп'ютерних мереж та систем, включаючи SCS, Wireless, PowerLine та інші. Автор однієї книги, понад 70 статей та тез конференцій, 4 патентів.

E-mail: irina.zhuravska@chmnu.edu.ua.

ПОСТАНОВКА ПРОБЛЕМИ

У сучасному світі існують безліч процесів, які є потенційно небезпечними для людини або неможливі для виконання безпосередньо нею. Тому ще у минулому сторіччі було запропоновано використовувати апарати, які не потребують управління ними людиною-оператором безпосередньо. З розвитком комп'ютерних технологій такі апарати мають все більше можливостей і участь оператора у їх роботі стає все меншою. Одними з таких апаратів є безпілотні літальні апарати (БПЛА).

Під час управління БПЛА на оператора покладається відповідальність за прийняття рішення, в якому порядку відвідати задані вузли, що робити, якщо раптом втрачено контроль або БПЛА збито з курсу. Доцільно цю відповідальність деякою мірою покласти на комп'ютер, а також прискорити процес генерації можливих маршрутів БПЛА за рахунок автоматизації такого процесу. Для цього необхідно створити програмний продукт, який після отримання від оператора координат точок, які має відвідати БПЛА, генеруватиме маршрут за якимось з критеріїв. Крім того, у випадку порушення курсу, такий програмний додаток повинен коригуватиме його. Таку задачу можливо привести до відомої задачі комівояжера.

АНАЛІЗ ОСТАННІХ ДОСЛІДЖЕНЬ ТА ПУБЛІКАЦІЙ

Задачу комівояжера достатньо докладно розглядає Левитин А. В. і Мудров В. І. [1, 2].

Як математична проблема задача комівояжера відома з 19-го століття. Історично вона полягає в тому, який маршрут має вибрати комівояжер (торговець-постачальник), щоб об'їхати всі точки призначення, обравши найкоротший шлях і не рази не опиняючись в одному місці двічі [1].

Задача комівояжера – одне з найвідоміших завдань комбінаторної оптимізації, що полягає у пошуку найвигіднішого маршруту, що проходить через вказані точки хоч би по одному разу з подальшим поверненням в початкову точку [2]. В умовах завдання вказуються критерій вигідності маршруту (найкоротший, найдешевший, сукупний критерій і тому подібне) і відповідні матриці відстаней, вартості і тому подібного. Як правило, вказується, що маршрут повинен проходити через кожну точку тільки один раз. Як вже зазначалось, вважатимемо найкращим маршрут найкоротший, а кожен вузол апарат має відвідати лише один раз.

Вперше нейронна мережа Хопфілда була представлена на початку 80-х років минулого сторіччя у працях Джона Хопфілда. Дж. Хопфілд показав, що нейронна мережа із зворотними зв'язками може бути системою, що мінімізує енергію (так звана мережа Хопфілда) [3].

Нейронна мережа Хопфілда – повнозв'язна нейронна мережа з симетричною матрицею зв'язків. В процесі роботи динаміка таких мереж сходиться до одного з положень рівноваги. Ці положення рівноваги

визначаються заздалегідь в процесі навчання, вони є локальними мінімумами функціонала, що називається енергією мережі (у простому випадку – локальними мінімумами негативно певної квадратичної форми на n -мірному кубі) [4].

На рис. 1 зображено архітектуру нейронної мережі Хопфілда [5].

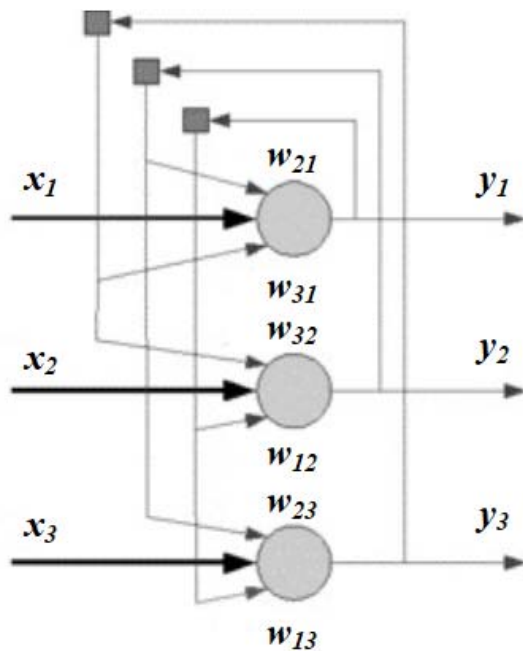


Рис. 1. Архітектура нейронної мережі Хопфілда [5]

Нейронна мережа Хопфілда може використовуватися для вирішення завдання комівояжера, але треба враховувати, що використання нейронної мережі Хопфілда для реалізації задачі комівояжера дуже обмежена через завелику похибку у розрахунках оптимального шляху [6–8].

Таким чином, отримані в результаті вирішення вказаної задачі маршрути БПЛА можна вважати лише суб-оптимальними, тому остаточне рішення щодо вибору з них одного залишається за наземним оператором [9–11]. Практична цінність запровадження алгоритму генерації таких суб-оптимальних маршрутів полягає у тому, що буде полегшена робота оператора, прискорений процес формування маршрутів та зменшений процент людської помилки під час управління БПЛА.

Формулювання цілей статті. Метою статті є розробити та дослідити алгоритм генерації суб-

оптимального маршруту БПЛА на основі використання нейронної мережі Хопфілда.

ВИКЛАД ОСНОВНОГО МАТЕРІАЛУ

Оптимальним маршрутом вважатимемо маршрут, який є найкращим згідно до деякого критерію (довжина, час подолання тощо).

Для вирішення задачі було погоджено, що суб-оптимальним маршрутом вважатиметься такий найкоротший маршрут, за яким БПЛА, облетівши всі вузли, має повернутись у точку вильоту. При цьому, кожен вузол БПЛА має відвідати лише один раз; на шляху БПЛА немає перешкод або жодної іншої умови, при якій можливе порушення курсу.

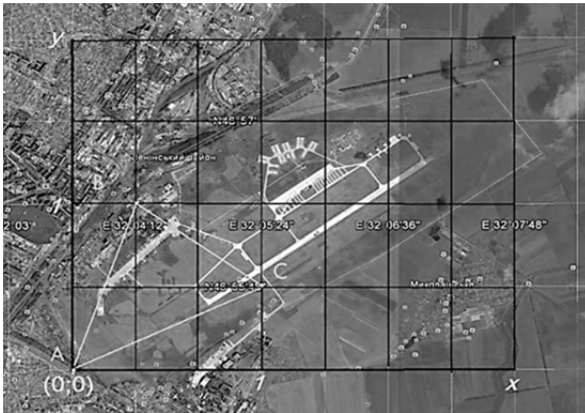
Методом для вирішення задачі було обрано нейронну мережу Хопфілда (НМХ).

Нехай задано місцевість для обстеження її з допомогою БПЛА. На мапу місцевості наноситься координатна сітка і задається точка відліку (рис. 2).

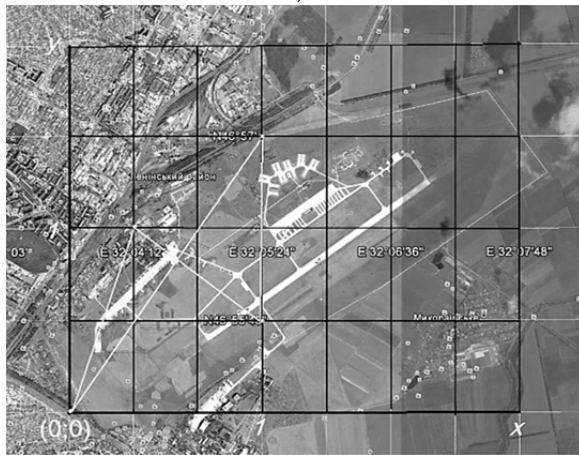


Рис. 2. Карта місцевості з нанесеною системою координат

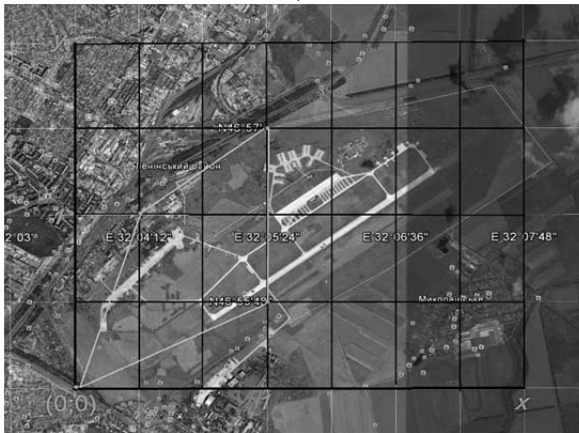
Нехай, оператор хоче, щоб апарат зробив знімки місцевості у вузлах $(0; 0), (0,3; 1), (1; 0,5)$. Точка вильоту має координати $(0; 0)$, як наведено на рис. 3, а. В такому разі існує лише один маршрут і, відповідно, він і є найкоротшим. Проте, якщо додати ще один вузол, наприклад $(1; 1,5)$, то маршрутів буде значно більше (рис. 3, б та рис. 3, в), а точніше $\frac{(n-1)!}{2}$ маршрутів, де n – кількість вузлів у маршруті.



a)



b)



в)

Рис. 3. Варіанти маршрутів БПЛА для трьох (а) і чотирьох (б, в) вузлів

Для архітектури НМХ (див. рис. 1) x_1, x_2, x_3 – входи, w_{ij} , де i – номер нейрона, а j – номер виходу, – ваги, u_1, u_2, u_3 – виходи. При цьому кожен нейрон може приймати два стани:

$$S(t) = \{-1; 1\}. \quad (1)$$

Динаміка стану в часі i -ого нейрона в мережі з N нейронів описується дискретною динамічною системою (2):

$$S_i(t + 1) = \sum_{j=1}^N w_{i,j} * S_j(t), i=1..N, \quad (2)$$

де $w_{i,j}$ – матриця вагових коефіцієнтів.

Слід зазначити, що навчання мережі відбувається за правилом Хеба [5]. Тобто, поправка вагових коефіцієнтів відбуватиметься за формулою (3).

$$J_{i,j} = \frac{1}{N} \cdot \sum_{\mu=1}^M [\varepsilon_{i,\mu}^{in} \cdot \varepsilon_{j,\mu}^{in}], \quad (3)$$

де $\varepsilon_{i,\mu}^{in}$ і $\varepsilon_{j,\mu}^{in}$ i -й і j -й елементи образу (вхідного набору) ε_{μ}^{in} .

Для того, щоб НМХ могла використовуватися для вирішення завдання комівояжера, необхідно забезпечити дотримання деяких вимог:

1. Мережа повинна складатися з $N = n \times n$ нейронів, які будемо розглядати, як квадрат з n рядків і n стовпців.
2. Відповідь мережі повинна містити тільки один активний нейрон в кожному рядку і кожному стовпці.
3. Активний нейрон в першому стовпці задає першу точку маршруту, в другому стовпці – другу точку маршруту, і так далі.

Усім цим умовам задовольняє формула (4) обчислення ваги між нейроном, що відповідає точці x на позиції в i -му маршруті, і нейроном, що відповідає точці y на позиції j :

$$W_{xi,yi} = -A\delta_{x,y} \cdot (1 - \delta_{i,j}) - B\delta_{i,j} \cdot (1 - \delta_{x,y}) - Cd(x,y) \cdot (\delta_{i,j+1} + \delta_{i,j-1}) + D, \quad (4)$$

де A, B, C, D – деякі константи;

$d(x, y)$ – відстань між точками x і y ;

$\delta_{x,y}$ – символ Кронекера, що набуває значення 1, якщо $x = y$ і значення 0 інакше.

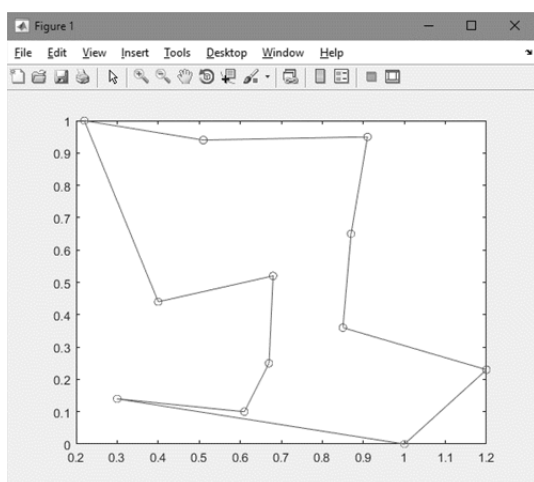
У формулі (4) перший член рівний $(-A)$ для усіх зв'язків в тому ж рядку ($x = y$), окрім зв'язку нейрона з самим собою (при $i = j$). Другий член рівний

(-B) для усіх зв'язків в тому ж стовпці ($i = j$), окрім зв'язку з самим собою ($x = y$). Третій член пропорційний відстані між точками x і y , якщо ці міста сусідні в маршруті ($i = j - 1$ чи $i = j + 1$).

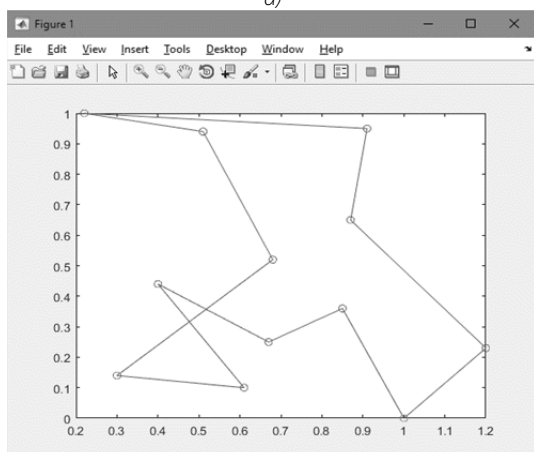
Якщо таку мережу привести у випадковий початковий стан, то можна очікувати, що результуючий стабільний стан дасть нам суб-оптимальний шлях, довжина якого не занадто перевершує оптимальну (сам шлях може значно відрізнятись від оптимального). Відповідно, для практичного застосування мережу слід запуснути кілька разів, і вибрати найкращий шлях.

На основі вищезазначеного алгоритму для дослідження ефективності використання нейронної мережі Хопфілда для вирішення задачі комівояжера було створено додаток, який генерує маршрут на основі введених оператором точок для обстеження. На рис. 4, а – рис. 4, в зображено маршрути, згенеровані програмою в результаті її послідовних викликів на основі одних й тих самих даних.

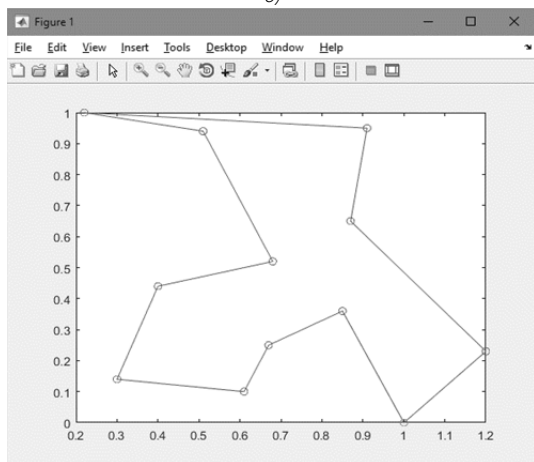
З рис. 4 й табл. 1 можливо побачити, що маршрути, отримані в результаті виконання програми, мають приблизно однакову довжину, проте, вигляд мають абсолютно різний.



а)



б)



в)

Рис. 4. Згенеровані маршрути
Точкою вильоту у всіх випадках є точка (1; 0).

Таблиця 1

Аналіз згенерованих НМХ маршрутів

№№ маршруту	Довжина маршруту, Len (км)	Відхилення довжини маршруту	
		абсолютне, $ \Delta D $ (км)	відносне, $\varepsilon = \frac{\Delta D}{D}$ (%)
1	4,3536	0,0092	2,96
2	4,5278	0,1834	
3	4,1519	0,1925	

Як вже зазначалось, річ у тім, що нейронна мережа Хопфілда може вирішити задачу комівояжера лише приблизно (результат зазвичай трохи довший за ідеальне значення). В результаті маршрути, що генеруються НМХ, завжди приблизно однакові по довжині, але різні за порядком проходження вузлів. Тому оператор – при генерації маршруту для обльоту БПЛА місцевості – має згенерувати декілька маршрутів, а потім обрати найкращий.

ВИСНОВКИ ТА ПЕРСПЕКТИВИ ПОДАЛЬШИХ ДОСЛІДЖЕНЬ

У даному дослідженні було запроваджене використання нейронної мережі Хопфілда для генерації суб-оптимального маршруту БПЛА. Було погоджено, що на політ апарата нічого не впливає, він має облетіти всі задані оператором точки, причому в жодній точці не

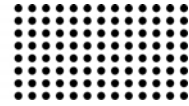
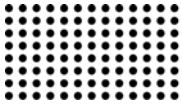
повинен бути більше одного разу, і в кінці польоту апарат має повернутись у точку вильоту. Оптимальним вважається шлях найкоротшої довжини. Показано, що нейронна мережа Хопфілда є доволі приблизним методом вирішення проблеми, оскільки генерує маршрути, що досить схожі на оптимальний за довжиною (незначно його перевищують), але можуть суттєво відрізнятися від нього. В результаті не зменшується повною мірою відповідальність оператора за вибір оптимального

маршруту, оскільки йому все одно доведеться вибрати найоптимальніший серед декількох запропонованих програмою, отриманих в результаті декількох послідовних прогонів з тими самими вхідними даними. Проте, в результаті досліджень доведена можливість суттєво зменшити вірогідність людської помилки, оскільки оператор буде, умовно кажучи, вибирати «кращих з найкращих» маршрутів.

ЛІТЕРАТУРА:

1. Levitin A.V. Metod gruboy sily: Zadacha kommivoyazhora // Algoritmy. Vvedeniye v razrabotku i analiz : glava 3. – М. : Vil'yams, 2006. – 576 s.
2. Mudrov V.I. Zadacha o kommivoyazhore. – М. : Znaniye, 1969. – 62 s.
3. Hopfield J. J. What is a Moment? Transient synchrony as a collective mechanism for spatiotemporal integration / J. J. Hopfield, C. D. Brody // Proceedings of the NAS of the USA. – 2001. – Vol. 98, No. 3. – P. 1282–1287. DOI: 10.1073/pnas.98.3.1282.
4. Uossermen F. Neyrokomp'yuternaya tekhnika: Teoriya i praktika. – М. : Mir, 1992. – 240 s.
5. Zhong C. A continuous hopfield neural network based on dynamic step for the traveling salesman problem / C. Zhong, C. Luo, Z. Chu, W. Gan // Neural Networks (IJCNN'2017) : Proceeding of IEEE International Joint Conference, Anchorage, AK, USA, 14–19 May, 2017. – P. 3318–3323. DOI: 10.1109/IJCNN.2017.7966272.
6. Khaykin S. Neyronnyye seti. Polnyy kurs. – М. : Vil'yams, 2008. – 1113 s.
7. Musiyenko M.P. Simulation the behavior of robot sub-swarm in spatial corridors / M.P. Musiyenko, I.M. Zhuravska, I.V. Kulakovska, A.V. Kulakovska // Electronics and Nanotechnology (ELNANO'2016) : Proceeding of the 2016 IEEE 36th International Conference, Kyiv, Ukraine, April 19–21, 2016 / National Technical University of Ukraine "Kyiv Polytechnic Institute". – P. 382–387. DOI: 10.1109/ELNANO.2016.7493090.
8. Komashinskiy V. I., Smirnov D. A. Neyronnyye seti i ikh primeneniye v sistemakh upravleniya i svyazi. – М. : Goryachaya liniya-Telekom, 2003. – 94 s.
9. Wang N. Hopfield neural network guided evolutionary algorithm for aircraft penetration path planning / N. Wang, L. Wang, X. Go, L. Chen, L. Shen // Advances in neural network research and applications (LISEE, 67). – Berlin; Heidelberg : Springer-Verlag, 2010. – P. 235–243.
10. Intel Plans to Break Drone Light Show Record with Over 1,500 Drones Flown at 50th Anniversary Celebrations [Electronic resource] // Intel Newsroom : [site]. – Publ. April 30, 2018. – Available at : <https://newsroom.intel.com/news/intel-plans-break-drone-light-show-record-over-1500-drones-flown-50th-anniversary-celebrations/>.
11. Musiyenko M.P. Alhorytmy prokladannia marshrutu bezpilotnykh litalnykh aparativ na osnovi zastosuvannia neironnykh merezh Khopfil'da / M.P. Musiyenko, I.M. Zhuravska // Visnyk Cherkaskoho derzh. tekhnol. un-tu : zb. nauk. prats. Seriya : Tekhnichni nauky. – 2016. – № 1. – S. 20–27.

*Рецензент: д.т.н., проф. Мусієнко М. П.
Чорноморський національний університет
імені Петра Могили*



МАТЕМАТИЧНА МОДЕЛЬ АЛГОРИТМУ ВИБОРУ ЗБРОЇ БОТОМ В КОМП'ЮТЕРНІЙ ГРІ ЖАНРУ ШУТЕР

УДК

САЛТАН Борис Андрійович

Чорноморський національний університет ім. Петра Могили, магістр,
м.Миколаїв, Україна,
e-mail: bs.arezius@gmail.com

СОБКО Дмитро Анатолійович

Чорноморський національний університет ім. Петра Могили, магістр,
м.Миколаїв, Україна
e-mail: sobkod23@gmail.com

КУЛАКОВСЬКА Інесса Василівна

Чорноморський національний університет ім. Петра Могили, канд. фіз.-мат. наук,
завідувач кафедри інтелектуальних інформаційних систем Чорноморського національного університету
ім. Петра Могили, м.Миколаїв, Україна
ORCID: 0000-0002-8432-1850,
e-mail: kulaknic@gmail.com

ВСТУП

XXI століття — століття комп'ютерних технологій, які щодня розвиваються і використовуються як для праці, так і для відпочинку. Окрему нішу займають комп'ютерні ігри, тобто комп'ютерні програми або частини комп'ютерних програм, що служать для організації ігрового процесу (геймплея).

Шутер (від англ. shoot - стріляти) - популярний жанр відео ігор, ігровий процес яких заснований на боях в режимі реального часу з використанням, як правило, вогнепальної зброї. Програмування ігрового штучного інтелекту в шутерах зводиться, головним чином, до розробки моделі поведінки ботів. Ботами прийнято називати ворожих до гравця персонажів, які контролюються комп'ютером.

Для того, щоб ігровий процес шутера був цікавим і реалістичним, боти повинні бути запрограмовані таким чином, щоб з одного боку, їх було складно переіграти, але з іншого боку, у людини залишалися шанси на пе-

ремогу. В ідеалі, боти повинні діяти подібно персонажам, контрольованим реальними людьми.

У зв'язку з цим, виникає ряд непростих завдань, таких як реалізація тактики командного ведення бою ботами; моделювання прицілювання, стрільби, переміщення ботів по карті; реалізація алгоритму вибору зброї ботом в залежності від ігрової ситуації. У даній роботі розглядається рішення останньої з перерахованих завдань.

Аналіз останніх досліджень та публікацій Інформаційна база у сфері математичних моделей для комп'ютерних ігор жанру шутер майже відсутня. Аналіз джерел, знайдених в мережі Інтернет([1]—[6]), показав, що данні моделі представлені тільки у формі опису на форумах для гравців.

Основною метою роботи є дослідження математичної моделі алгоритму вибору стратегії ігрової програми, які визначають рівень складності гри і гарантують певний відсоток вигравшів гравцеві, за умови, коли він

визначить вірну стратегію гри для себе з використанням сценарного аналізу, нейронних мереж, їх застосування в алгоритмі вибору зброї ботом в тривимірному шутері від першої особи, тестування реалізованих алгоритмів і виявлення переваг і недоліків кожного підходу.

Задача штучного інтелекту для ігор полягає в імітації поведінки об'єктів реального світу. Для цього слід почати розгляд штучного інтелекту з базових компонентів - від низькорівневих правил і алгоритмів до більш високого рівня, на якому працює тактичний і стратегічний ШІ. При цьому, слід домогтися високої ефективності роботи системи ШІ, оптимізувати її для використання на комп'ютерах. Це означає, що потрібні нові наукові і практичні розробки, які б дозволили більш широко використовувати технологію створення засобів ШІ.

ОСНОВНІ ПІДХОДИ ДО РЕАЛІЗАЦІЇ МОДЕЛІ ПОВЕДІНКИ БОТА

В ході вивчення предметної області, було проаналізовано ряд матеріалів, що стосуються деталей використання сценарного аналізу, штучного інтелекту в відео іграх, в

основному, в шутерах від першої особи. У багатьох сучасних іграх поведінка ботів характеризується високою передбачуваністю. Це пов'язано з тим, що розробники наголошують на ефективності застосовуваних алгоритмів, не замислюючись безпосередньо про модель прийняття рішень ботом. Так, в шутерах Left 4 Dead і Killzone для розрахунку траєкторії переміщення ботів використовується алгоритм A*. При цьому, моделі поведінки ботів засновані на використанні кінцевих автоматів. В результаті, поведінка ботів повністю детермінована, на відміну від поведінки ігрових персонажів, контрольованих реальними людьми [1]. Безумовно, такий підхід до створення ігрового штучного інтелекту істотно знижує інтерес користувачів до проекту.

Що стосується підходів до реалізації вибору зброї в шутерах від першої особи, то як правило, для цієї мети використовуються експертні системи, засновані на правилах.

Нижче представлений приклад псевдокоду алгоритму вибору зброї, заснованого правилах.

<pre> if(enemy.distance < 5) { ATTACK-WITH-KNIFE() } else if(enemy.distance >= 5 AND enemy.distance <= 30) </pre>	<pre> { ATTACK-WITH-SUBMACHINE-GUN() } else { ATTACK-WITH-RIFLE() } </pre>
--	--

Лістинг 1 – Правило вибору зброї

Крім передбачуваності прийнятих ботом рішень, недоліком такого підходу є велика кількість параметрів, які необхідно підбирати (в описаному вище прикладі таким параметром є enemy.distance - відстань між ботом і ворогом). Для того, щоб наблизити поведінку бота до реальності, потрібно враховувати багато факторів, і налаштування експертної системи стає складним завданням. Пропонується використовувати генетичні алгоритми для вирішення цього завдання.

Навчання з підкріпленням - сукупність методів машинного навчання, в ході яких агент навчається, взаємодіючи з деяким навколишнім середовищем. Агент взаємодіє з оточенням (Environment), застосовуючи різні дії (Actions) в залежності від значень певних параметрів (State) і отримуючи нагороду (Reward). Нижче наведена схема, що ілюструє описану взаємодію агента з навколишнім середовищем.

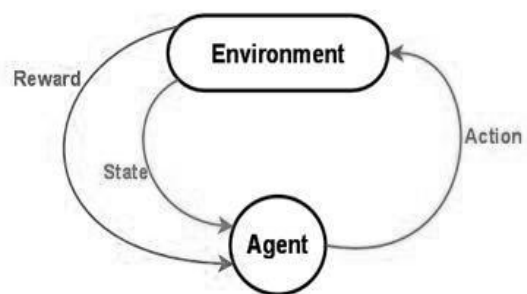
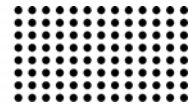
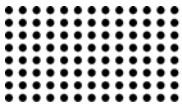


Рис. 1 – Взаємодія агента з навколишнім середовищем

Мета таких методів полягає в навчанні агента вибрати дії, ґрунтуючись на поточному стані навколишнього середовища, таким чином, щоб отримати максимальну нагороду [2]. У термінах поставленого завдання, використання методів навчання з підкріпленням в алгоритмі вибору зброї, дозволяє боту протягом гри накопичувати досвід використання кожної зброї в різних ігрових ситуаціях і діяти відповідно із зібраними



знаннями, постійно модифікуючи тактику вибору зброї з метою збільшення одержуваної нагороди.

Штучна нейронна мережа (ШНМ) - математична модель, побудована за принципом організації мереж нервових клітин живого організму. Нейронні мережі широко застосовуються в різних областях. Зокрема, вони активно використовуються в аналізі даних, наприклад, при вирішенні задач класифікації даних, кластеризації даних, апроксимації неперервних функцій.

Залежно від характеру навчання, ШНМ прийнято розділяти на навчаються з учителем і навчаються без учителя. Також, останнім часом, як правило, окремо виділяють нейронні мережі, які навчаються з підкріпленням. Кожен з перерахованих типів ШНМ призначений для вирішення певного класу задач. Навчання з учителем застосовується, коли заздалегідь відомі вихідні значення мережі при певних вхідних значеннях. Нейронні мережі даного типу підходять для вирішення таких завдань, як класифікація даних. Серед алгоритмів навчання таких мереж найбільш відомий метод зворотного поширення помилки (backpropagation)[3]. Нейронні мережі, які навчаються без вчителя, має сенс використовувати в тих випадках, коли немає ніякої апріорної інформації про те, якими мають бути вихідні значення мережі. Такі нейронні мережі, як правило, застосовуються для вирішення завдань кластеризації даних. У даній роботі була використана нейромережева модель FALCON, яка навчається з підкріпленням. Детальний опис архітектури даної нейронної мережі, її алгоритму навчання та особливостей застосування, буде наведено в наступних публікаціях.

Під час військових дій можна зіштовхнутися із багаточисленними і різноманітними сценаріями, такими як рейди по тилах противника або засідки, лобові атаки і флангові напади, маневрена війна і позиційні бої і т.д. Солдати, опинившись в певній ситуації, безперечно, прагнуть знайти найкращі інструментальні засоби для виконання свого бойового завдання, що дозволяє збільшити ймовірність успіху або просто забезпечити шанси на виживання.

До категорії стратегій високого рівня відносяться навички вибору зброї, які доповнюють навички володіння зброєю.

Вибір зброї вдає із себе задачу, при вирішенні якої в процесі гри виникає багато варіантів. Під час бою найчастіше виявляється успішним один-єдиний якісний спосіб поведінки, незалежно від контексту. А коли доводиться маніпулювати самою зброєю, боту доводиться динамічно враховувати набагато більше факторів.

При вирішенні завдання вибору зброї може використовуватися навчання. При цьому виникають цікаві тенденції в розвитку проекту гри або навіть засобів ШІ. Наприклад, ми можемо визначити, чи є вибір зброї цілком виправданим і, в разі необхідності, модифікувати логіку поведінки гри.

Вихідні дані:

- розроблювані боти повинні бути здатними до стрільби, вони не зобов'язані мати ідеальну поведінку;
- для вибору зброї в грі абсолютно необхідна підтримка декількох видів зброї, як завжди, таку можливість має надавати ігрова машина;
- передбачається, що середовище дозволяє учасникам битви приймати вельми різноманітні розташування, що може призводити до того, що бойова ситуація кожен раз буде складатися по-різному

ВИБІР ЗБРОЇ ІЗ ЗАСТОСУВАННЯМ СЦЕНАРІЇВ

Безумовно, засоби сценарної підтримки не є результатом досліджень виключно в області ШІ, але неможливо заперечувати й те, що ці засоби дуже широко використовуються в додатках ШІ, і особливо в комп'ютерних іграх [5].

Необхідність в застосуванні зручних мов сценаріїв не вичерпується традиційними підходами до створення ігрових засобів ШІ; в міру покращення додатків ШІ сценарна підтримка в них не виключається, а просто починає обслуговувати інші потреби (відмінні від потреб реалізації форм поведінки). Системи, які діють на основі застосування сучасних методів ШІ, виграють за рахунок використання мов сценаріїв також і в тому, що ці мови повністю забезпечують взаємодію основних компонентів. Рішення можуть бути отримані логічним шляхом на підставі вихідних фактів. Ми будемо керуватися даними про властивості зброї (наприклад, про ступінь завданої шкоди і темп стрільби) для вибору найбільш підходящої зброї.

Застосування декларативного підходу для визначення властивостей зброї. Це дозволяє надати засобам ШІ інформацію про скорострільність і очікувану шкоду.

Для визначення способу ранжування кожної з цих властивостей зброї служить система голосування. По суті, в розрахунок беруться багато різних характеристик і відбувається голосування з метою з'ясування того, яка властивість зброї являється найбільш важливою. Після цього кожен вид зброї оцінюється відповідно з отриманим числом голосів і відбувається повернення даних про найкращу зброю. Такий підхід діє успішно, оскільки

найчастіше кількість кандидатів на звання найкращої зброї більше двох, причому основна їх частина відповідає повністю допустимому варіанту вибору. Кожен аспект рішення розглядається більш детально: визначення властивостей зброї, організація роботи системи голосування, збір голосів, визначення характеристик і облік простих обмежень.

Властивості зброї

В документі Quake 2 Weapons and Combat FAQ можна знайти статистичні дані, що приводяться в табл. 1

Таблиця 1

Властивості зброї у грі Quake 2

Зброя	Темп стрільби	Шкода	Шкода заподіяна за секунду	Швидкість	Розсіювання
Бластер	0,5	20	40	1000	0
Дробовик	1	48	40	-	500/500
Автомат	0,1	8	80	-	300/500
Кулемет	0,05 і 0,025	6	120 і 240	-	300/500
Гранатомет	1	120*	120	400-800	0
Ракетна пускова установка	1,25	120*	96	650	0
Гіпербластер	0,1	20	200	1000	0
Рельсотрон	1,5	100	67	-	0

Таким чином, розглядаються такі властивості зброї, з урахуванням тієї послідовності, в якій вони наведені в табл. 1.

- Темп стрільби.
- Шкода, заподіяна з розрахунку на постріл.
- Шкода, заподіяна за секунду.
- Швидкість уражаючого елемента.
- Точність (величина, зворотна розсіюванню).

Дані про властивості, наведені в таблиці, повинні бути оголошені в окремому файлі, щоб їх можна було легко модифікувати. У сценарії передбачено включення цього файлу, для того щоб дані про властивості зброї можна було враховувати при втіленні логічного висновку.

Головна мета розробників комп'ютерних ігор — створити програму, яка б мала кілька рівнів складності, ігровий процес, викликала високий рівень зацікавленості гравця і дозволяла б йому при виборі правильних стратегій поведінки досягати перемоги. Крім того, перемога гравця не повинна бути досить легкою. Отже, при розробці програмного коду комп'ютерної гри необхідно мати досить гнучкий алгоритм підбору стратегій, що гарантують певний процент позитивних результатів для гравця. Цей відсоток визначає рівень складності гри. Саме проблемі математичного моде-

лювання алгоритму вибору стратегій комп'ютерної програми у відповідності до рівня складності гри присвячена дана робота. Вибір ботом зброї залежить від так званих параметрів керування. Розв'язання задачі визначення параметрів вибору базується на математичній теорії ігор.

Засоби ШІ повинні прийняти до розгляду зазначені факти і виробити рішення, що стосується того, наскільки підходить той чи інший вид зброї в поточній ситуації. Результатом застосування функції придатності стає одиничне значення, яке представляє рівень придатності зброї.

В даному випадку до складу характеристик входять компоновка середовища, відстані між учасниками гри, поточний рівень життєздатності та інші. Критерії прийняття рішення про те, який вид зброї повинен використовуватися, фактично представляють собою результати аналізу комбінацій цих характеристик [6].

В основі системи голосування лежить така ідея, що характеристики використовуються для вироблення рішення про те, яка властивість зброї в найбільшій мірі потрібна для їх реалізації, і сформована при цьому оцінка розглядається як голос на користь цієї властивості (див. рис. 2). На відміну від тієї системи голосування, яка застосовується в людському суспільстві, в

даній системі голосування можна простіше передбачити можливість подавати більше ніж один голос, і навіть є можливість використовувати дробові значення кількості голосів.

Після цього, як показано нижче, отримані голоси застосовуються для масштабування властивостей зброї. Якщо за якусь властивість подано багато голосів, то це означає, що властивість є важливою, тому її числове значення множиться на більше число; і навпаки, отри-

мання меншої кількості голосів свідчить про те, що властивість не є надто важливою.

$$f(w) = P_i(w) * V_i + P_n(w) * V_n \quad (1)$$

В наведеній формулі $f(w)$ позначає придатність зброї w , $P_i(w)$ позначає i -у властивість зброї, що розглядається (наприклад, швидкість та збиток від пострілу), а V_i показує загальну кількість голосів, поданих за цю властивість.

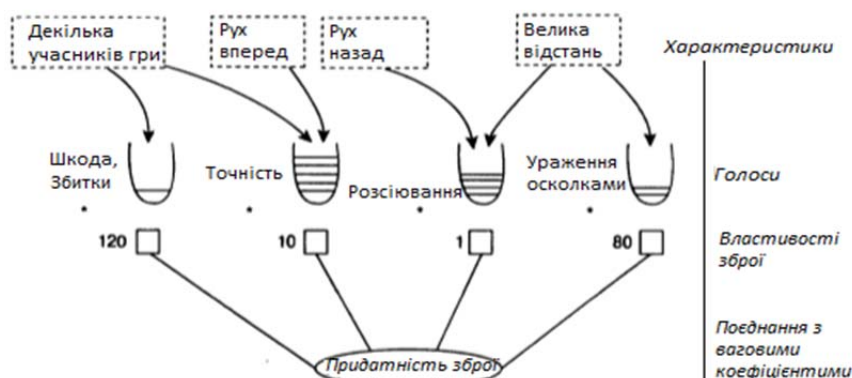


Рис. 2 – Схема організації системи голосування, що застосовується, в якій свій вклад в визначення кількості голосів вносять характеристики

Значення кількості голосів множаться на значення властивості зброї і обраховується зважена сума, що визначає придатність зброї.

Перш за все необхідно залучити до роботи експерта, щоб він визначив важливість властивостей в різних ситуаціях. Перш ніж почнеться голосування, необхідно виявити можливих кандидатів. У цьому розділі використовуються всі властивості зброї, які були перераховані

в табл. 1, і до них додаються дві зазначені нижче властивості.

- **Точність стрільби.** Визначається як величина, зворотна розсіюванню.
- **Потенціал ураження.** Визначається як максимальна шкода з розрахунку на постріл, але використовується інше визначення вагового коефіцієнта, для того щоб цей фактор став менш значущим.

Таблиця 2

Уточнення значимості характеристик зброї з допомогою голосування

Характеристика	Кількість голосів
Декілька гравців	Шкода, +4
Велика відстань	Точність, +2
Середня відстань	Кількість пошкоджень за секунду, +1; точність, +1
Коротка відстань	Кількість пошкоджень за секунду, +2
Рух уперед	Точність, +1
Рух назад	Розсіювання, +1
Рух в обмеженому просторі	Ураження осколками, +3
Напад із засади	Точність, +2
Пошук	Шкода, +1
Втеча	Потенційна ймовірність поразки, +атто(кількість боєприпасів)

На наступному етапі необхідно вибрати характеристики ситуації, що враховуються при проведенні голосування. Рішення про використання тих чи інших характеристик засноване, головним чином, на інтуїції, а їх відносна значимість, безсумнівно, повинна уточнюватися на етапі проведення експериментів, як показано в табл. 2, деякі характеристики володіють голосами, величина яких перевищує одиницю, а інші характеристики надають свої голоси на користь багатьох інших характеристик. Дані останнього рядка таблиці застосовуються незалежно від того, яку величину мають всі інші характеристики, а сам голос визначається пропорційно кількості боєприпасів.

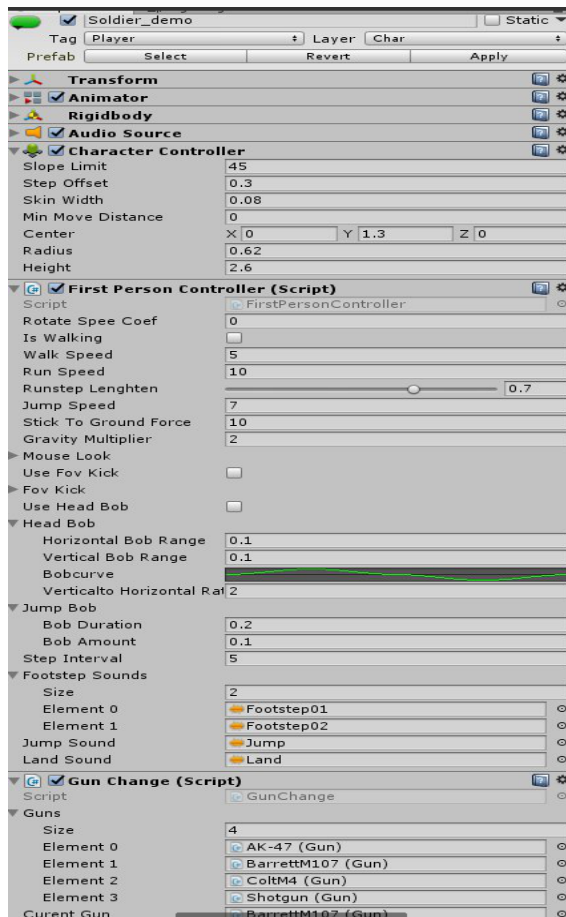


Рис. 3 – Інспектор у редакторі, що відображає налаштування бота

Після збору всіх значень голосів отримані данні множаться на значення кожної властивості зброї. Сума зважених значень властивостей являє собою кінцеву величину придатності зброї. Зброя з найбільшим зна-

ченням придатності вибирається як найкраща для налаштування бота (див. рис. 3).

На величину витрат часу при виборі зброї вручну повинні поширюватися деякі обмеження. Для цього застосовуються два описаних нижче обмеження.

- Зміна зброї не допускається, якщо попередня зміна відбулася лише за кілька секунд перед цим.

- Перехід до використання іншого виду зброї під час стрільби по противникам, у яких ще залишилися боєприпаси, не допускається; необхідно почекати до того моменту, коли противники сховаються або припинять вести вогонь.

Для успішного проведення етапу експериментування потрібно застосовувати велику кількість поправок. Повинна бути проведена коригування не тільки значень кількості голосів, але і вагових коефіцієнтів властивостей зброї. Налаштування обох цих значень можна проводити одночасно, оскільки немає необхідності в використанні вагових коефіцієнтів, якщо голоси виражаються числами з плаваючою точкою. Але якщо в розпорядженні розробника залишаються і ті і інші величини, то йому стає простіше домогтися правильного вихідного значення за допомогою невеликих послідовних коригувань.

Ми можемо розглядати дані про кількість голосів як цілочисельні і враховувати лише їх відносну значимість. В такому випадку з'явиться можливість налаштовувати значення вагових коефіцієнтів незалежно від властивостей зброї (див. рис. 4).

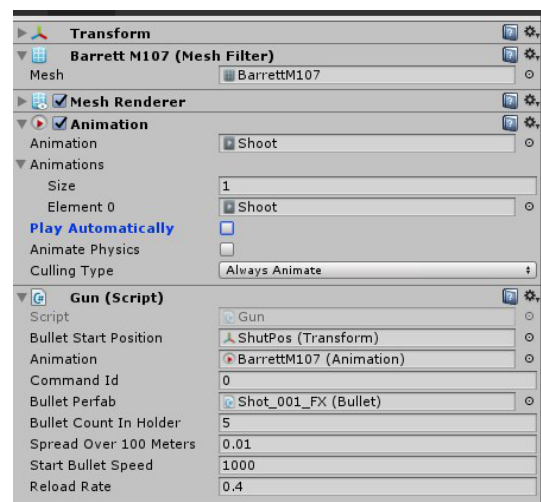


Рис. 4 – Інспектор у редакторі, що відображає налаштування зброї

На етапі експериментування необхідно організувати слідування за тим, як відбувається вибір зброї тим або іншим персонажем під час гри. Якщо отримане рішення є сумнівним, то проводиться аналіз процедури голосування для визначення того, у чому була допущена помилка. Зокрема, є можливість визначити, яка властивість зброї отримала занадто багато голосів, і виявити характеристики, які внесли найбільший вклад. Потім значення таких характеристик коригуються до тих пір, поки сумарна кількість голосів не починає входити в межі розумного. Після цього, як тільки стане ясно, що голоси, здебільшого, врівноважені, можна приступати до коректування вагових коефіцієнтів, що відносяться до кожної властивості зброї. Ці значення можна коригувати, виявляючи ті властивості, які внесли занадто великий вклад в остаточну оцінку при-

датності, і зменшуючи їх вагові коефіцієнти. Привабливою особливістю дерев рішень є те, що вони можуть знайти дуже широку сферу застосування. З іншого боку, велика кількість способів, що дозволяють забезпечити вибір зброї. Як і слід було очікувати, кількість можливих варіантів спільного застосування дерев рішень і таких способів ще більше. Проте, як показано на рис. 5. найбільш підходящими являються чотири варіанти, перераховані нижче.

1. Навчання вибору відповідної зброї.
2. Навчання оцінці придатності зброї.
3. Визначення за допомогою навчання властивостей зброї.
4. Навчання оцінці важливості властивостей.

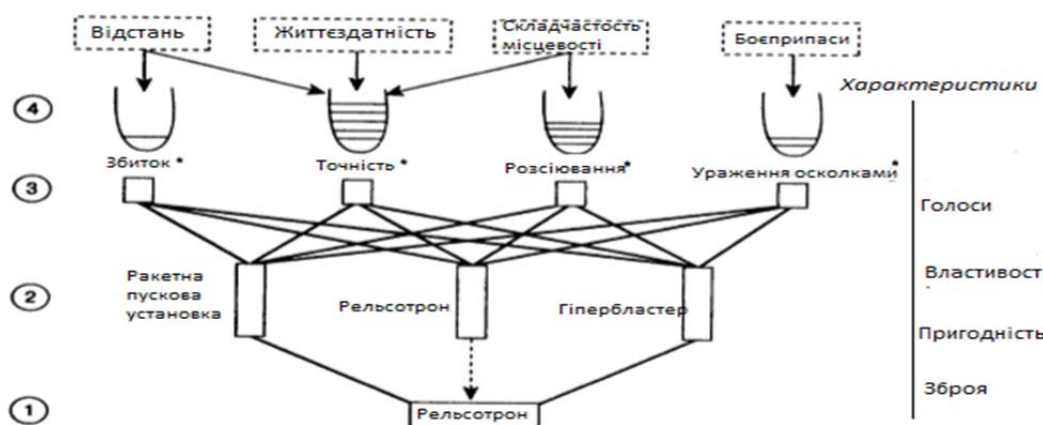


Рис. 5 – Концепції і процеси, пов'язані з прийняттям рішень по вибору зброї

Навчання вибору відповідної зброї полягає в тому, що дерево рішень застосовується для отримання інформації про зброю, найкращу в кожній ситуації. Такий підхід рівносильний формуванню відображення з множини характеристик середовища на тип зброї. З точки зору теорії, в цьому підході зосереджується все, що потрібно для вибору зброї. Але, на жаль, як описано нижче, використання єдиного компонента системи ШІ для рішення лише завдання визначення найкращої зброї призводить до виникнення на практиці багатьох проблем.

- Дерево рішень повертає тільки одну пропозицію на вибір зброї, тому не може застосовувати в тих умовах, коли необхідна зброя недоступна.

- Дерево рішень повертає єдиний варіант і не дозволяє зрозуміти хід процесу, в якому цей варіант був сформований.

- Засоби ШІ повинні мати в своєму розпорядженні якийсь спосіб самостійно визначати найкращу зброю для контролю над навчанням дерева рішень.

Очевидно, що для рішення вказаних проблем можуть бути запропоновані три різних способи, що дозволяють більшою чи меншою мірою знайти вихід з положення. По-перше, можна продублювати дерева рішень, щоб можна було використовувати різні дерева в залежності від того, які види зброї є в наявності, але для реалізації цього методу потрібно більше пам'яті, а процес навчання сповільнюється. По-друге, можна вказати наявні види зброї в якості додаткових вхідних

даних, але це призводить до комбінаторного вибуху обсягу простору задачі, тому спроба наближеного формування дерева рішень може викликати помилки. По-третє, можна зберігати результат вибору зброї, отриманий з допомогою дерева рішень, у вигляді списку видів зброї, ранжований по пріоритетам, але для цього потрібна адаптація дерева рішень, щоб його можна було використовувати в умовах наявності багатовимірних змінних відгуку [6].

Короткий висновок полягає в тому, що всі ці способи є застосовними, але не ідеальними. Більш прийнятний підхід може передбачати навчання оцінки придатності зброї в кожній ситуації, як описано в наступному пункті

Завдяки використанню такої методології завдання коригування всіх розглянутих значень стає трохи простіше. Крім того, розглянута система голосування має ту перевагу, що її можна відносно легко доповнити. Зокрема, в цю систему голосування можна легко ввести нові характеристики, причому ці характеристики, в основному, стають незалежними від інших існуючих характеристик. Отримана в результаті система відносно

непогано справляється із завданням вибору зброї. Безумовно, поведінка цієї системи далека від оптимальної, але її практично неможливо відрізнити від поведінки людей-гравців.

Недоліком даної системи голосування є те, що для її налаштування потрібне проведення етапу експериментування, описаного вище, оскільки результати підрахунку голосів не завжди стають відомими. У таких випадках неможливе застосування методів контрольованого навчання, виявляється також проблема, пов'язана з тим, що прийняті припущення не завжди є обґрунтованими; і дійсно, навички учасника гри рідко відповідають тим характеристикам зброї які були визначені на підставі точних статистичних даних. З цього випливає, що в своїй поведінці система голосування не підлаштовується до здібностей окремих гравців.

В наступних публікаціях буде наведено опис архітектури використання нейронної мережі FALCON, її алгоритму навчання та особливостей застосування та для прогнозування переміщення супротивника.

REFERENCES

1. Abu-Mostafa Y. S., Jaques J. N. St. Information capacity of the Hopfield model//IEEE Trans. Inform. Theory. 1985. V. 31. P.461.
2. Ackley D. H., Hinton G. E., Sejnowski T. J. A learning algorithm for Boltzmann machines//Cognit. Sci. 1985. V. 9. N 1. P. 147-169.
3. Amari S. Field theory of self-organizing neural networks//IEEE Trans. Syst., Man, Cybern. 1983. V. 13. P. 741.
4. Takefuji D. Y. A new model of neural networks for error correction.//Proc. 9th Annu Conf. IEEE Eng. Med. and Biol. Soc., Boston, Mass., Nov. 13-16, 1987. V. 3, New York, N. Y., 1987 - p. 1709-1710.
5. Ténk D. U., Khopfyld D. D. Kollektivni obchysleniya v neyronopodobnykh elektronnykh skhemakh .// V sviti nauky. 1988 r. N 2. S. 44-53.
6. Rokach L. Data Mining with Decision Trees. Theory and Applications / L. Rokach, O. Maimon. – London : World Scientific Publishing Co, 2008. – 264 p
7. Dzheffri E. Hinton. Kak obuchayutsya neyronnyie seti.// V mire nauki - 1992 - N 11 - N 12 - c. 103-107.
8. Shitikov V.K., Rozenberg G.S., Zinchenko T.D. Neyrosetevoe modelirovanie: mnogoslonyiy perseptron [WWW document]. www.ievbran.ru/kiril/Library/Book1/content394/content394.htm
9. Metodyi obucheniya neyrosistem [WWW document]. http://zdo.vstu.edu.ru/umk/html/manual/L5_6.html Soldatova O.P., Semenov V.V. Primenenie neyronnyih setey dlya resheniya zadach prognozirovaniya// Elektronnyy nauchnyy zhurnal «issledovano v Rossii» http://zdo.vstu.edu.ru/umk/html/manual/L5_6.html
10. Osovskiy S. Neyronnyie seti dlya obrabotki informatsii / Per. s polsk. I.D. Rudinskogo. – M.: Finansy i statistika, 2004. – 344 s.

*Рецензент: д.т.н., проф. Мусієнко М. П.
Чорноморський національний університет
імені Петра Могили*



АНАЛІЗ БЕЗПЕКИ ІНТЕРФЕЙСУ 802.11 (WI-FI З'ЄДНАНЬ), КЛАСИФІКАЦІЯ ПАРОЛІВ

УДК 004.056.5

БОСКІН Олег Осипович

старший викладач кафедри інформаційних технологій ХНТУ.

Наукові інтереси: людино-машинна взаємодія, інформаційна безпека.

e-mail: bbbosss@i.ua

МАЗМАНЯН Сергій Русланович

Студент кафедри інформаційних технологій ХНТУ.

Наукові інтереси: людино-машинна взаємодія, тривимірна графіка, інформаційна безпека.

e-mail: sergey.mazmanian@gmail.com

ЛЕВИЦЬКА Анна Михайлівна

Студентка кафедри інформаційних технологій ХНТУ.

Наукові інтереси: людино-машинна взаємодія, інформаційна безпека, тривимірна графіка.

e-mail: good.anna.marya@gmail.com

В сучасному світі з великою інформаційною перевагою, коли кожен день ЗМІ повідомляють про хакерські атаки на інформаційні ресурси державних установ та викрадення персональних даних різних організацій та приватних осіб, безпекою не опікуються лише вкрай недалекоглядні користувачі.

У жовтні 2016 року кількість підключень до сайтів з мобільних пристроїв у всьому світі вперше перевищила використання Інтернету зі стаціонарних комп'ютерів і ноутбуків, свідчать дані аналітичної компанії StatCounter. Частка мобільних пристроїв в жовтні 2016 року склала 51,3% від загальної кількості пристроїв, яка використовують Інтернет.

Серед персональних комп'ютерів/ноутбуків та мобільних пристроїв найбільш розповсюдженим типом інтернет з'єднання є стандарт 802.11(Wi-Fi).

Більшість користувачів не здогадується, що кваліфікований зловмисник може заволодіти персональними даними користувача ноутбука або смартфона на базі Андроїд або IOS, використовуючи стандартні пакети спеціалізованих програм, що знаходяться у вільному доступі.

ПОСТАНОВКА ЗАДАЧІ

Задачами розглянутого у статті дослідження є:

- аналіз безпеки інтерфейсу бездротових мереж зі стандартом шифрування WPA / WPA2,
- класифікація та кластеризація паролів, які були застосовані користувачами при аутентифікації.

Дані для дослідження були зібрані з відкритих джерел, зокрема [1], а також з анонімного опитування студентів ХНТУ спеціальності 121 «Інженерія програмного забезпечення» (понад 40000 паролів).

Класифікація та кластеризація проведена на основі запропонованих авторами стереотипів паролів.

ОСНОВНА ЧАСТИНА

До WPA, що був визначений в специфікації 802.11i, для забезпечення безпеки бездротових з'єднань використовувався WEP. Пізніше, після масових розкрадань персональних даних користувачів (зокрема номери кредитних карт), з'ясувалося що алгоритм шифрування WEP досить вразливий і було розроблено алгоритм WPA. Незабаром, після повсюдного впровадження WPA

шифрування, WEP було визнано застарілим, та не рекомендовано до використання[2].

Специфікація 802.11i складається з трьох основних частин, організованих у 2 рівні. Це зображено у таб.1

Таб. 1

Структура специфікації 802.11i

802.1x	
RC4-based TKIP	AES-based CCMP

Нижній рівень представлений алгоритмами шифрування, такими як Temporal Key Integrity Protocol

(TKIP) та CBC-MAC protocol (CCMP). Задля зворотної сумісності з WEP, TKIP використовує швидкий потоковий алгоритм шифрування RC4. CCMP базується на Advanced Encryption Standart (AES) – одному з найпопулярніших алгоритмів шифрування із симетричним ключем.

На верхньому рівні лежить 802.1x, що є стандартом портового контролю доступу. 802.1x забезпечує надійну аутентифікацію та поширення ключа шифрування. Обидва компоненти були відсутні у початковому стандарті 802.11[3].

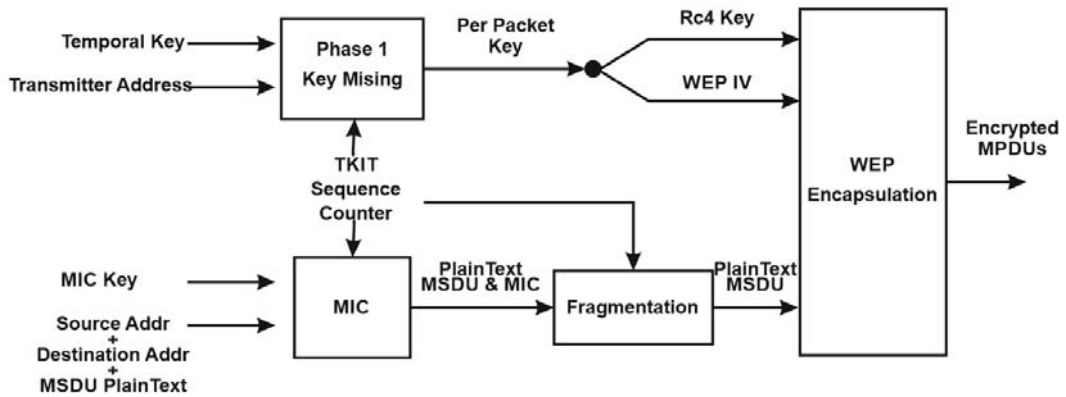


Рис. 1 Алгоритм шифрування за протоколом TKIP

Протокол TKIP (Temporal Key Integrity Protocol - протокол тимчасової цілісності ключа) виконує функції забезпечення конфіденційності і цілісності даних. Функціонально TKIP є розширенням WEP. Аналогічно з WEP, у ньому використовується алгоритм шифрування RC-4, але більш ефективний механізм управління ключами. Протокол TKIP генерує новий секретний ключ для кожного переданого пакета даних. Відрізняється також і механізм генерації ключа – його отримують із трьох

компонентів: базового ключа довжиною в 128 біт (TK), номеру переданого пакета (TSC) та MAC-адреси пристрою-передавача (TA). Також в TKIP використовується 48-розрядний вектор ініціалізації (IV - initialization vector). Для уникнення повторного використання IV використовується наскрізний лічильник пакетів (TSC) довжиною 48 біт. Він постійно збільшується, скидаючись в 1 тільки при генерації нового ключа. Молодші 16 біт TSC включаються в новий IV (рис. 2)[4].

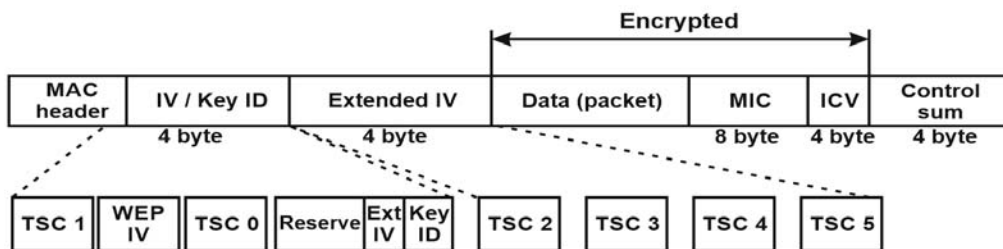


Рис. 2 Пакет після шифрування TKIP

Основна відмінність CCMP від TKIP і WEP – централізоване управління цілісністю пакетів, яке виконується на рівні AES. Пакет CCMP збільшений на 16 октетів, заголовок складається з трьох частин: PN (номер пакета, 48-розрядний), ExtIV (вектор ініціалізації), та ідентифікатора ключа (рис. 3). Алгоритм інкапсуляції даних з використанням CCMP:

1. Номер пакета збільшується на якесь число, щоб уникнути повторення пакетів;
2. Створіть нові аутентифікаційні дані;
3. Створити службове поле попсе;
4. Номер пакета і ідентифікатор ключа поміщаються в заголовок пакета;
5. Поле попсе і додаткові аутентифікаційні дані шифруються з використанням тимчасового ключа.

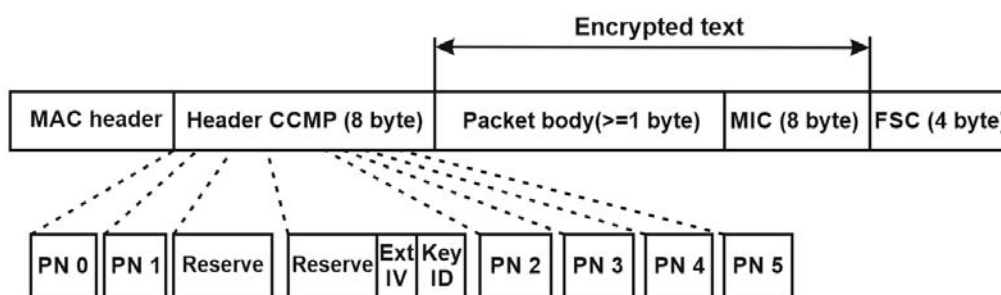


Рис. 3 Структура зашифрованого пакету WPA2

Алгоритм декапсуляції даних з використанням CCMP:

1. Створюються поля додаткових ідентифікаційних даних та поле попсе з використанням даних пакета;
2. Поле додаткових ідентифікаційних даних витягується з заголовка зашифрованого пакета;
3. Витягується поле A2, номер пакета та поле пріоритету;
4. Витягується поле MIC;

5. Виконується розшифровка пакета та перевірка його цілісності, з використанням шифротексту пакета, додаткових ідентифікаційних даних, тимчасового ключа та власне MIC;
 6. Виконується збірка пакета в розшифрованому вигляді;
 7. Пакети з повторюваним номером відкидаються.
- Даний метод шифрування в бездротовій мережі на даний момент є найбільш надійним.

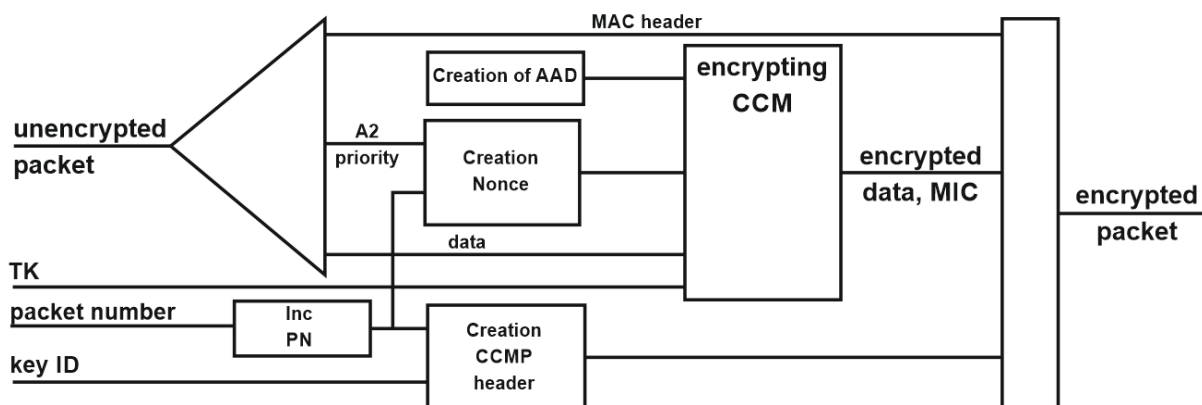


Рис. 4 Структурна схема протоколу шифрування CCMP

Корпоративна бездротова локальна мережа (WLAN) складається з трьох основних об'єктів – автентифікатор або точка доступу (AP) у 802.11, клієнт-

ський пристрій у 802.11 та сервер автентифікації (AS). Автентифікатор – це порт, який забезпечує автентифікацію та спрямовує трафік відповідним об'єктам в

мережі. Клієнтський пристрій – порт, що має доступ до мережі. Сервер автентифікації виконує фактичну автентифікацію. AS може бути в AP або бути підключеним до

дротової мережі. Сервер RADIUS – це типовий AS, який використовується сьогодні.

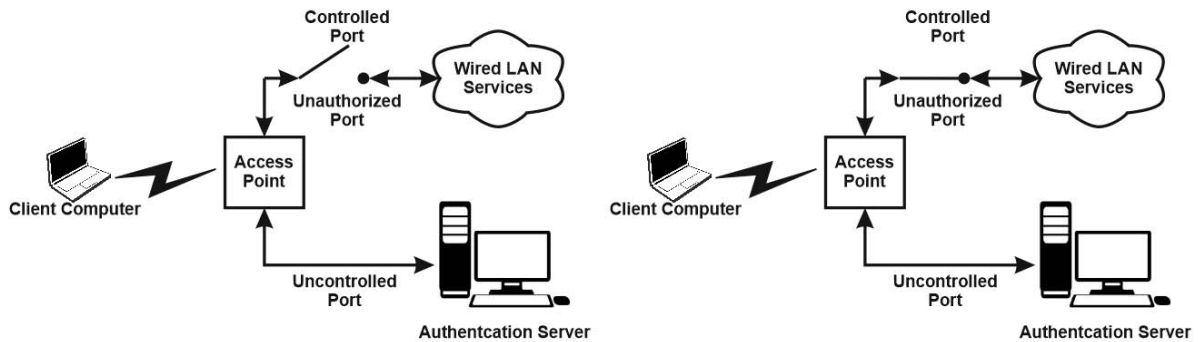


Рис. 5 Спрощена схема устрою мережі WLAN

Перш ніж авторизований заявник може дозволити спілкуватися з АС після його санкціонування, заявник може отримати доступ до інших ресурсів у мережі. Процес автентифікації є взаємним. І клієнт, і мережа повинні довести свою ідентичність[3,4].

Якщо у корпоративних мережах стандарту 802.11 способи авторизації можуть бути різними, то в повсякденному житті надання користувачу доступу до мережі відбувається після введення кодової фрази (пароля), довжиною від 8 до 63 символів[1,5,6].

Є два види ключів в мережі з підтримкою 802.1x:

За визначенням стандарту WPA і умови дотримання винятків прошивки деяких роутерів досліджені паролі в загальному випадку можна розділити на сім великих груп:

1. Сеансовий ключ або парний ключ. Сеансовий ключ використовується лише між клієнтом та його AP.
2. Груповий ключ. Груповий ключ генерується один на всіх клієнтів, підключеними одного AP. Він використовується для багатоадресного трафіку.

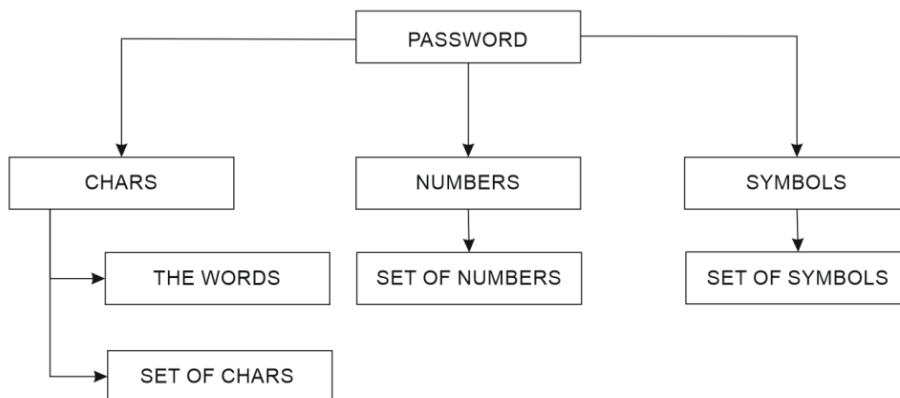


Рис. 6 Основні стереотипи паролів

Як видно на рисунку 6, основу класифікації паролів складають три великі групи: «букви», «цифри» і «спеціальні символи», які в свою чергу діляться на: «слова», «набір букв», «набір цифр» і «набір символів», з яких

складаються різні комбінації, такі як «множина символи + цифри», «множина символи + літери» тощо.

У числовому співвідношенні дані наведені на рис.7:

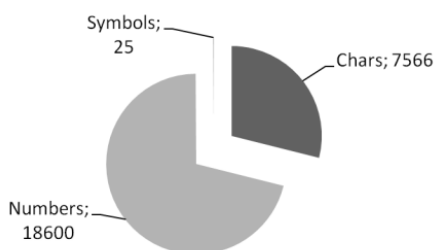


Рис. 7 Діаграма 1

З наведеної діаграми можна зробити висновок про те, що серед паролів, які складаються з однотипного набору (цифри, букви або спеціальні символи (далі просто символи) переважають числові. Тобто при виборі пароля (секретного слова) користувачі віддають перевагу набору з цифр. Які це набори буде розглянуто пізніше.

Значну частину (близько 1/3) всіх проаналізованих стереотипів паролів складають комбіновані стереотипи (рис.8).



Рис. 8 Діаграма 2

Співвідношення між цифровими і комбінованими стереотипами приблизно однаково (рис.9):

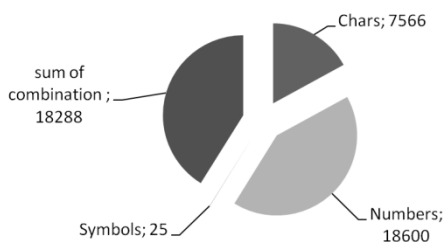


Рис. 9 Діаграма 3

Авторами статті були виділені (і в подальшому підтверджені експертами) такі стереотипи паролів:

- 8 цифр (в тому числі дати / дати народження),
- набір букви + цифри,
- слово + набір цифр,
- 10 цифр або більше, в тому числі номери телефонів,

- 1 слово,
- 2 слова,
- набір букв,
- ім'я + набір цифр,
- ім'я + дата / дата народження,
- 9 цифр,
- 3 слова або більш,
- набір букви + цифри + символи,
- слово + символи,
- слово + цифри + символи,
- набір цифр + символи,
- набір букви + символи,
- дата / дата народження + набір символів,
- набір символів.

Зв'язки між простими і складними стереотипами паролів представлені на рис. 10.

В таб. 2 наведені кількісні значення кожного з стереотипів.

Таб. 2

Кількісні значення стереотипів паролів

Кількість	Стереотип
14136	8 numbers
7882	Set of chars + Set of numbers
7284	The words + Set of numbers
3710	10 numbers or more
3121	1 word
2533	2 words
1241	Set of chars
1109	Name+ Set of numbers
932	Name + Date/Birthday
754	9 numbers
671	3 words or more
517	Set of chars + Set of numbers + Set of symbols
190	The words + Set of symbols
160	The words + Set of numbers + Set of symbols
122	Set of Numbers+Set of symbols
53	Set of chars + Set of symbols
39	Date/Birthday + Set of symbols
25	Set of symbols
44479	Total

Вочевидь, найбільш популярними у використанні є паролі з 8-ми цифр. До цього числа входять дати, дні народження (наприклад, 14121997), як найпопулярніший стереотип, пін-код від WPS роутера, який встановлений паролем за замовчуванням. Частота використання 8-ми цифр пов'язана ще з стереотипністю мислення самого користувача і особливостями пам'яті

людини. Набагато простіше запам'ятати «що є паролем», ніж посимвольного значення. При налаштуванні роутера BIOS запитує користувача ввести пароль з мінімум (!) 8-ми символів і максимум 63-х. Цей факт є причиною великої частоти використання не тільки стереотипу пароля «8 цифр», а також і стереотипу «на-

бір символів + набір цифр». Тут одним з лідерів за частотою використання є «4 букви + 4 цифри». У цьому стереотипі очікувано повторюється стереотипність мислення користувача, так як був запит на введення мінімум 8-ми символів.

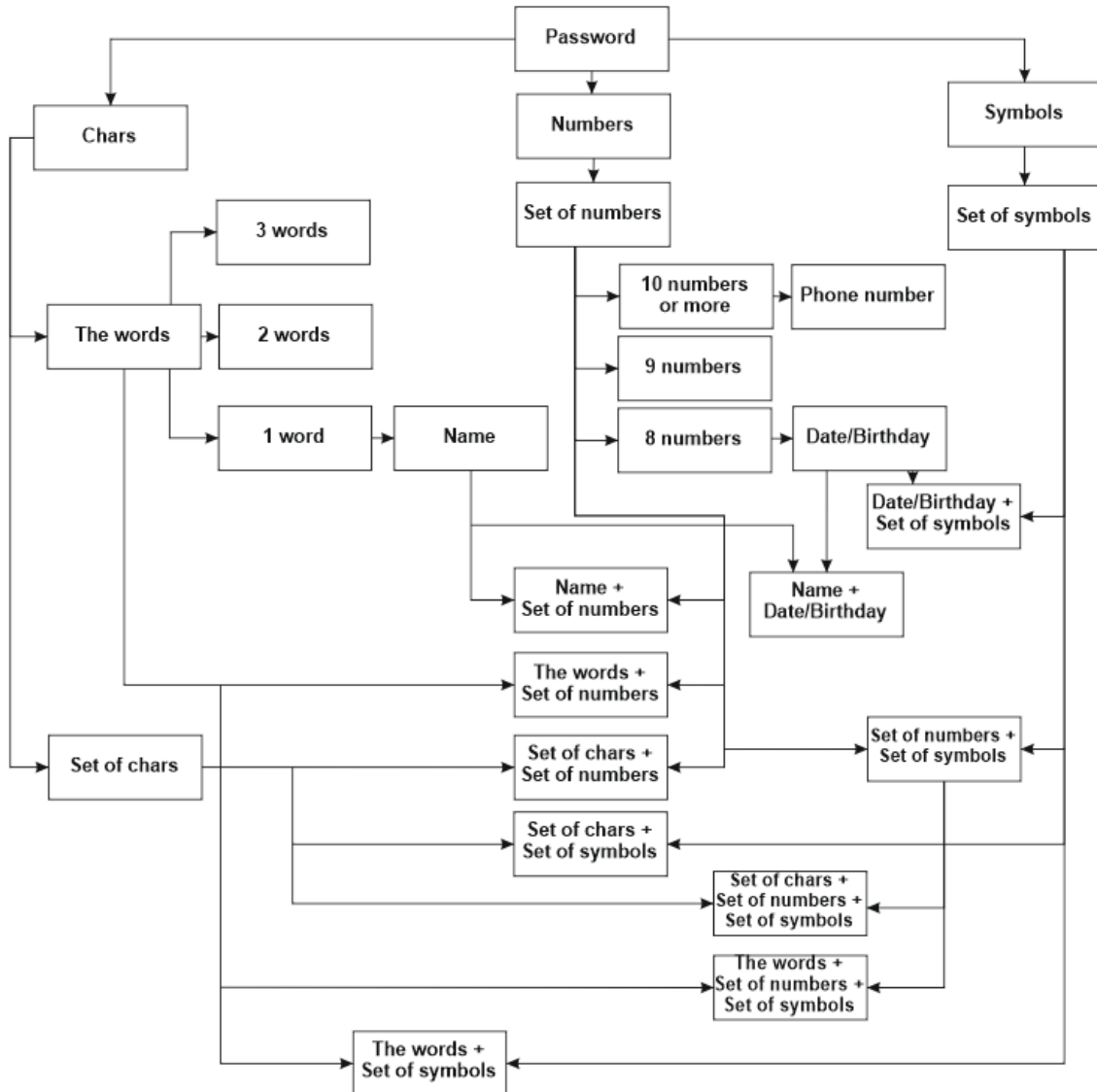
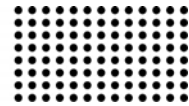
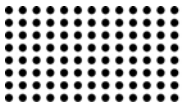


Рис. 10 Класифікація стереотипів паролів

Як приклад стереотипу «4 букви + 4 цифри» розглянемо набір «abcd1234». Набором символів найчастіше є якась аббревіатура, що легка для запам'ятовування користувачем, така як «sssr» або «ODMA», а в якості цифр, як правило, виступає рік - «1998», але іноді зустрічаються «lena1234».

На третьому місці за популярністю знаходиться стереотип «слово + набір цифр». В якості слова може виступати як власна назва, так і слово з загальноживаної мови, з повсякденного спілкування. Наприклад: «password123456» або «partkom1». Складність розшиф-



ровки подібних стереотипів невелика і буде розглянута нижче.

Стереотип «10 цифр і більш» являє собою, найчастіше, номер мобільного телефону, що складається з коду оператора (3 цифри) і самого номера (7 цифр). У цей набір може входити також код країни. При уявній скла-

дності розшифровки через довжину, насправді даний стереотип обчислюється за пару годин, в залежності від потужності відеокарти.

Для наочності наведемо таблицю частоти використання всіх наведених стереотипів паролів у вигляді діаграми (рис.11):

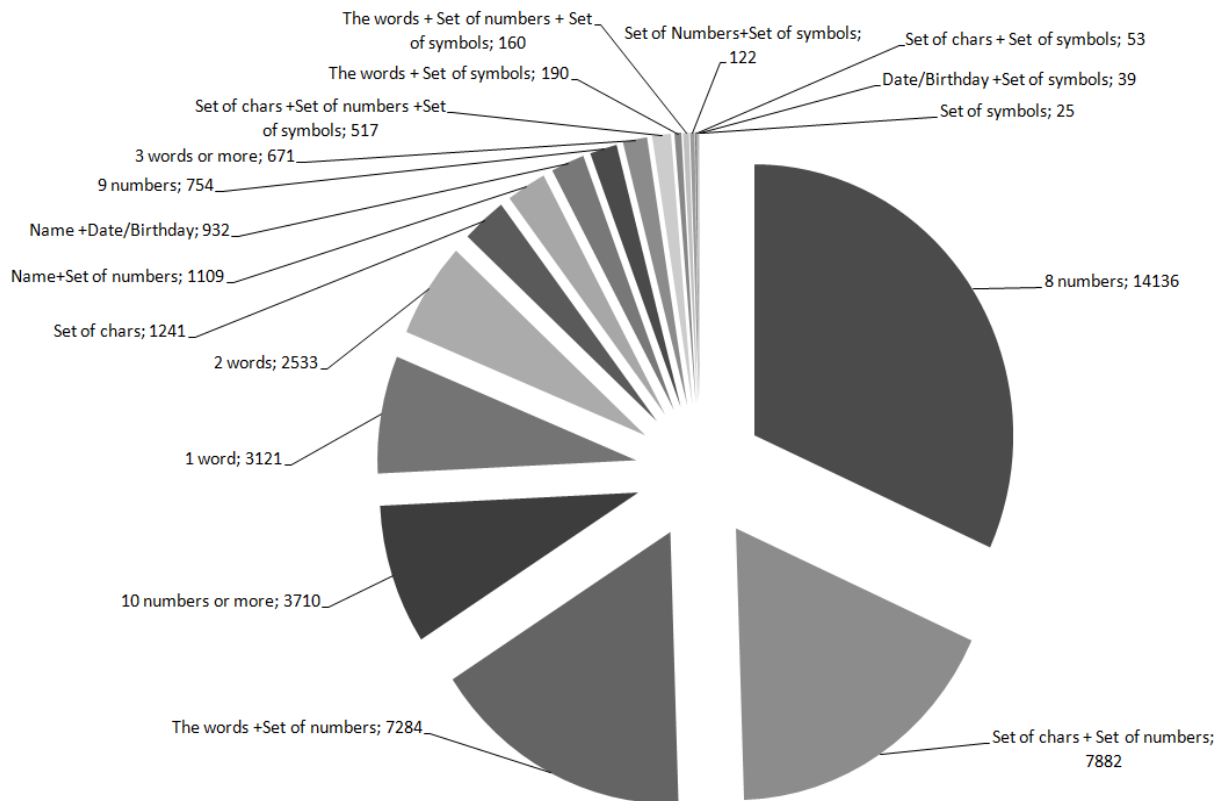


Рис. 11 Діаграма 4

З рис.11 і з даних відсортованої таб.2 нескладно помітити, що частота використання простих стереотипів набагато вище, ніж складних. Що зайвий раз свідчить про занижений рівень інформаційної культури більшості користувачів і необізнаності про можливі види атак на їхні персональні дані через уразливість використовуваного пароля. Характерними представниками складних стереотипів можна назвати «два слова», «три слова», «слово + набір символів», «слово + набір символів + набір символів» (рис.12).

Як видно з діаграми частоти використання складних стереотипів беззаперечно перевагу користувачами віддається «набору букв + набір цифр» і «слову + набір цифр». Це цілком природно пояснюється тим, що кори-

стувачеві простіше запам'ятати такий пароль. Як характерні приклади таких стереотипів можна розглядати «abracadabra123456789».

Одним з найшвидших інструментів по відновленню паролів визнаний Hashcat [7,8]. Одна з різновидів мультиплатформенна oclHashcat використовує CUDA технології. CUDA - це архітектура паралельних обчислень від NVIDIA, що дозволяє істотно збільшити обчислювальну продуктивність завдяки використанню GPU (графічних процесорів).

OclHashcat відновлює безліч хеш-функцій (MD4, MD5, Half MD5, SHA1 і т.д., повний список на офіційному сайті Hashcat [8]), включаючи WPA / WPA2, з використанням Multi-GPU (до 128 GPU!).

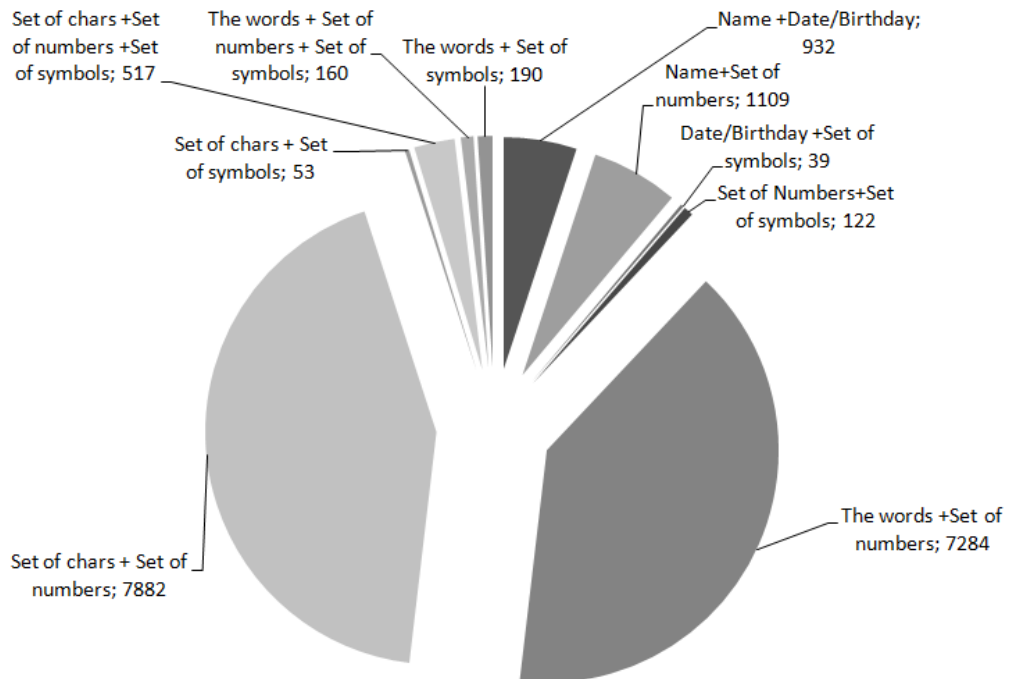


Рис. 12 Діаграма 5

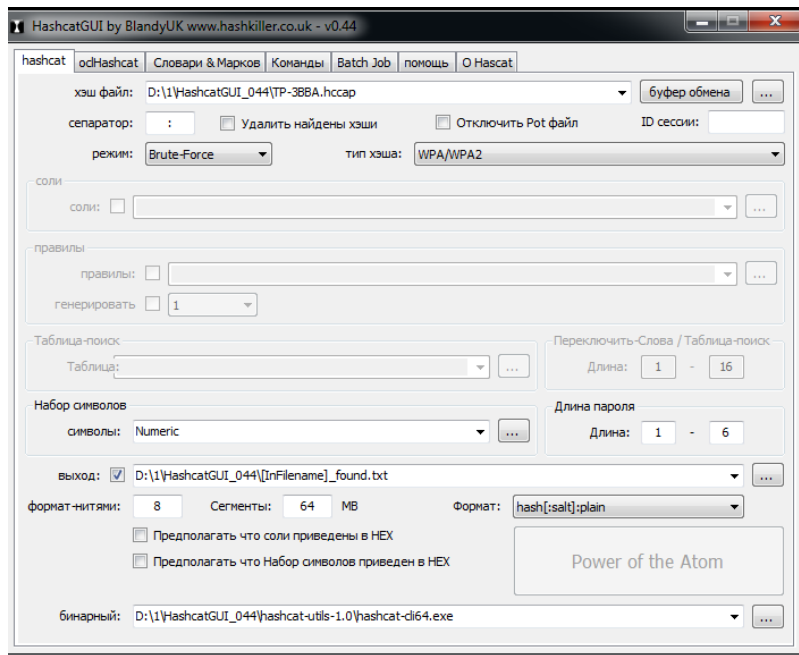


Рис. 13 Загальний вигляд GUI oclHashcat

Атаки проводяться за такими модами (рис.14):

- Straight (прямий перебір по словнику / списку словників), рис.15;
- Combination (комбінація двох словників);
- Brute-force (перебір по масці);
- Hybrid dict + mask (перебір по масці + словник), рис.16;
- Hybrid mask + dict (перебір по словнику + маска).

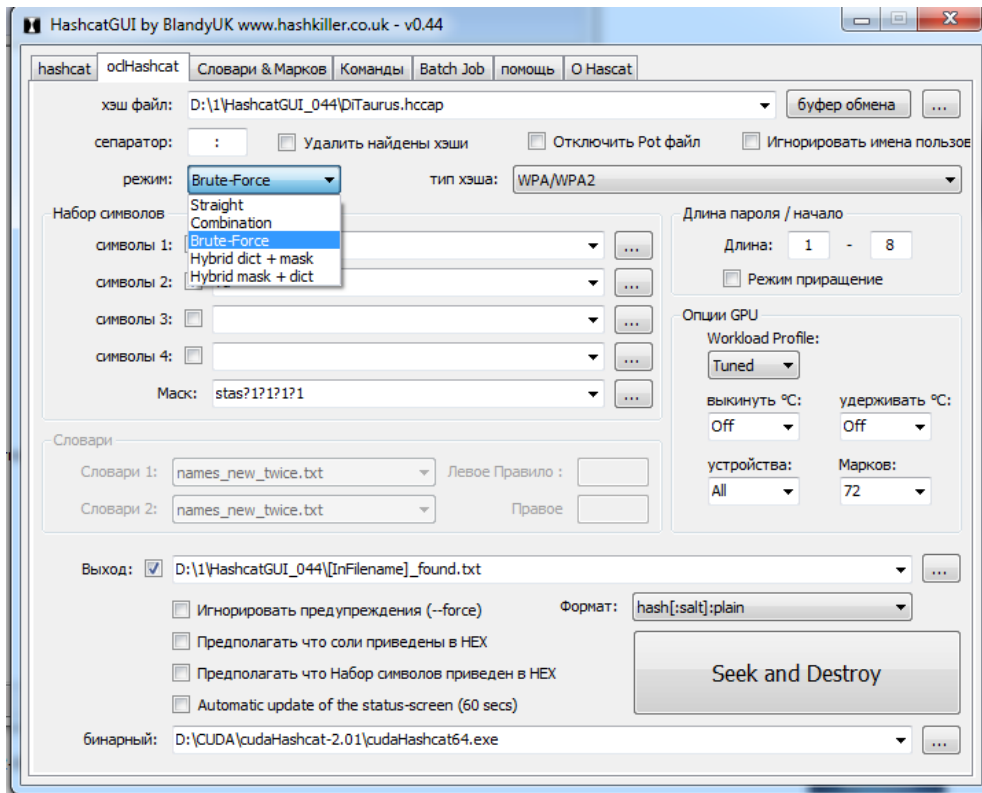


Рис. 14 Вибір типу атаки

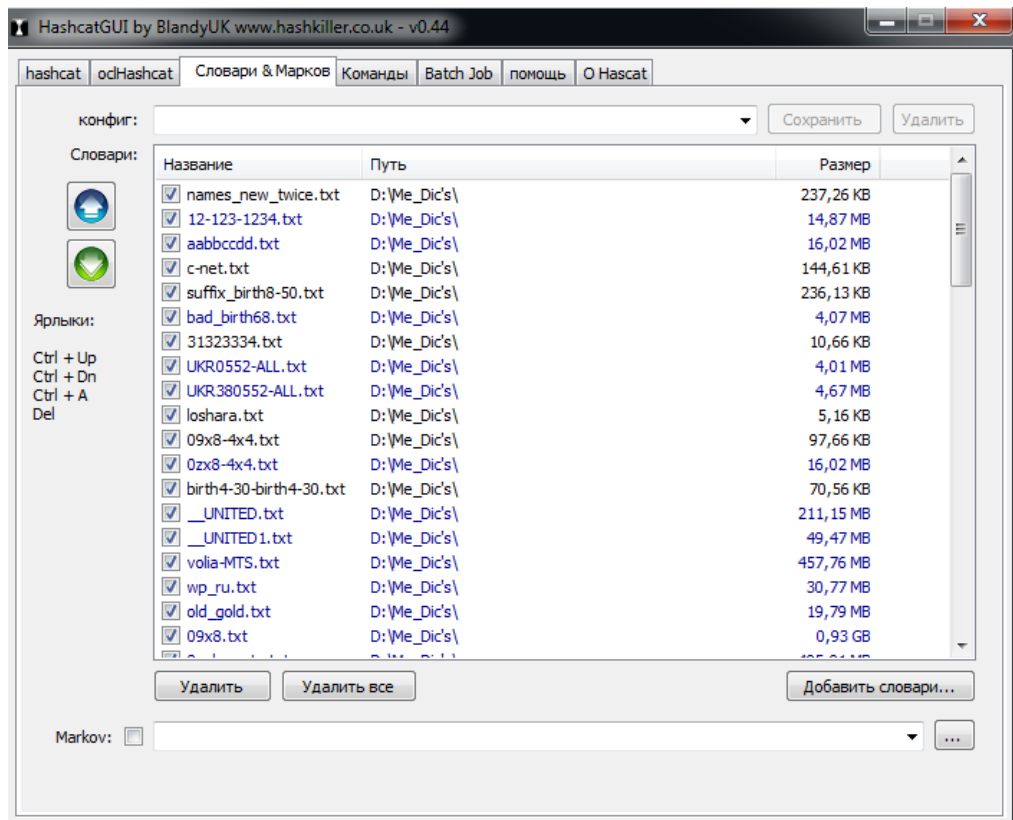


Рис. 15 Перелік словників

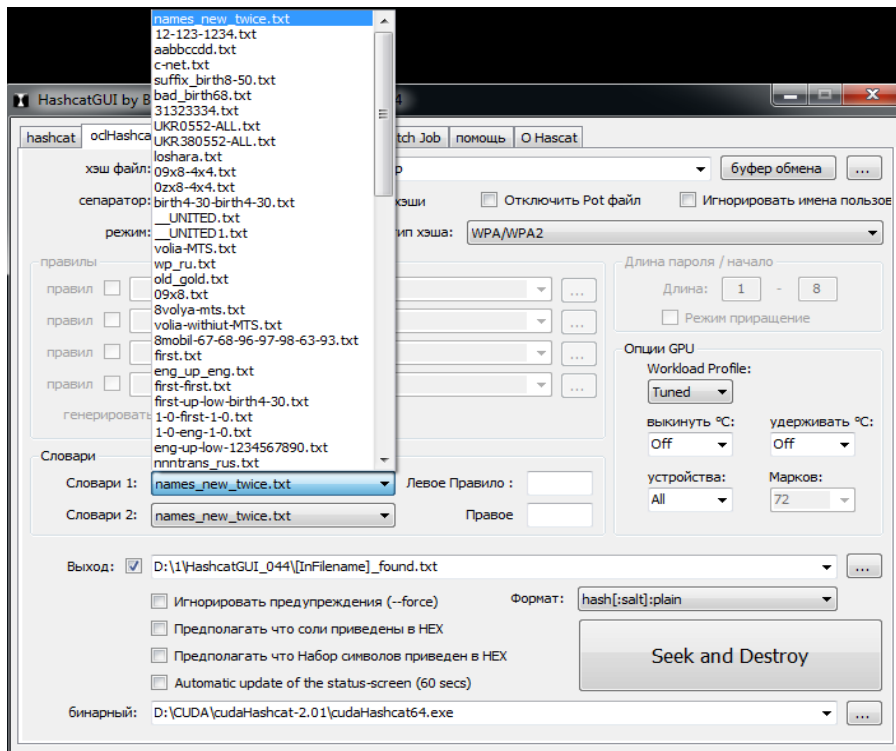


Рис. 16 Вибір словників для комбінації

На сьогоднішній день це найбільш потужний інструмент для відновлення / проведення атак на паролі. Остання версія v4.1.0 / 2018.02.21 знаходиться у вільному доступі на ресурсі [8].

В рамках даного дослідження використовувалася відеокарта GeForce GT 750M, яка при переборі за словником показує швидкість близько 18000 хешів за секунду (рис.17).

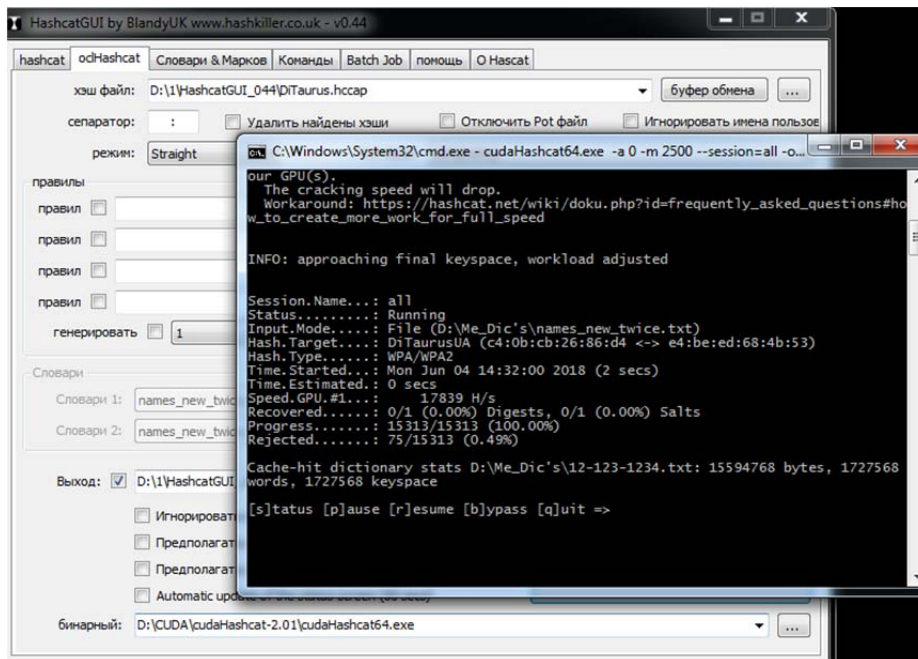
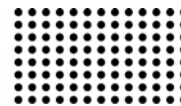
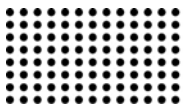


Рис. 17 Приклад роботи oclHashcat



При даній швидкості перебору паролів список дат від 01.01.1950 в форматі дд/мм/рр (наприклад «13031997») буде перерахований за 1 секунду. Що свідчить про те, що використання дат в форматі дд/мм/рр, або навіть з роздільниками, неприйнятно в рамках дотримання елементарних правил безпеки.

Використання в якості пароля номера мобільного або стаціонарного телефону також небезпечно, так як зловмисникові потрібно всього лише пара годин для перебору всіх мобільних операторів і комбінацій номерів. При цьому довжина пароля буде вагомою, 10 або 12 цифр і недосвідчений користувач вважатиме його досить надійним, але з використанням «стереотипного» (запропонованого авторами дослідження) підходу для підбору пароля зловмисник вирахує його досить швидко.

Розглянемо ще один приклад складного пароля «Volodymyr12091964». При уявній надійності стереотипу в 17 символів, посимвольний підбір якого займе не одну тисячу років, він досить простий з точки зору стереотипного підходу задля розшифровки WPA паролів, так як перебір згенерованого словника «ім'я + дата» займе менше години.

ВИСНОВКИ

Алгоритм шифрування WPA/WPA2 можна вважати достатньо надійним, але частина безпеки в бездрото-

вих з'єднаннях покладається на користувача. Тому безпека в бездротових з'єднаннях в більшій мірі залежить від надійності пароля. Нажаль, аналіз класифікації вживаних паролів свідчить про те, що більшість користувачів використовує ненадійні паролі, нехтуючи безпекою своїх персональних даних, та даних компаній та організацій.

Вибірку паролів можна вважати легітимною, збір даних проводився впродовж декількох років, в різних країнах і серед користувачів різних вікових категорій. Кількість зібраних даних можна вважати достатньою.

Деякі користувачі помилково вважають, що надійність залежить виключно від довжини паролю, але не усвідомлює, що зловмисник може використовувати стереотипний підхід.

Запропонований в статті метод класифікації та кластеризації паролів, які були застосовані користувачами при аутентифікації в бездротових мережах зі стандартом шифрування WPA / WPA2, значно скорочує час на підбор паролю.

В якості способу відновлення хешу паролю безальтернативно запропоновано використовувати oclHashcat, обчислювальні здатності якого значно перевищують навіть останні найпотужніші процесори в десятки разів, за рахунок використання CUDA технологій, в залежності від класу відео карти.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ:

1. Distributed WPA PSK auditor [Yelektronniy resurs] // Distributed WPA PSK auditor URL: <https://wpa-sec.stanev.org/?dicts> (data zvernennya 10.02.18).
2. Zegzhda D., Ivashko A.M. Osnovy bezopasnosti informatsionnykh sistem. – M.: Goryachaya liniya – Telekomu 2008. – 452 s.
3. P. Roshan, Dzh. Lieri. Osnovy postroyeniya lokal'nykh setey standart 802.11. – Per. s angl. – M.: Izdatel'skiy dom «Vil'yams», 2004. – 296 s.
4. Tomas Maufer. WLAN: prakticheskoye rukovodstvo dlya administratorov i professional'nykh pol'zovateley. – M.: KUDITS-Obraz, 2005. – 368 s.
5. Penetration Testing Services [Yelektronniy resurs] // Offensive Security. 2018. URL: <https://www.offensive-security.com/offensive-security-solutions/penetration-testing-services/> (data zvernennya 11.03.18).
6. Official site of Wi-Fi Alliance [Yelektronniy resurs] // Wi-Fi Test Suite 2018. URL: <https://www.wi-fi.org/certification/wi-fi-test-suite> (data zvernennya 11.03.18).
7. Instrumenty Kali Linux [Yelektronniy resurs] // © 2018 Instrumenty Kali Linux. All Rights Reserved. URL: <https://kali.tools/?p=538> (data zvernennya 11.05.18).
8. Hashcat advanced password recovery [Yelektronniy resurs] URL: <https://hashcat.net/hashcat/> (data zvernennya 11.05.18).
9. Malyuk, A.A. Informatsionnaya bezopasnost': kontseptual'nyye i metodologicheskiye osnovy zashchity informatsii: Uchebnoye posobiye dlya vuzov. / A.A. Malyuk. - M.: Goryachaya liniya -Telekom, 2004. – 280 c.
10. Shan'gin, V.F. Informatsionnaya bezopasnost' i zashchita informatsii / V.F. Shan'gin. - M.: DMK, 2014. - 702 c.
11. Chipiga, A.F. Informatsionnaya bezopasnost' avtomatizirovannykh sistem / A.F. Chipiga. - M.: Gelios ARV, 2010. - 336 c.

Рецензент: д.т.н., проф. Ходаков В.Е.

Херсонский национальный технический университет

МОДЕЛІ ЕЛЕКТРОННОГО НАВЧАННЯ ТА ВИМОГИ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

УДК 004.89:519.8

ГОРБАЧ Тетяна Вікторівна

науковий співробітник кафедри програмної інженерії ХНУРЕ

Наукові інтереси: Розробка і впровадження комп'ютерних навчальних програм, розробка алгоритмічного та програмного забезпечення для дистанційної освіти, розробка програмних засобів мультимедіа.

e-mail: yova.tanya@gmail.com.

СЛАВГОРОДСЬКИЙ Владислав Юрійович

аспірант кафедри програмної інженерії Харківського національного університету радіоелектроніки

Наукові інтереси: Математичне моделювання, розробка і впровадження комп'ютерних навчальних програм, розробка алгоритмічного та програмного забезпечення для дистанційної освіти

e-mail: kearir12@gmail.com.

ШУБІН Ігор Юрійович,

кандидат технічних наук, професор кафедри програмної інженерії Харківського національного університету радіоелектроніки

Наукові інтереси: Розробка і впровадження комп'ютерних навчальних програм, розробка алгоритмічного та програмного забезпечення для дистанційної освіти, розробка програмних засобів мультимедіа.

e-mail: igor.shubin@nure.ua.

КОВАЛЕВСЬКА Алла Володимирівна

доцент кафедри економіки підприємств міського господарства Харківського національного університету міського господарства імені А.Н. Бекетова

Наукові інтереси: Математичне моделювання, розробка алгоритмічного та програмного забезпечення для дистанційної освіти, розробка програмних засобів мультимедіа, економіко-математичне моделювання методів вирішення управлінських завдань.

e-mail: torpal@ukr.net.

ВСТУП

Сучасний стан освітньої системи України характеризується активними змінами на шляху до євроінтеграції та міжнародної співпраці в області вищої освіти. З цього погляду активізація наукових досліджень з питань ефективного управління на рівні вищого навчального закладу, зокрема на основі використання прогресивних інформаційних технологій, є важливим завданням сьогодення. Розвиток новітніх інформаційних технологій в освіті неодмінно спричиняє появу нових форм навчання. Поряд із очною та заочною формою дедалі частіше використовують електронну освіту. Відсутність стандартів та уніфікованого підходу до створення сис-

тем електронного навчання в Україні впливає не лише на ринок таких програмних продуктів, але й на якість дистанційної освіти. Одним із шляхів вирішення проблеми якості навчання є створення окремих педагогічних програмних засобів для кожного предмету. Але цей процес є дуже витратним як з огляду на матеріальні, так і на людські ресурси. Отже, постає питання розробки програмного забезпечення, яке б задовольняло вимоги щодо організації освітнього процесу на основі методології електронного навчання.

Огляд процесу розробки програмного забезпечення. Як правило, всі системи дистанційного навчання мають модульну структуру. Навіть якщо фізич-



но модулі важко виділити, вони легко виокремлюються на концептуальному рівні та рівні розробки. До таких модулів належить навчально-методичний модуль, який реалізує розроблення і публікацію навчального матеріалу; педагогічний модуль, який реалізує функції структурування і послідовності подання навчального матеріалу (НМ) та моделювання зв'язку студента з підсистемою контролю знань; інтерфейсний модуль (навігація між елементами навчального матеріалу); комунікаційний модуль (функції інтерактивної взаємодії та зворотного зв'язку); організаційно-адміністративний модуль (функції авторизації, захисту, фінансових операцій та інші).

Визначені покоління електронного та дистанційного навчання розрізняються не тільки наявністю того або іншого модуля, але й їхнім внутрішнім змістом та сукупністю зв'язків між ними. Знання предметної області, сформовані навчально-методичним модулем, організовані у відповідні інформаційні одиниці навчального матеріалу, визначають предмет вивчення. Педагогічний модуль визначає оптимальну послідовність подання навчального матеріалу. Крім того, до складу педагогічного модуля входить підсистема контролю знань студента, зазвичай у вигляді тестів [1]. У системах дистанційного навчання нового покоління рівень знань студента оцінюється також безпосередньо під час вивчення навчальних матеріалів. Модель підвищує ефективність навчання, оскільки знаючи кого навчати, навчальний процес максимально індивідуалізується, тобто адаптується. Інтерфейсний модуль забезпечує зв'язування інших модулів системи дистанційної освіти, а комунікаційний забезпечує взаємодію та зворотний зв'язок студента із системою. Організаційно-адміністративний модуль (або підсистема керування навчальним процесом) забезпечує функції авторизації, захисту, здійснення фінансових операцій та інші.

Еталонна модель об'єкта контенту для спільного використання (SCRM). Sharable Content Object Reference Model – еталонна модель об'єкта контенту для спільного використання – створення цього стандарту є першим кроком на шляху розвитку концепції Advanced Distributed Learning (ADL), оскільки цей стандарт визначає структуру навчальних матеріалів і інтерфейс середовища виконання, за рахунок чого навчальні

об'єкти можуть бути використані в різних системах дистанційного і комп'ютерного навчання.

Основними цілями проекту є надання освітньому співтовариству таких можливостей:

- 1) доступність – можливість пошуку і доступу до об'єктів, що знаходяться в різних місцях;
- 2) адаптивність – можливість налаштовувати навчання під індивідуальні або організаційні потреби;
- 3) ефективність – можливість скоротити час і вартість доставки знань до студентів;
- 4) здатність до перенесення – можливість перенести об'єкти, створені одним набором засобів розробки або платформою, на інші і використовувати без змін;
- 5) захист інвестицій – зміни технологій не ведуть до необхідності переробки об'єктів;
- 6) повторне використання об'єктів – можливість складати курси з найдрібніших об'єктів, гнучкість при використанні в різних контекстах.

Всі ці принципи успішно можуть бути дотримані в тому випадку, якщо з самого початку орієнтуватися на використання освітнього контенту в веб-середовищі [2].

Оскільки Web є ідеальним середовищем поширення та використання освітніх матеріалів, тому розробники SCORM зробили його сумісним з можливостями мережі з таких причин:

- 1) web-технології та інфраструктура швидко розширюють можливості освітніх технологій;
- 2) web-стандарти освітніх технологій ще не існують в розповсюдженій формі;
- 3) web-контент можна поширювати і використовувати в будь-якому середовищі (автономні системи або мережеві середовища).

ОПИС СИСТЕМИ

Система подається як набір сервісів, які конфігуруються залежно від вимог замовника. Замовником може виступати фізична або юридична особа, якій потрібно автоматизувати процеси навчання. Система будується з автономних сервісів, які є незалежними один від одного. Спількування між сервісами відбувається за допомогою HTTPS протоколу. Також замовник може надати свій сервіс, який підтримує контракт системи. Сервіси є незалежними та можуть бути реалізовані різними мовами програмування та розміщені на

різних апаратних платформах. Для повноцінного функціонування продукту програмне забезпечення розробляється з можливістю підключення до п'ятдесяти модулів одночасно, а також його надійності, відкритості до змін, захищеності.

В контексті роботи під системою розуміється проста взаємопов'язана структура, яка складається із таких компонентів:

- 1) сервіс зв'язку компонентів;
- 2) навчально-методичний сервіс;
- 3) педагогічний сервіс;
- 4) сервіс контролю знань;
- 5) сервіс авторизації та аутентифікації;
- 6) сервіс оповіщень.

Структура системи представлена на рис. 1.

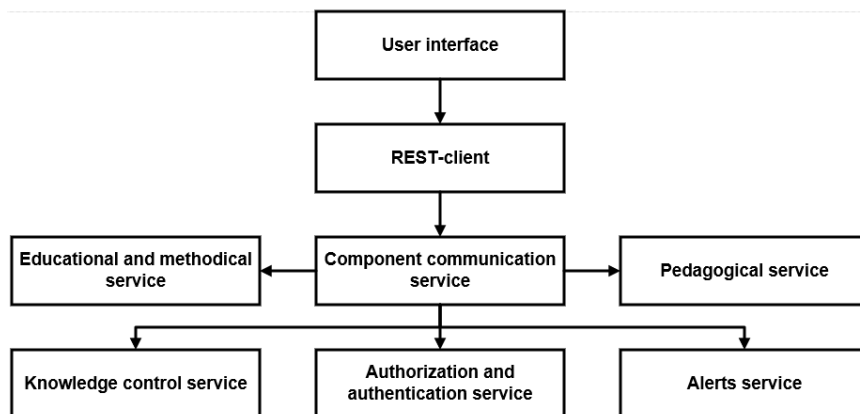


Рисунок 1 – Структура системи

Оскільки сервіси є змінними, то надалі будемо розглядати лише головний (незмінний) сервіс, а саме сервіс зв'язку компонентів.

ВИМОГИ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Функціональні вимоги описують внутрішню роботу системи, її поведінку (калькулювання даних, маніпулювання даними, опрацювання даних) та інші специфічні функції, які повинна виконувати система. Функціональні вимоги визначають, що система повинна робити [3].

Сервіс є головним вузлом, який приймає запити з REST-клієнта та перенаправляє на потрібний сервіс. Також він виступає внутрішнім маршрутизатором, який забезпечує комунікацію між внутрішніми сервісами. Для того щоб сервіс мав інформацію про сервіси всередині системи та їх контракти, при старті додатку відбувається реєстрація всіх сервісів.

До функціональних вимог сервісу зв'язку компонентів (рис. 2) належать:

- 1) створити конфігураційний файл;
- 2) редагувати конфігураційний файл;
- 3) видалити конфігураційний файл;
- 4) зареєструвати сервіси;

- 5) ввімкнути сервіс;
- 6) вимкнути сервіс.

Вимога «Створити конфігураційний файл» надає можливість створення файлу за конфігурацією. Для цього слід вказати назву файлу та додати основні атрибути, такі, як хости, порти та назви сервісів, які будуть використовуватися при такій конфігурації.

Прецедент «Редагувати конфігураційний файл» включає в себе вибір файлу, який необхідно змінити.

Прецедент «Видалити конфігураційний файл» включає в себе вибір файлу, який є неактуальним та потребує видалення.

Варіант використання «Зареєструвати сервіси» дозволяє зареєструвати всі сервіси, між якими потрібно налаштувати комунікацію, для цього слід вказати конфігураційний файл з описом доступних сервісів.

«Ввімкнути сервіс» – прецедент, що дозволяє ввімкнути сервіс та зробити його видимим для інших. Слід зазначити, що сервіс має бути зареєстрованим.

«Вимкнути сервіс» – прецедент, що дозволяє вимкнути сервіс та зробити його недоступним для інших, при цьому сервіс має бути зареєстрованим.

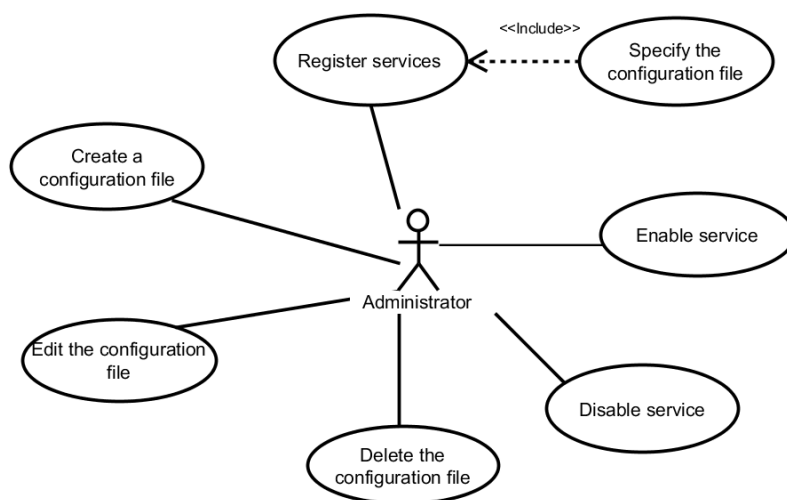


Рисунок 2 – Діаграма варіантів використання для сервісу зв'язку компонентів

Нефункціональні вимоги можна поділити на дві категорії: покращення (безпека, надійність, швидкодія та інше) та вдосконалення (масштабування, відновлюваність та ін.) властивостей системи [4].

Було виділено найбільш важливі нефункціональні вимоги.

Надійність – це властивість програмної системи функціонувати з рівнем помилок (збоїв), що не перевищує заданого рівня.

Основні вимоги до надійності сервісу:

- 1) при відмові одного із сервісів, сервіс зв'язку повинен дати коректну відповідь клієнту;
- 2) перенаправлення запиту на потрібний сервіс не повинен займати більше 1 секунди.

Здатність до перенесення – вимога для програмного додатка або системи щодо переносу на інші платформи. Головною вимогою до портативності сервісу є підтримка незалежності від операційної системи та робота із такими ж незалежними сервісами.

Вимоги до безпеки:

- 1) обмін даними між сервісами та клієнтом повинен відбуватися у вигляді передачі зашифрованих даних із ключем не меншим ніж 128 біт за технології Secure Socket Layer (SSL) через захищене HTTP з'єднання (HTTPS);

- 2) операційна система на сервісах повинна забезпечити ідентифікацію та аутентифікацію адміністратора операційної системи при його локальних запитах на доступ;

- 3) всі паролі повинні передаватися в зашифрованому вигляді.

Масштабованість – можливість збільшувати продуктивність системи пропорційно до виділених ресурсів, буває горизонтальною і вертикальною. Горизонтальне масштабування – це збільшення кількості серверів, що взаємодіють один з одним в прозорому режимі для розподілу загального завантаження системи. Вертикальне масштабування досягається за рахунок збільшення потужності окремого сервера або заміною апаратного забезпечення на більш швидкодіюче. Таким чином, система (сервіс) має можливість горизонтального та вертикального масштабування.

Моделювання послідовності виконання. Діаграма послідовності відображає взаємодії об'єктів, впорядкованих у часі. Це діаграма, на якій для деякого набору об'єктів на єдиній тимчасовій осі показані життєвий цикл і взаємодія модулів системи.

Основними елементами діаграми послідовності є позначення об'єктів, вертикальних «ліній життя», що відображають плин часу, прямокутників, що відображають діяльність об'єкта або виконання ним певної функції і стрілки, що показують обмін сигналами або повідомленнями між об'єктами. На рисунку 3 зображено діаграму послідовності для проходження тесту.

На діаграмі можна спостерігати взаємодію користувача з браузером, а також браузера з сервісом комунікацій, який вирішує куди перенаправити запит, та іншими сервісами. В сервісі комунікації відбувається

перевірка доступності сервісу для обробки запиту та перенаправлення запиту [5].

Вибір архітектури програмного забезпечення.

Зважаючи на вимоги, які були описані вище, розглянемо такі підходи до розробки архітектури, як еталонна тривірнева архітектура та мікросервісна архітектура.

Тривірнева архітектура – архітектурна модель програмного комплексу, що допускає наявність в ньому трьох компонентів: клієнта, сервера додатків (до якого підключено клієнтський додаток) і сервера баз даних (з яким працює сервер додатків).

Клієнт – це інтерфейсний компонент комплексу, що надається кінцевому користувачеві. Цей рівень не повинен мати прямий зв'язок з базою даних, містити основну бізнес-логіку і зберігати стан програми. На цей рівень зазвичай виноситься тільки найпростіша бізнес-логіка: інтерфейс авторизації, алгоритми шифрування,

перевірка вхідних даних на допустимість і відповідність формату, нескладні операції з даними.

Сервер додатків розташовується на другому рівні, в ньому зосереджена велика частина бізнес-логіки. Поза ним залишаються тільки фрагменти, експортовані на клієнта, а також елементи логіки, поміщені до бази даних. Сервери додатків проектуються таким чином, щоб додавання до них додаткових екземплярів забезпечувало горизонтальне масштабування продуктивності програмного комплексу і не вимагало внесення змін в програмний код додатка.

Сервер баз даних забезпечує зберігання даних і виноситься на окремий рівень; реалізується, як правило, засобами систем управління базами даних, підключення до цього компоненту забезпечується тільки з рівня сервера додатків [4].

На рисунку 4 показано еталонну тривірнову систему архітектури «клієнт-сервер» з виділеним сервером додатків.

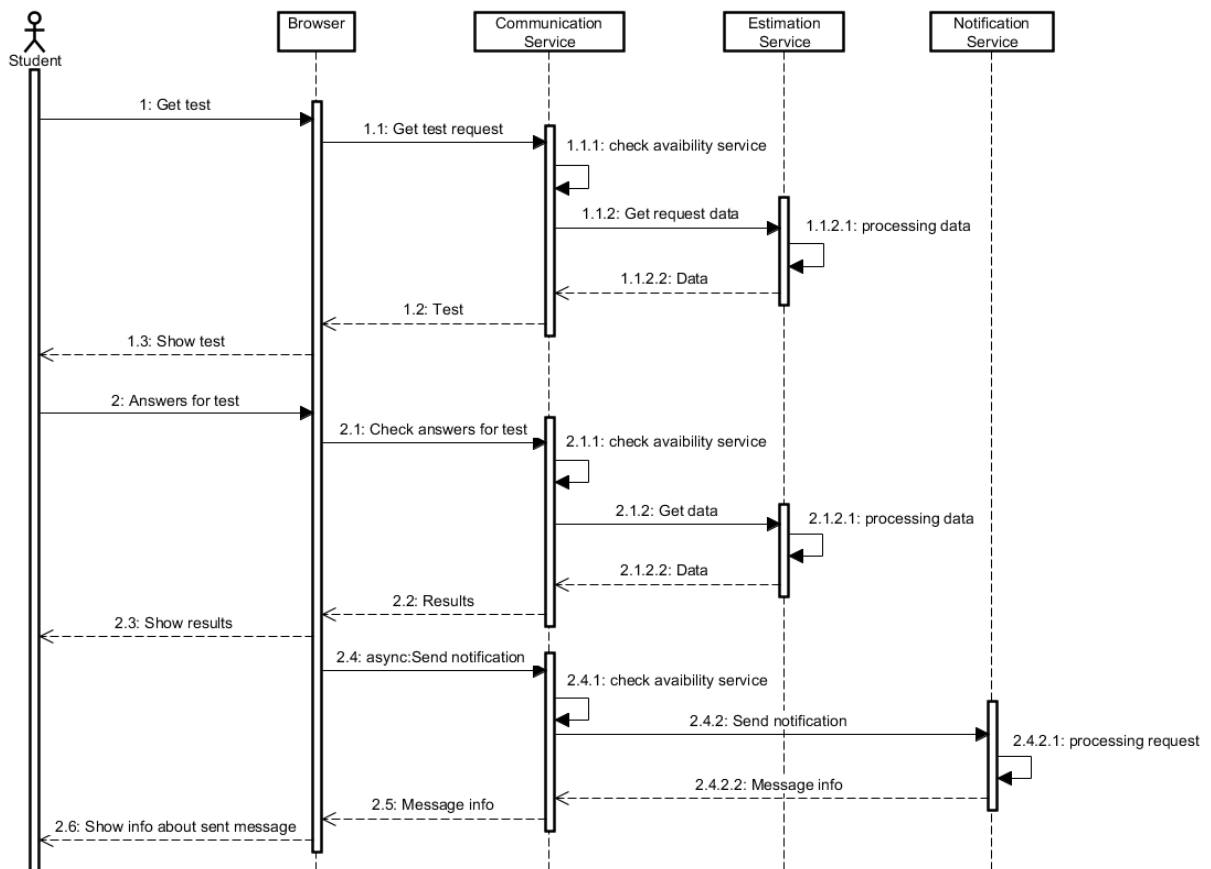


Рисунок 3 – Діаграма послідовності

Вузол Client має компонент «WebBrowser» – це програма, яка дозволяє користувачу отримати доступ до інтернету та набір бібліотек JavaScript, які полегшують роботу на стороні «Client».

Вузол WebServer – це веб-сервер системи, клієнт взаємодіє з ним за допомогою протоколу HTTPS. Розмірність відповідного зв'язку вказано як «0..*» до «1», тобто, передбачена можливість одночасної роботи декількох клієнтів з одним web-сервером.

Вузол DataBase Server – це вузол розміщення бази даних системи, яка реалізована за допомогою конкрет-

ної MySQL, зв'язок з цим вузлом забезпечується за допомогою JDBC драйвера.

Мікросервісний стиль архітектури – підхід до розробки цілісної програми як набору маленьких сервісів, кожен із яких працює у власному процесі та з'єднується з іншими за допомогою легких механізмів, таких, як HTTP. Сервіси будуються відповідно до певної задачі та можуть незалежно розгортатись автоматизованими системами.

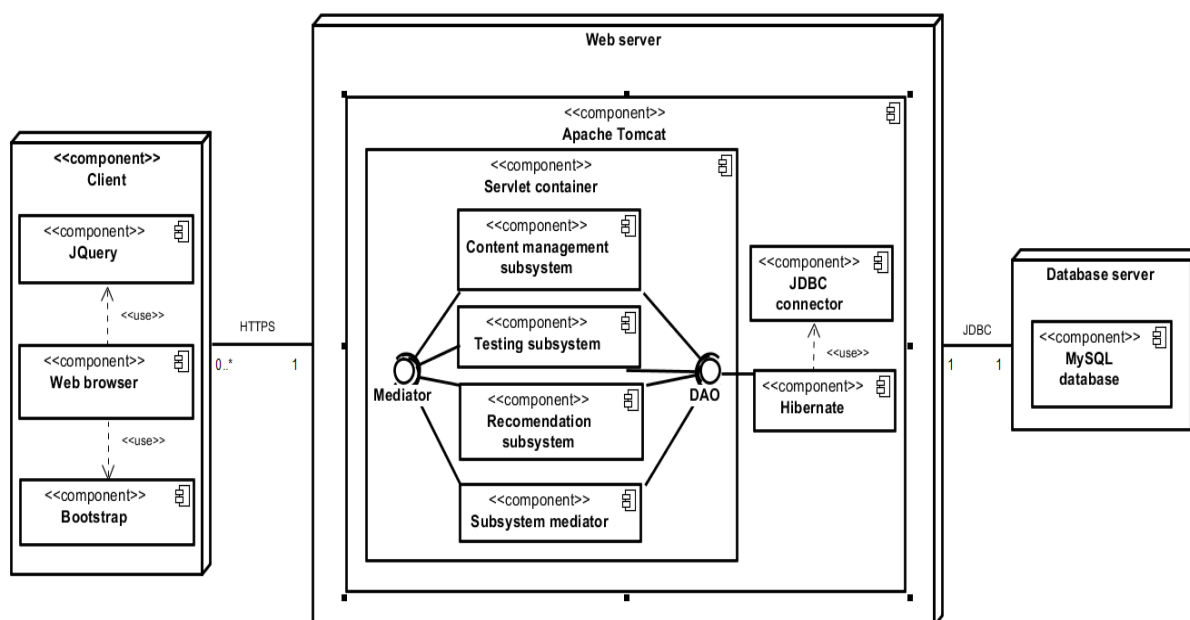


Рисунок 4 – Еталонна трирівнева архітектура системи

Уся логіка обробки запитів працює в одному процесі, що дозволяє використовувати наявні інструменти мови програмування для поділу програми на класи, функції та простори імен.

Мікросервіси можна розгортати та масштабувати незалежно один від одного. Вони мають чіткі межі між модулями та дозволяють реалізовувати окремі підсистеми різними мовами програмування. Таке розмежування допомагає управляти складністю, оскільки кожен модуль матиме публічний API, який містить лише потрібну функціональність, а все інше буде інкапсульовано та не мати значення для розробки інших сервісів, які залежать від цього [6].

На рисунку 5 зображена діаграма компонентів для системи при мікросервісному підході.

Як показано на діаграмі, всі сервіси є незалежними один від одного та взаємодіють лише на основі відкритого інтерфейсу. Також на діаграмі зображений API GateWay, який виступає своєрідним проксі сервером. Це надає зручності роботи з системою, оскільки звернення відбувається за однією адресою, а не за індивідуальною адресою кожного мікросервісу. Також використання API GateWay інкапсулює мікросервіси, що в свою чергу дозволяє змінювати їх структуру та API.

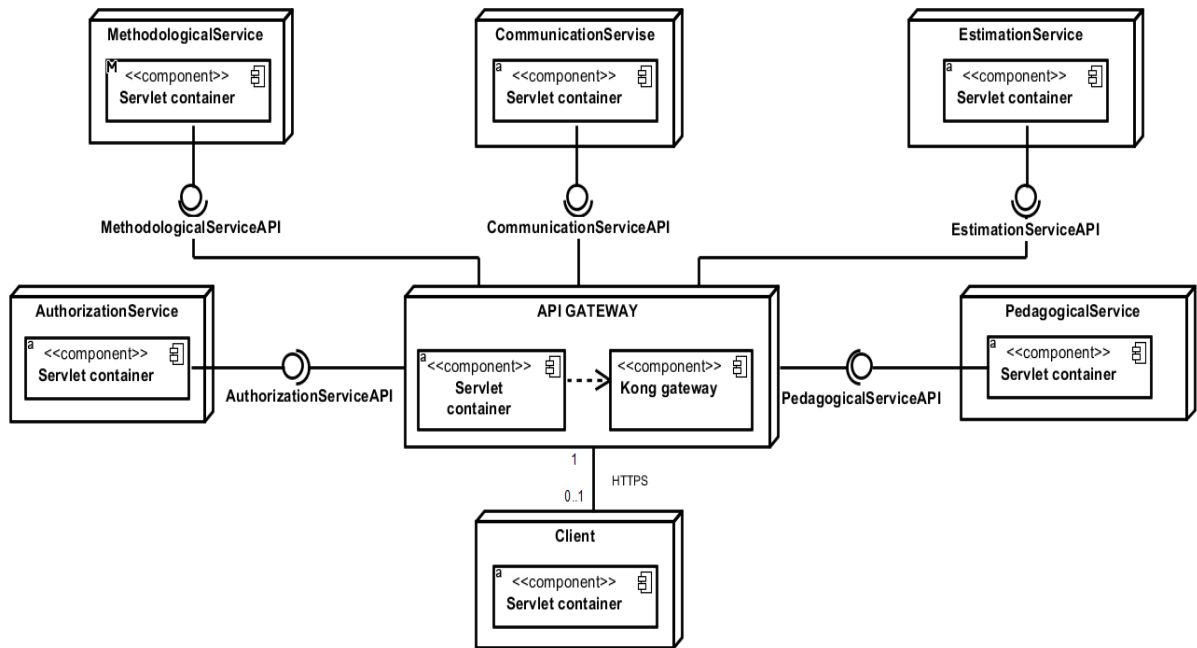


Рисунок 5 – Діаграма компонентів при мікросервісному підході

Еталонні рішення. Функціональна декомпозиція додатка є ключем до створення успішної мікросервісної архітектури. Це дозволяє досягти слабкої зв'язності та високої згуртованості (декілька менших сервісів можуть створювати один більший композитний, вищий за рівнем сервіс). Функціональна декомпозиція дає швидкість, гнучкість, масштабованість та інші переваги, але її метою є створення архітектурного рішення, яке було б стійким до змін.

Еталонним рішенням або патерном проектування називають рішення для проблем, які часто зустрічаються в області розробки програмного забезпечення. Патерни проектування не є готовими рішеннями, які можна трансформувати безпосередньо в код, а являють собою загальний опис вирішення проблеми, який можна використовувати в різних ситуаціях.

За класифікацією GoF патерни розрізняють за типами і видами. Існує кілька типів патернів проектування, кожен з яких призначений для вирішення свого кола завдань [7]:

- 1) породжувальні патерни, призначені для створення нових об'єктів в системі;
- 2) структурні патерни, які вирішують задачі компонування системи на основі класів і об'єктів;

3) патерни поведінки, призначені для розподілу обов'язків між об'єктами в системі.

Агрегатор є найбільш поширеним патерном при проектуванні мікросервісів. У найпростішій формі агрегатор буде звичайною веб-сторінкою, яка викликає декілька служб для досягнення функціональності, якої вимагає додаток. Кожен із мікросервісів надає легковисний REST, за допомогою якого веб-сторінка може отримати дані та обробити або відобразити їх відповідним чином (рис. б).

Інший варіант шаблону агрегатора – коли не треба відображати дані, а замість цього потрібно просто створити композитний мікросервіс, який буде використовуватись іншими сервісами. В цьому випадку агрегатор повинен зібрати дані від інших мікросервісів, застосувати бізнес-логіку до них та в подальшому опублікувати їх як кінцеву точку REST. Ці дані потім можуть бути використані іншими сервісами, які їх потребують [8].

Посередник – поведінковий шаблон проектування, що забезпечує взаємодію множини об'єктів, формуючи при цьому слабку зв'язаність і позбавляючи об'єкти необхідності явно посилатися один на одного.

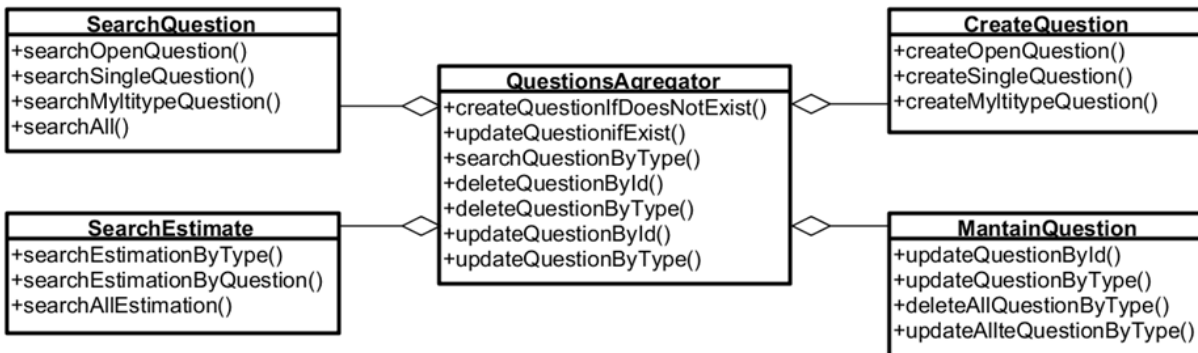


Рисунок 6 – Використання агрегатора мікросервісів

Оскільки система, що проектується, складається з трьох підсистем (підсистеми управління контентами, підсистеми тестування, підсистеми аналізу результатів та рекомендацій), виникає необхідність організації взаємодії між ними [8].

Патерн-посередник використовується для того, щоб позбутися необхідності явної взаємодії підсистем. Натомість функцію взаємодії бере на себе посередник (рис. 7).

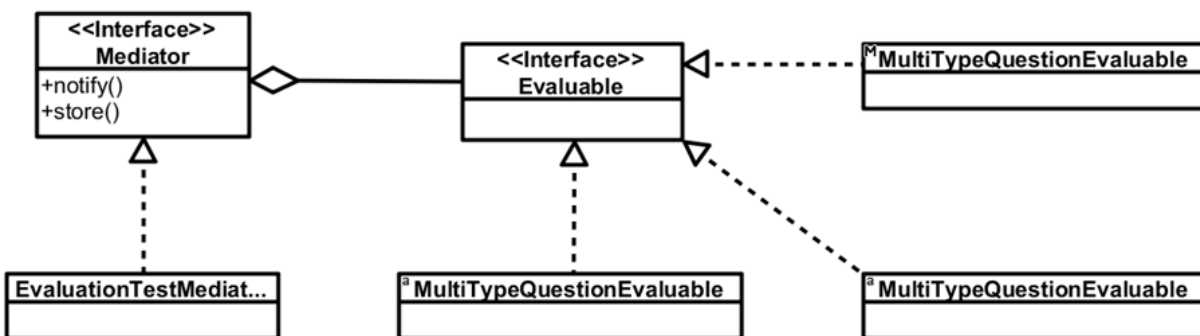


Рисунок 7 – Використання посередника для взаємодії між підсистемами

Превагами застосування цього патерну є зменшення числа породжуваних підкласів, оскільки посередник інкапсулює поведінку, яку необхідно було б розподіляти між декількома об'єктами. Для зміни поведінки необхідно лише породжувати посередника, а «колеги» можна використовувати повторно [7, 8].

Окрім того, до переваг даного рішення відноситься усунення зв'язку між колегами. Це дає можливість змінювати класи колеги та посередника незалежно один від одного.

Мінусом даного підходу є централізація управління. Посередник переносить всю складність взаємодії в себе, що робить його монолітним та складним для розуміння [7].

Ланцюговий шаблон проектування мікросервісів надає єдину відповідь на певний запит. У цьому випадку

запит від клієнта отримано сервісом А, який в свою чергу з'єднаний із сервісом В, який може мати зв'язок з сервісом С. Всі сервіси, ймовірно, використовують синхронний обмін повідомленнями по HTTPS.

Слід враховувати те, що клієнт буде заблокований, поки не закінчиться повний ланцюжок запитів / відповідей між сервісами. Класи, які є складовою ланцюжка, містять у собі метод обробки даних, а також поле, в якому міститься посилання на наступного обробника в ланцюжку [7].

Асинхронний обмін повідомленнями. Водночас, коли використання REST є дуже поширеним та добре зрозумілим, воно накладає обмеження на те, щоб бути синхронним та неблокуючим. Асинхронність може бути досягнута, але тільки специфічним прикладним шляхом. У деяких випадках мікросервісна архітектура мо-

жуть використовувати черги повідомлень замість запитів / відповідей у REST.

У цьому шаблоні проектування сервіс А може викликати сервіс С синхронно, який в свою чергу зв'язаний з сервісами В і Б в асинхронному режимі з використанням загальної черги повідомлень. Зв'язок А та С може бути асинхронним, можливо, з використанням WebSocket, для досягнення бажаної масштабованості. Поєднання запитів/відповідей у REST із чергами повідомлень може бути використано для реалізації задач бізнес-логіки [7].

ВИЗНАЧЕННЯ ТЕХНОЛОГІЙ ДЛЯ РЕАЛІЗАЦІЇ ТА ФУНКЦІОНУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Аналізуючи наведені вище діаграми та специфіки програмного рішення, яке моделюється, було обрано мову програмування та платформу виконання Java, а також такий стек технологій:

1. JDK 1.8 – комплект розробника додатків мовою Java, який включає в себе компілятор Java, стандартні бібліотеки класів Java, приклади, документацію, різноманітні утиліти і виконавчу систему Java (JRE). До складу JDK не входить інтегроване середовище розробки на Java, тому розробник, що використовує тільки JDK, повинен використовувати текстовий редактор і компілювати та виконувати свої програми через утиліти командного рядка.

2. IntelliJ Idea 15.0 Community Edition – інтегроване середовище розробки для різних мов програмування (Java, Python, Scala, PHP та ін.) від компанії JetBrains. Система поставляється у вигляді урізаної за функціоналом, безкоштовної версії «Community Edition» і повнофункціональної комерційної версії «Ultimate Edition», для якої активні розробники відкритих проектів мають можливість отримати безкоштовну ліцензію. Сирцеві тексти Community-версії поширюються рамках ліцензії Apache 2.0 [9].

3. Spring Framework – програмний каркас (фреймворк) з відкритим кодом та контейнером з підтримкою інверсії управління для платформи Java. Основні особливості Spring Framework можуть бути використані будь-яким застосунком Java, але є розширення для створення веб-додатків на платформі Java Enterprise Edition не нав'язує будь-яку конкретну модель програмування.

4. Spring Cloud Config – це сховище конфігурацій, що горизонтально масштабується для розподіленої системи. Як джерело даних на даний момент підтримуються Git, Subversion і прості файли, що зберігаються локально. За замовчуванням Spring Cloud Config віддасть файли, відповідного імені запитувача Spring-додатка (але можна забирати властивості залежно від конкретного Spring-профілю і з певної гілки системи контролю версій).

5. Kong – масштабований API шлюз. Kong проходить перед будь-яким RESTful API і поширюється через плагін, який забезпечує додаткові функціональні можливості і послуги за межами основної платформи.

6. jQuery – JavaScript-бібліотека з відкритим вихідним кодом.

Для розробки програмного забезпечення на Java Enterprise Edition краще за все використовувати IntelliJ Idea від NetBeans.

IntelliJ IDEA – комерційне інтегроване середовище розробки для різних мов програмування (Java, Python, Scala та ін.) від компанії NetBeans. Система поставляється у вигляді урізаної за функціональністю, безкоштовної версії «Community Edition» і повнофункціональної комерційної версії «Ultimate Edition», для якої активні розробники відкритих проектів мають можливість отримати безкоштовну ліцензію. Вихідний код Community-версії поширюються в рамках ліцензії Apache 2.0. Бінарні складові підготовлені для Linux, MacOS і Windows.

Основними можливостями розробки Community-версії середовища IntelliJ IDEA, є підтримка інструментів для проведення тестування TestNG і JUnit, системи контролю версій Subversion, Mercurial і Git, засобів складання Maven і Ant, мов програмування Java, Scala, Clojure, Groovy і Dart. Підтримується розробка додатків для мобільної платформи Android.

Є доступними засоби інтеграції з системами відстеження помилок JIRA, Trac, Redmine, Pivotal, Tracker, GitHub, YouTrack, LightHouse.

Тестування програмного забезпечення – це процес технічного дослідження, призначений для виявлення інформації про якість продукту відносно контексту, в якому він має виконуватись. Техніка тестування також включає як процес пошуку помилок або інших дефектів, так і випробування програмних складових з метою оцінки, при цьому може оцінюватись:

- 1) відповідність вимогам, якими керувалися проєктувальники та розробники;
- 2) правильна відповідь для усіх можливих вхідних даних;
- 3) виконання функцій за прийнятний час;
- 4) практичність;
- 5) сумісність з програмним забезпеченням та операційними системами;
- 6) відповідність задачам замовника [8].

Оскільки число можливих тестів навіть для нескладних програмних компонент практично нескінченне, тому стратегія тестування полягає в тому, щоб провести всі можливі тести з урахуванням наявного часу та ресурсів. Як результат, ПЗ тестується стандартним виконанням програми з метою виявлення помилок або інших дефектів.

Тестування ПЗ може надавати об'єктивну, незалежну інформацію про якість ПЗ, ризики відмови як для користувачів, так і для замовників.

Тестування може проводитись, як тільки створено код. Процес розробки зазвичай передбачає час для тестування. Наприклад, при поетапному процесі більшість тестів відбувається після визначення системних вимог і тоді вони реалізуються в тестових програмах. На противагу цьому, відповідно до вимог гнучкої розробки ПЗ, програмування і тестування часто відбувається одночасно. Як висновок, можна зауважити, що тестування ПЗ – техніка контролю якості, що перевіряє відповідність між реальною і очікуваною поведінкою програми завдяки кінцевому набору тестів, які обираються певним чином.

Перш за все, тестування поділяється на дві категорії: автоматичне і ручне.

У цій роботі використовувались обидва види тестування, однак ручне тестування не було задокументовано. Як автоматичне було використано модульне та інтеграційне тестування.

Модульне тестування або **юніт-тестування** – процес в програмуванні, який дозволяє перевірити коректність певних модулів усієї програми. Ідея полягає в тому, щоб писати тести для кожної нетривіальної функції чи методу. Це дозволяє достатньо швидко перевірити, чи не привели певні зміни до регресії, тобто до появи помилок у вже протестованих місцях програмного рішення, а також прискорює пошук та усунення таких помилок. Таким чином, юніт-тестування являє собою перший бастіон у боротьбі за якість програмного забезпечення [9, 10].

Модуль в контексті модульного тестування являє собою мінімальну смислову одиницю вихідного коду, придатну для тестування (функція, процедура, метод, в окремих випадках – клас). Модульний тест перевіряє працездатність модуля в умовах ізоляції від інших модулів програми. Виключивши вплив інших модулів, модуль перевіряється в стерильних умовах, якщо результат виконання модульного тесту не відповідає очікуваному, то помилка криється в логіці модуля, а не в зовнішніх чинниках. Приклад таких тестів для програмного додатку зображений на рис. 8.

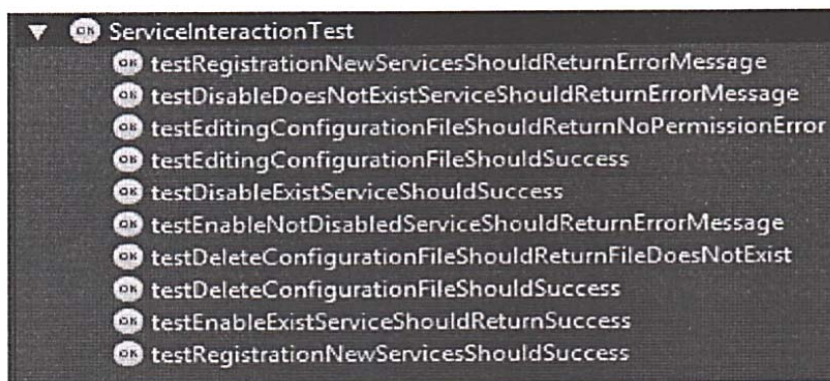


Рисунок 8 – Приклад модульних тестів

Інтеграційне тестування – це фаза тестування ПЗ, під час якої окремі модулі системи комбінуються та тестуються разом, у взаємодії. Інтеграційне тестування виконується після модульного тестування та перед

верифікацією та валідацією ПЗ. Якщо розглядати цей процес як систему, то на вхід їй подаються модулі, які вже пройшли модульне тестування, потім модулі групуються в більші частини, виконуються тести передбачені планом, а на виході системи – інтегрована система, що готова до системного тестування або тестування сценаріїв.

Метою інтеграційного тестування є верифікація вимог з функціональності, продуктивності, надійності до основних компонентів програми. Ці компоненти, тобто групи модулів, тестуються методом «чорної скриньки», успішні та неуспішні тест-кейси симулюються відповідними вхідними параметрами. В процесі інтеграційного тестування тестуються симульоване використання спільних даних та комунікація між процесами. Тест-кейси перевіряють, чи коректно взаємодіють всі компоненти, наприклад, через виклик процедури чи акти-

візацію процесу. Дуже важливо те, що до інтеграційного тестування приступають тільки після модульного. Це дозволяє реалізувати стратегію «будівельних блоків», коли до верифікованої системи інтегруються верифіковані модулі.

В цілому, інтеграційне тестування перевіряє інтерфейси між компонентами, взаємодію з різними частинами системи, такими, як операційна система, файлова система та апаратне забезпечення, інтерфейси між системами. Інтеграційне тестування може складатися з одного чи більше рівнів та може виконуватися на тестових об'єктах різного розміру.

Приклади прецедентів (Test cases) для інтеграційного тестування подані в додатку.

Приклад автоматичних інтеграційних тестів відносно розроблюваного веб-додатка зображений на рис. 9.

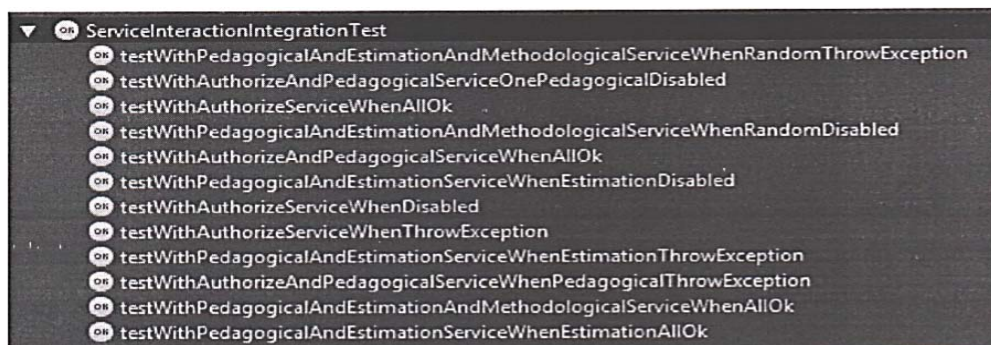


Рисунок 9 – Приклад автоматичних інтеграційних тестів

В ході інтеграційного тестування всі тести пройшли вдало. Помилку у взаємодії підсистем не виявлено.

Опис вказівок для користувача. Розроблюване програмне рішення створене для підприємств, які хочуть поліпшити знання свої працівників. Це програмне забезпечення являє собою веб-додаток. Для того щоб запустити веб-додаток на персональному комп'ютері, потрібно мати набір засобів для розробника, таких, як Apache TomCat контейнер сервлетів та СКБД MySQL, але якщо додаток буде розміщений в інтернеті, то для того, щоб почати використовувати його, потрібно просто перейти за посиланням.

Перш за все, для аутентифікації користувача необхідно зареєструватися у системі. Форма для заповнення зображена на рис. 10.

Рисунок 10 – Форма реєстрації

Для того щоб зареєструватися, користувач має ввести валідний логін, e-mail, пароль, вказати роль в системі, а також ввести капчу.

Після успішної реєстрації можна увійти в систему. На рис. 11 зображена форма авторизації.

Для авторизації необхідно ввести логін та пароль, а також капчу. Користувач може обрати опції «Remember me».

Після успішної авторизації користувача буде перенаправлено до дошки приладів. Розглянемо можливості системи під різними ролями.

Рисунок 11 – Форма авторизації

Спочатку розглянемо можливості системи для ролі «Викладача». На рис. 12 зображено меню користувача.

Головними можливостями в системі є створення курсу та редагування курсів, перегляд студентів, а також комунікація з ними.

Рисунок 12 – Меню користувача

На рисунку 13 зображена форма створення курсу.

Рисунок 13 – Форма створення курсу

Для того щоб створити курс, необхідно вказати його назву, вибрати до якої категорії він належить, а також задати його опис. Після цього він з'явиться в меню з курсами та буде доступний для редагування (рис. 14).

Рисунок 14 – Меню редагування курсу

До основних можливостей роботи з курсом належать перегляд студентів, які його проходять, додавання лекції, а також редагування інформації про курс.

На рисунку 15 зображено створення нової лекції до курсу.

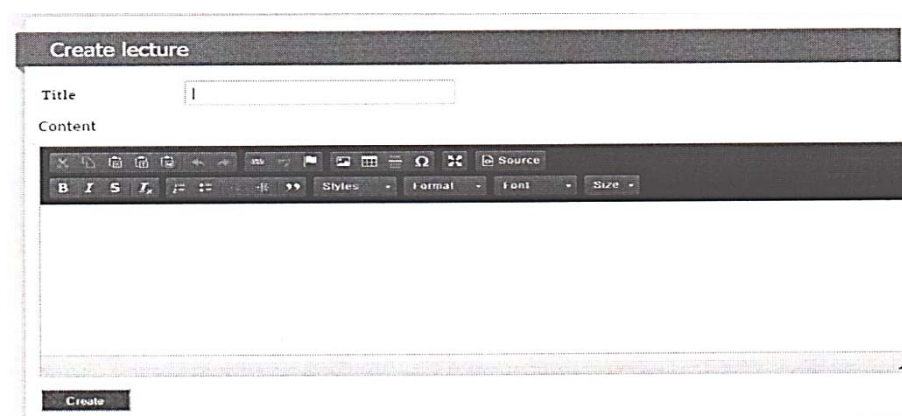


Рисунок 15 – Створення нової лекції

Після створення лекцію можна знайти в меню курсу. Для кожної лекції можна задати свій набір тестів (рис. 16).

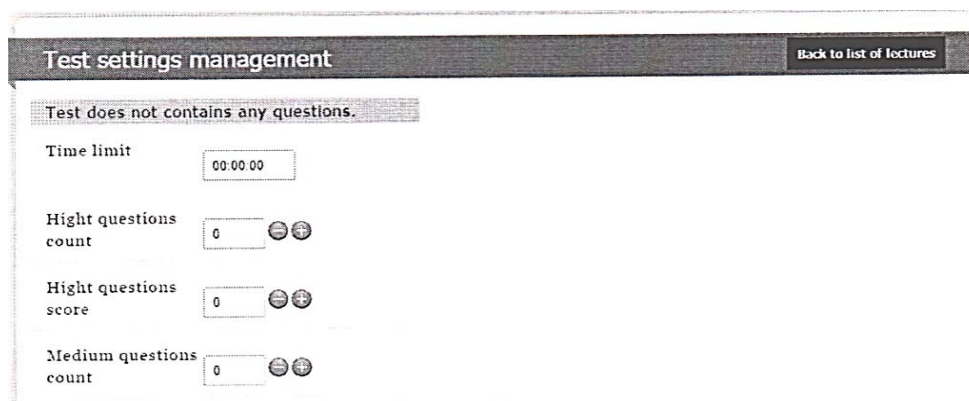


Рисунок 16 – Створення нового тесту

Далі розглянемо можливості системи з роллю «Студент». На дошці приладів студента відображається головне меню, а також останні додані курси (рис. 17).

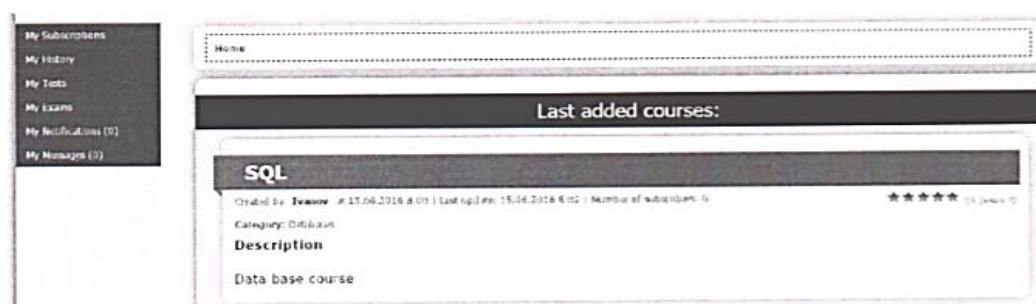


Рисунок 17 – Дошка приладів «Студента»

Головними можливостями системи у ролі «Студент» є перегляд доступних курсів, перегляд підписок, тестів, екзаменів, а також повідомлень.

Після переходу на сторінку з курсом користувач може лише переглянути основну інформацію. Для відкриття інших опцій необхідно підписатися.

Після оформлення підписки студент може ознайомитися з лекцією, а також пройти тест на засвоєння матеріалу.

На рисунку 18 зображено меню тестування.

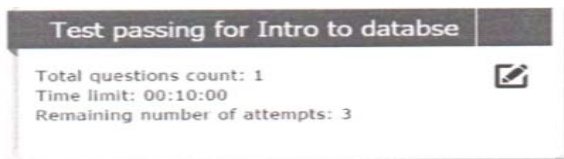


Рисунок 18 – Меню тестування

Після проходження тесту його результати можна переглянути перейшовши в «My Test» на дошці приладів. На рисунку 19 зображені результати проходження тесту.



Рисунок 19 – Результати проходження тесту

До інформації про тест входить назва лекції, дата та час проходження, результат у відсотках, а також статус перевірки.

ВИСНОВКИ

В результаті дослідження проведено аналіз проблеми розробки вимог та створення програмного забезпечення системи електронного навчання. Були розроблені функціональні та нефункціональні вимоги,

на основі яких було створено веб-додаток на основі мікросервісів. Працездатність розробленого програмного забезпечення підтверджено за допомогою автоматичного (модульного та інтеграційного) і ручного тестування. Запропоноване програмно-архітектурне рішення може бути використане в системі дистанційного навчання.

ЛІТЕРАТУРА:

1. <http://www.elitarium.ru/distancionnoe-obrazovanie-obuchenie-sistema-organizacija>, 05.05.2018.
2. Afanasjev Yu.I. Sovremennie problem nauki i obrazovaniya, 2015 – S. 1–45.
3. Advanced Distributed Learning, Intelligent Tutoring System, and SCORM 2. – 2013. – P. 76–80.
4. Karl I. Vigers. Razrabotka trebovanij k programmnomu obespecheniyu. – 2009 – S. 243–260.
5. Gang of Four Patterns // <http://c2.com/cgi/wiki?GangoffFour/>, 20.04.2018.
6. Jess Chedvik. MVC ta rozrabotka realnih veb-prilozhenij s pomoshchju ASP.NET MVC. – М.: Wiljams, 2013 – 432 s.
7. Lucas Krause. Microservices: Patterns and Applications, - 2015.
8. Офіційна сторінка JetBrains // <https://www.jetbrains.com/idea/features/>, 01.05.2018.
9. Sinicin S.V, Nalutin N.Yu. Verifikacija programmного obespechenija. – М.: VINOM, 2008. – 368 s.
10. Lavrishcheva E.M. Metody I sredstva snzhenerii programmного obespecheniya: Uchebnik / E.M. Lavrishcheva, V.A. Petruhin. – М.: MFTS (GU), 2006. – 304 p.

Рецензент: д.т.н., проф. Коваленко В.Ф.,
Херсонський національний технічний університет

ПІДХІД ДО ЗБОРУ ІНФОРМАЦІЇ ЩОДО ЕКОЛОГІЧНОЇ ОБСТАНОВКИ ПРИ ВИНИКНЕННІ НАДЗВИЧАЙНИХ СИТУАЦІЙ ТЕХНОГЕННОГО ХАРАКТЕРУ

УДК 004.02:519.8

ТКАЧЕНКО Василь Володимирович

Начальник науково-дослідного відділу в НДЦ ЗС України «Державний океанаріум»

Наукові інтереси: екологічна безпека, інформаційні технології, системи підтримки прийняття рішень, автоматизація збору та обробки екологічних даних.

e-mail: rhb78@ukr.net

ЧЕРЕДНІЧЕНКО Ольга Юрїївна

доцент, кандидат технічних наук, доцент кафедри програмної інженерії та інформаційних технологій управління «Національного технічного університету «ХПІ»

Наукові інтереси: інтелектуальний аналіз даних, інформаційні технології, багатоагентні системи, веб-моніторинг.

e-mail: olha.cherednichenko@gmail.com

ВОВК Марина Анатоліївна

кандидат економічних наук, доцент кафедри програмної інженерії та інформаційних технологій управління «Національного технічного університету «ХПІ»

Наукові інтереси: системи підтримки прийняття рішень, комплексна оцінка багатоозначених об'єктів.

e-mail: marihavovk@gmail.com

ЕРШОВА Світлана Іванівна

старший викладач кафедри програмної інженерії та інформаційних технологій управління Національного технічного університету «ХПІ»

Наукові інтереси: інформаційні технології, системи підтримки прийняття рішень, математичне моделювання.

e-mail: svetlana.ershova.2016@gmail.com.

ВСТУП

В сучасному суспільстві проблеми антропогенного впливу на екологію зовнішнього середовища без сумніву є найбільш актуальними та гострими. Протягом декількох десятиліть демографічна ситуація на території України значно погіршила показники здоров'я громадян, особливо на тлі соціально-економічних зрушень. Починаючи з 2015 року, з прийняттям Стратегії національної безпеки України, затвердженої указом Президента України № 287/2015 від 6 травня 2015 року, нової редакції Воєнної доктрини, затвердженої указом Президента України № 555/2015 від 2 вересня 2015 року, а також в рамках Угоди про асоціацію між Україною та Європейським союзом та задекларованими

намірами до Європейської та Євроатлантичної інтеграції Україна взяла на себе зобов'язання по переходу до міжнародних стандартів, в тому числі і в сфері екологічної безпеки. Потужною силою в розвитку сучасного суспільства є інтенсивне глобальне поширення інформаційно-комунікативних технологій, які допомагають збирати, зберігати, аналізувати та розповсюджувати інформацію. Сучасне управління наслідками промислових аварій, стихійних лих та усіх видів надзвичайних ситуацій базується на інтенсивному використанні інформаційних технологій [1]. За допомогою інформаційних технологій визначаються управлінські дії та пов'язані з цим ресурси, які розгортаються там, де це необхідно. Проблема організації збору, обробки та аналізу даних є



однією з найбільш актуальних і невирішених проблем сьогодні [2-4]. Використання систем підтримки прийняття рішень (СППР) робить процес більш ефективним [5].

Будь-який техногенний вплив спричиняє певні зміни параметрів навколишнього природного середовища. Заходи, що спрямовані на виконання поставлених цілей, повинні спиратися на детальний аналіз існуючої екологічної обстановки. При реалізації екологічної політики варто використовувати інструменти, застосування яких забезпечить можливість диференціювати досліджувані об'єкти залежно від екологічного стану, що сформувався. На практиці часто відбувається комплексний моніторинг стану навколишнього середовища для визначення ситуації та прийняття правильних рішень [6-8]. У цьому випадку необхідно одночасно контролювати всі показники екологічного стану. Основні проблеми зі збором, обробкою та обміном інформацією про стан навколишнього середовища обумовлені складністю комбінованого використання різномірних даних вимірювань, потоків оперативної інформації моніторингу та також стосуються багатьох невирішених питань інформаційної взаємодії [9-11]. Різні пристрої та спеціальне обладнання для збору та обліку даних використовується службами реагування для оцінки викидів та виявлення найбільш небезпечних речовин [12-14]. Зібрана інформація повинна бути швидко та безперебійно передана командному центру для координації діяльності та забезпечення зв'язку між географічно відокремленими локаціями.

В роботі [15], присвяченій питанням обробки даних моніторингу довкілля, запропоновано використовувати хмарну платформу як єдину систему збору, зберігання, аналізу, обробки та обміну даними моніторингу навколишнього середовища. Основною перевагою такої платформи є можливість використання обчислювальних ресурсів за вимогою. Ісаєв І.В. [16] також звертає увагу на питання зберігання хмарних даних для забезпечення прийняття рішень з повними та структурованими даними при створенні СППР для моніторингу стану навколишнього середовища, але основна увага приділяється процесам формування показників для визначення рівня еколого-економічного стану території, не враховуючи характеристики вхідної інформації.

При управлінні збором та обробкою моніторингових даних щодо визначення параметрів екологічної ситуації зазвичай необхідно забезпечити в реальному масштабі часу розрахунок і оптимізацію режиму моніторингу, який гарантовано буде забезпечувати центри прийняття рішень актуальною та достовірною інформацією. Як показав проведений аналіз, методи, які зазвичай застосовують, мало підходять для вирішення завдань такого класу через можливість появи довільних неконтрольованих помилок в кінцевих результатах при наявності похибок у вихідних даних. Це обумовлено невизначеністю інформації щодо надзвичайної ситуації, особливо це важливо в умовах транскордонних техногенних аварій, коли встановлення сенсорів для вимірювання екологічних параметрів неможливе, як і організація тотального моніторингу території. Тому при управлінні такими об'єктами доводиться орієнтуватися на найнесприятливіше поєднання чинників невизначеності і використовувати стратегію гарантованого результату. До теперішнього часу математичне моделювання дискретних слабо структурованих процесів і систем, для яких характерні множина критеріїв, стохастичність, інтервальність або нечіткість значень вихідних даних, все ще знаходиться у зародковому стані.

ПОСТАНОВКА ЗАДАЧІ

Необхідною частиною будь-якого проекту екологічного моніторингу є використання обладнання. Деякі пристрої, такі як лічильники та комплекти для польових, призначені для відбору проб. Інші, наприклад, аналізатори, реєстратори даних та буйки даних, призначені для довгострокових програм моніторингу. Хоча буї, зонди та автоматизовані пробовідбірники, як правило, спеціфічні для проектів перевірки якості води, датчики та реєстратори даних можуть використовуватися для моніторингу практично будь-яких параметрів навколишнього середовища. Інтернет-центри даних, коли вони використовуються разом із телеметрією та реєстратором даних, надають в режимі реального часу дані та доступ до проекту з будь-якого місця в будь-який час. Обладнання для вимірювання підбирається виходячи з вимог до точності, бюджету та того, чи потрібен віддалений моніторинг.

Реагуючи на виникнення надзвичайних ситуацій, де існує потенційна небезпека для навколишнього середо-

вища, дуже важливо, щоб служба порятунку та її працівники спочатку визначилися, які небезпечні фактори потрібно виміряти. Особливо важливою є задача оперативного визначення меж небезпечної зони та локалізація зони найвищої концентрації небезпечних речовин. Якість прийняття рішень, а також масштаби наслідків надзвичайної ситуації, суттєво залежать від часу збору екологічних даних. Сучасні прилади та обладнання дозволяють вимірювати та передавати моніторингові дані до центру обробки. Слабким місцем залишається розміщення та оперативне пересування вимірювального обладнання.

Формально проблему, пов'язану з неефективною організацією процесу моніторингу екологічної ситуації, можна описати таким чином. Є процес збору первинних даних, яким можна керувати шляхом зміни його параметрів. Існує необхідність підвищити ефективність моніторингу екологічної ситуації, змінюючи керуючі параметри щодо збору первинних даних. При цьому технологічний процес повинен задовольняти виділеним в результаті проведеного аналізу особливостям.

1. Управління процесом моніторингу екологічної ситуації здійснюється зміною його параметрів (зокрема умов вимірювання параметрів).

2. Регулювання параметрів можливе в досить широких межах.

3. Оцінку кінцевого результату управління можна отримати, вимірявши відповідні характеристики тільки після закінчення циклу моніторингу.

4. Процес не має в явному вигляді функціональної залежності між його параметрами і вихідним результатом.

5. Побудова математичної моделі процесу моніторингу недоцільна.

Враховуючи наявність декількох критеріїв оцінки процесу моніторингу екологічної ситуації (час збору даних, час пошуку джерела небезпеки, час локалізації зон екологічної небезпеки тощо) доцільно розглянути багатокритеріальну постановку задачі ефективного керування технологічним процесом. До основних проблем такої постановки задачі управління процесом збору даних слід віднести наступні: 1) неузгодженість; 2) неформалізований, якісний характер локальних критеріїв; 3) необхідність формалізації правил порів-

нянь рішень щодо отримання найкращого; 4) складність методів пошуку компромісного рішення.

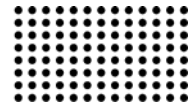
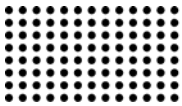
Процес збору екологічних даних можна представити в дискретно-статичному вигляді. Тоді з таким процесом можна взаємодіяти як з «чорним ящиком» протягом одного циклу. З метою зменшення складності задачі, а також вирішення проблеми подолання багатокритеріальності, в даній роботі запропоновано використання наступного підходу. Задачу оптимізації збору даних екологічного моніторингу в умовах виникнення надзвичайних ситуацій можна розглядати, як задачу пошуку у просторі критеріїв оцінки рівня екологічної безпеки.

В роботі пропонується для визначення послідовності точок вимірювання екологічних параметрів застосування методів деформованих конфігурацій [17], які використовують правильні симплекси в якості основної конфігурації. Управління процесом пошуку в таких методах здійснюється за рахунок вибору локально оптимального напрямку зсуву центру симплекса і вибору величини кроку, що забезпечує оптимізацію функції в центрі симплекса. Це надає можливість визначити напрямок руху пересувних вимірювальних приладів, що дозволяє скоротити час локалізації місць з найвищим рівнем небезпечної речовини за рахунок збіжності алгоритму за кінцеве число кроків. Отримані оперативні дані використовуються для первинної оцінки рівня екологічної безпеки та класифікації надзвичайної ситуації.

ВИРІШЕННЯ ЗАДАЧІ

Порівняльний аналіз відомих алгоритмів пошукової оптимізації показав переваги запропонованих академіком Риковим О. С. [17] методів деформованих конфігурацій, як найбільш придатного підходу для вирішення поставленої задачі. Конфігурація – це множина вершин, що складається з k точок, число яких більше розмірності n простору незалежних змінних і які лежать одночасно у підпросторі розмірності, меншою за n . Для пошуку максимального значення функції на кожній ітерації використовується певна конфігурація.

Для розв'язку задачі безумовної мінімізації скалярної функції запропоновано методи деформованих конфігурацій, які використовують правильні симплекси в якості основної конфігурації. Управління процесом



пошуку в таких методах здійснюється за рахунок вибору локально оптимального напрямку зсуву центру симплекса і вибору величини кроку, що забезпечує зменшення значення функції, яка оптимізується в центрі симплекса. Величина кроку змінюється шляхом зміни розміру симплексу зі збереженням правильної форми симплекса.

Використані симплексні методи для розв'язання деякої задачі двовимірної безумовної мінімізації. Область допустимих значень в цьому випадку збігається з усім простором R^2 і є площиною.

Для розв'язання задачі мінімізації функції мети $f(x)$, де $x \in R^2$, застосовано симплексний метод з відображенням однієї вершини на кожному кроці і правильним симплексом. У даному випадку правильним симплексом є рівносторонній трикутник. При русі до екстремуму на кожному кроці оптимізації відображається та вершина трикутника, в якій функція $f(x)$ приймає максимальне значення. В результаті послідовних відображень утворюється ланцюжок трикутників $S_1, S_2, \dots, S_N, \dots$ причому центри цих трикутників x^1, x^2, \dots, x^N здійснюють коливальні рухи вздовж градієнтного напрямку (рис. 1)

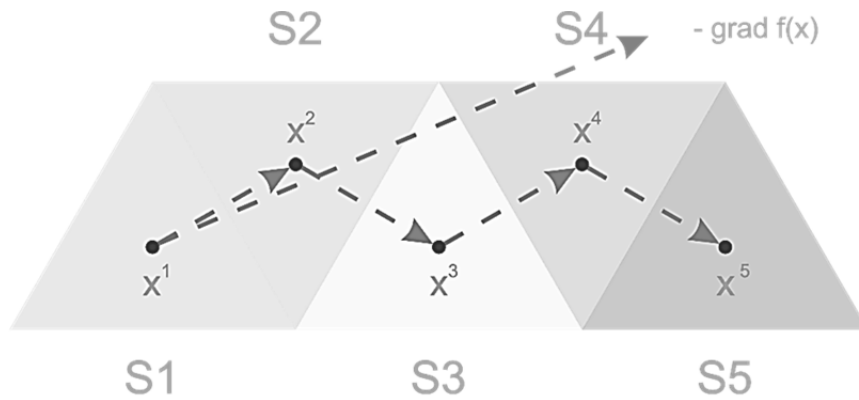


Рисунок 1. – Приклад процедури з відображенням однієї вершини на кожному кроці

У якості критерію локальної оптимальності нами використано критерій, за яким відображаються ті вершини, в яких значення функції, що оптимізується є біль-

шим за значення функції в центрі симплекса. На рис. 2 наведено приклад застосування такого критерію.

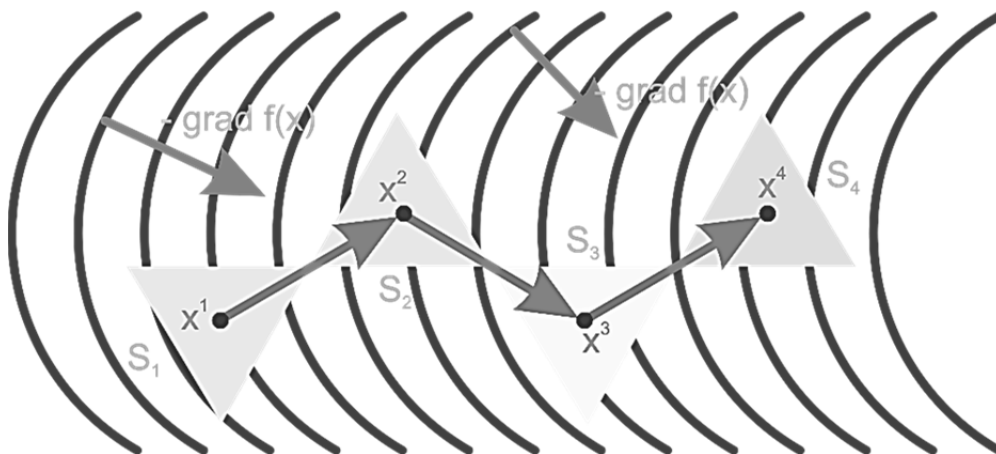


Рисунок 2. - Приклад процедури з відображенням декількох вершин, значення функції в яких більше, ніж значення функції в центрі симплекса

Було розв'язано ряд проблем: по організації можливих напрямків зміщення центру симплекса і вибору з них оптимального в деякому розумінні напрямку, а також за вибором розміру симплекса.

Під відображенням $m+1$ ($m = 1, \dots, n; l = 0, \dots, n-m$) вершин симплекса S_N розуміємо такий паралельний перенос $m+1$ його вершин вздовж напрямку від геометричного центру m відображуваних вершин симплексу S_N , до центру невідображуваних $n+1-m-1$ вершин симплексу $S_{N'}$, при якому симплекс S_{N+1} , що утворений $n+1-m-1$ невідображуваними вершинами симплексу S_N і $m+1$ новими вершинами, є правильним симплексом S_N .

Вибрано критерії локально оптимального напрямку зсуву центру симплекса з множини можливих напрямків $\Omega_N(p)$:

$$I_1^N = -(\text{grad}f(x^N), p); \quad (1)$$

$$I_2^N = -\Delta f_N = f^*(x^N) - f^*(x^{N+1}); \quad (2)$$

$$I_3^N = -\Delta f_N / (m+1); \quad (3)$$

$$I_4^N(m) = \sum_{i=1}^m (f(x^{N,i}) - f^*(x^N)). \quad (4)$$

де $x^{N,i}$ – i -та вершина симплексу S_N ; $f^*(x^N)$ дорівнює будь-якому вимірному значенню $f(x^N)$ у центрі x^N симплексу S_N або середньому арифметичному значенню функції $f(x)$ у вершинах симплексу S_N .

$$f^*(x^N) = \frac{1}{n+1} \sum_{i=1}^{m+1} f(x^{N,i}) \quad (5)$$

Значення критерію I_1^N дорівнює величині проєкції одиничного вектора p на вектор антиградієнту і характеризує близькість напрямку p і зміщення центру симплекса x^N до антиградієнтного напрямку. Значення критерію I_2^N пов'язано зі зменшенням значення функції $f(x)$ у центрі симплексу S_N при зміщенні центру симплексу x^N в напрямі p . Значення критерію I_3^N дорівнює зменшенню $f(x)$ в центрі симплекса, відне-

сеному до одного з вимірів функції $f(x)$. Критерій I_4^N можна застосувати для відображення m вершин і він дорівнює сумі відхилень значень функції $f(x)$ в m вершинах від значення або оцінки значення (5) функції в центрі симплекса.

Для визначення напрямку $p_N^{(i)}$, вибрано локально-оптимальний пошуковий алгоритм, де на кожній N -й ітерації вектор $p_N^{(i)}$ і відповідні відображення вершини визначаються у результаті розв'язання задачі:

$$p_N^{(i)} = \arg \max_{p \in \Omega_N} I_i^N(p), \quad i = 1, \dots, 5. \quad (6)$$

Оскільки в симплексному алгоритмі напрям $p_N^{(i)}$ визначається відображеними вершинами m і 1 , то вираз (6) можна переписати у вигляді:

$$(m^N, 1^N) = \arg \max_{p \in \Omega_N} I_i^N(m, 1). \quad (7)$$

Послідовність значень цільової функції $f(x)$ в центрах симплексів $\{f(x^N)\}$ була монотонно спадною, причому розмір симплексу зберігається сталим при виконанні умови монотонності і стрибкоподібно змінюється при її порушенні. Відповідно до описаного підходу необхідно ввести правило перевірки монотонного спадання $\{f(x^N)\}$ і правило зменшення розміру симплекса R_N . Розмір симплекса будемо змінювати в моменти порушення умови (умова успішності кроку):

$$f^*(x^N) - f^*(x^{N+1}) \geq \varepsilon \|x^N - x^{N+1}\|^2 \quad (8)$$

відповідно до правила (9).

$$R_{N+1} = \gamma(t)R_1 \quad (9)$$

де $f^*(x^N)$ обчислюється за формулою (5) або дорівнює вимірному значенню $f(x^N)$ в точці x^N ; $\varepsilon > 0$, $0 < \gamma(t) \leq 1$ – параметри алгоритму; t – номер стиснення симплексу.

Вибрано алгоритм зменшення розміру симплекса. Нерухомою залишається вершина $x^{N,n+1}$ з мінімаль-

ним значенням цільової функції $f(x)$, а відстань від решти n вершин до вершини $x^{N,n+1}$ зменшується в $\gamma(t)/\gamma(t-1)$ разів.

Положення вершин визначається за формулами (10):

$$x^{N+1,i} = x^{N,n+1} + (x^{N,i} - x^{N,n+1})\gamma(t)/\gamma(t-1),$$

$$i = 1, \dots, n;$$

$$x^{N+1,n+i} = x^{N,n+1}. \quad (10)$$

Алгоритм безумовної мінімізації зі сталим розміром симплексу ґрунтується на формулах (11):

$$x^{N+1} = x^N + \frac{2m^N}{n+1-l^N} \Delta_N(m^N, l^N);$$

$$x^{N+1,j} = x^{N,j} + 2\Delta_N(m^N, l^N), \quad j=1, \dots, m^N;$$

$$x^{N+1,j} = x^{N,j} + \frac{2m^N}{n+1-l^N} \Delta_N(m^N, l^N), \quad (11)$$

$$j = m^N + 1, \dots, m^N + l^N;$$

$$x^{N+1,j} = x^{N,j}, \quad j = m^N + l^N, \dots, n+1$$

Алгоритм безумовної мінімізації зі сталим розміром симплексу показано на рис.3. Він полягає у наступному:

1. Побудувати правильний симплекс S_1 з центром x^1 і радіусом описаної гіперсфери R_1 .
2. Приймаємо $N = 1$.
3. Виміряти значення функції $f(x)$ у вершинах симплекса S_N .
4. Визначити $f^*(x)$ за формулою (5).
5. Пронумерувати вершини симплекса S_N в порядку зменшення значень функції $f(x)$ в цих вершинах.
6. Обчислити значення критерію I_i^N
7. Визначити $I_i^N(m^N, l^N) = \max I_i^N$
8. Відобразити $m^N + l^N$ вершин, побудувати симплекс S_{N+1} за формулами (11).
9. У нових вершинах симплекса S_{N+1} виміряти значення функції $f(x)$
10. Переходимо до наступного кроку, коли $N = N + 1$.

11. Визначити $f^*(x)$ за формулою (5).

12. Перевірити виконання рівності $f^*(x^N) - f^*(x^{N+1}) \geq \varepsilon \|x^N - x^{N+1}\|^2$

При її виконанні перейти до п.5. а в разі невиконання – до п.13

13. Пошук припинити. Запам'ятати вершину з мінімальним значенням функції.

ВИСНОВКИ

Сучасне управління наслідками різного роду надзвичайних ситуацій базується на активному застосуванні інформаційних технологій. Діяльність, яка пов'язана з інформаційними технологіями в цьому процесі, полягає у визначенні системи управління та пов'язаних з нею ресурсів, встановленні зв'язку з цими ресурсами та їх розгортанні там, де це необхідно. Проблема організації збору, обробки та аналізу інформації, є в даний час однією з найбільш актуальних і невіршених проблем.

Система, яка б могла виконувати завдання моніторингу стану навколишнього середовища повинна мати відповідне програмне, апаратне та інформаційне забезпечення. Існує глобальна проблема контролювання показників навколишнього середовища на трансграничних територіях, адже забезпечення повної інформації та чіткого плану координації дій у разі виникнення надзвичайної ситуації є неможливим. Для оцінки викидів та ідентифікації найбільш шкідливих речовин зроблено упор на збір даних з використанням електронних вимірювальних пристроїв дистанційного спостереження в режимі реального часу. На місцях роботи служб з ліквідування надзвичайних ситуацій використовуються різноманітні портативні прилади, які збирають актуальну інформацію про поточний стан навколишнього середовища. Зібрана інформація повинна бути швидко та безперебійно передана в командні центри для координації діяльності та забезпечення взаємодії всіх підрозділів.

В роботі пропонується для визначення послідовності точок вимірювання екологічних параметрів застосування методів деформованих конфігурацій, які використовують правильні симплекси в якості основної конфігурації. Управління процесом пошуку в таких методах здійснюється за рахунок вибору локально оптимально-

го напрямку зсуву центру симплекса і вибору величини кроку, що забезпечує оптимізацію функції в центрі симплекса. Це надає можливість визначити напрямок руху пересувних вимірювальних приладів, що дозволяє скоротити час локалізації місць з найвищим рівнем

небезпечної речовини за рахунок збіжності алгоритму за кінцеве число кроків. Отримані оперативні дані використовуються для первинної оцінки рівня екологічної безпеки та класифікації надзвичайної ситуації.



Рисунок 3. – Алгоритм безумовної мінімізації зі сталим розміром симплексу



Таким чином, використання запропонованого алгоритму в системі підтримки прийняття рішень щодо визначення стану екологічної безпеки, надає можливість пошуку початкових параметрів для збору даних та їх корегування в процесі екологічного моніторингу, що

дозволяє скоротити час на збір необхідної інформації для класифікації надзвичайної ситуації, локалізації зон екологічної безпеки, прийняття рішень щодо оцінки та ліквідації можливих наслідків.

ЛІТЕРАТУРА

1. Янников И. М., Пономарева Д. В., Современные системы мониторинга и прогнозирования чрезвычайных ситуаций // Инновационные технологии в машиностроении: сборник трудов VIII Международной научно-практической конференции, 18-20 мая 2017 г., Юрга. – Томск, 2017. – 2017. – С. 209-213.
2. Сангинова О. В., Бондаренко С. Г., Андриук В. К., Компьютерно-интегрированная система мониторинга и прогнозирования качества водных объектов. – 2017.
3. Wang H., Xu Z., Pedrycz W., An overview on the roles of fuzzy set techniques in big data processing: Trends, challenges and opportunities // Knowledge-Based Systems. – 2017. – Т. 118. – P. 15-30.
4. Zhukov A. N., Leonov P. Y., Problems of data collection for the application of the Data Mining methods in analyzing threshold levels of indicators of economic security // KnE Social Sciences & Humanities. – 2018. – Т. 3. – №. 2. – С. 369-374.
5. Power D. J., Sharda R., Burstein F., Decision support systems. – John Wiley & Sons, Ltd, 2015.
6. King C., Thomas D. S. G. Monitoring environmental change and degradation in the irrigated oases of the Northern Sahara // Journal of arid environments. – 2014. – Т. 103. – P. 36-45.
7. Thu H. N., Wehn U., Data sharing in international transboundary contexts: the Vietnamese perspective on data sharing in the Lower Mekong Basin // Journal of Hydrology. – 2016. – Т. 536. – P. 351-364.
8. Chettri N. et al. Long term environmental and socio-ecological monitoring in transboundary landscapes // International Centre for Integrated Mountain Development (ICIMOD), Kathmandu, Nepal. – 2015.
9. Разумов В. В., Разумова Н. В., Алексеев О. А., Требования к информации при наземно-космическом мониторинге предвестников природных и техногенных катастроф на приграничных территориях России // Современные проблемы дистанционного зондирования Земли из космоса. – 2017. – Т. 14. – №. 7. – С. 52-61.
10. Плуталова Т. Г., Мониторинг трансграничных территорий // Водные и экологические проблемы Сибири и Центральной Азии. – 2017. – С. 80.
11. Шабанов В. В., Маркин В. Н., Ведение мониторинга водных объектов в современных условиях // М.: Изд-во РГАУ–МСХА. – 2015.
12. Transon J. et al., Survey of Hyperspectral Earth Observation Applications from Space in the Sentinel-2 Context // Remote Sensing. – 2018. – Т. 10. – №. 2. – P. 157.
13. Frohn R. C., Lopez R. D., Remote Sensing for Landscape Ecology: New Metric Indicators: Monitoring, Modeling, and Assessment of Ecosystems. – CRC Press, 2017.
14. Bilitewski U., Turner A., Biosensors in environmental monitoring. – CRC Press, 2014.
15. Frontasyeva M. et al., Cloud platform for data management of the environmental monitoring network: UNECE ICP vegetation case. – 2016.
16. Исаев И.В., Рогачев А.Ф., Применение технологий виртуализации для систем поддержки принятия решений в сфере эколого-экономического менеджмента // Современная экономика: проблемы и решения. – 2016. – Т. 6. – С. 97-106.
17. Рыков А.С., Системный анализ: модели и методы принятия решений и поисковой оптимизации. – М.: Издательский Дом МИСиС, 2009. – 608 с.

*Рецензент: д.т.н., проф. Коваленко В.Ф.,
Херсонський національний технічний університет*

ПОРІВНЯЛЬНА ХАРАКТЕРИСТИКА АКТУАЛЬНИХ ЗАСОБІВ РОЗРОБКИ ПІД МОБІЛЬНІ ПЛАТФОРМИ

УДК 004.9

ЯШИНА Оксана Миколаївна

к.т.н., доцент кафедри ІПЗ Хмельницького національного університету

Наукові інтереси: дослідження та впровадження смарт-технологій у різні сфери життєдіяльності людини.

e-mail: ksusha.ja@gmail.com.

ГРЕМЕЧЕВСЬКИЙ Руслан Васильович

студент Хмельницького національного університету

Наукові інтереси: дослідження та прикладне застосування останніх напрацювань програмної інженерії у сфері проектування та розробки мобільних додатків.

e-mail: grem.rusl@gmail.com.

БРАТАСЮК Денис Іванович

студент Хмельницького національного університету

Наукові інтереси: дослідження та впровадження смарт-технологій у різні сфери життєдіяльності людини.

e-mail: mqaak1@gmail.com.

ВСТУП

Із бурхливим розвитком інформаційних технологій, що дозволяють створювати різні мобільні пристрої, ринок програмних продуктів отримує потужний стимул до розвитку та вдосконалення. Сучасну мобільну техніку люди носять із собою завжди та всюди (у світі нараховується близько п'яти мільярдів користувачів смартфонів та інших мобільних гаджетів). Смартфони, планшети та інші пристрої відіграють значну роль у побуті, особистому житті людей, роботі (є можливість легко та швидко прочитати файл, зайти на пошту, надрукувати документ за допомогою мережевого принтера тощо). Із збільшенням продажу мобільних пристроїв поступово сформувався окремий напрямок програмного забезпечення – мобільні додатки. Інформаційні технології вливаються у бізнес-процеси сучасних підприємств все інтенсивніше, стаючи більш зручними у використанні, доступними по ціні та легкими у використанні. Важко назвати бізнес, що не використовує сучасні інформаційні технології. Однак, одночасно із таким бурхливим розвитком постає питання про зручність отримання

даних. Саме мобільні технології здатні прийти на допомогу електронному бізнесу. Мобільні технології – остання та найбільш актуальна тенденція розвитку ринку інформаційних технологій для автоматизації бізнес-процесів. Смартфони та планшети все більше проникають у корпоративне середовище і перетворюються для багатьох керівників та співробітників із особистого мобільного пристрою у робочий інструмент. Керівники компанії та організацій задумуються про те, щоб централізовано впроваджувати та використовувати ці технології в інтересах бізнесу. Розробка програм для мобільних пристроїв у сфері бізнесу необхідна для просування товарів чи послуг широкій аудиторії. Якісний та зручний додаток користувачі будуть активно використовувати та рекомендувати друзям та знайомим. Якщо створювати додаток із врахуванням інтересів потенційних користувачів, то це підвищить лояльність та довіру до бренду, а також допоможе оптимізувати виконання вирішуваних задач. Так, наприклад, мобільний банк дає можливість контролювати, оплачувати рахунки, здійснювати переведення коштів тощо.



Розробка мобільних додатків актуальна у багатьох сферах, таких як транспортні послуги, мережі ресторанів та кафе, інтернет-магазини, журнальний бізнес, організація корпоративного спілкування і багатьох інших підприємств та торгівлі продукцією.

В сучасних умовах підхід Mobile First залучається у всі сектори економіки, замінивши попередній робочий процес, починаючи з веб-сайту та закінчуючи задоволенням потреб користувача на кінцевому шляху. Зі зростанням популярності мобільної платформи, було впроваджено багато технологій, що полегшують просування у сфері смартфонів.

Це почалося з гібридних фреймворків для розробки додатків, що дозволяють використовувати основні веб-технології, такі як HTML, CSS та JavaScript для компіляції та розгортання додатків на різних платформах. Потім з'явилися нативні фреймворки, що не тільки дозволяють використовувати веб-інструменти для мобільної розробки, але також використовують нативний API для Android чи iOS.

Ці платформи служать такій же меті, як підтримка та створення мобільних додатків за допомогою веб-технологій, таких як HTML, CSS, JavaScript. Те, чим вони відрізняються, - це функції, що вони надають, і як вони спілкуються з API для конкретних платформ.

Метою статті є визначення характеристик сучасних засобів розробки мобільних додатків, обґрунтування доцільності використання кожного із них в залежності від вимог до кінцевого продукту.

АНАЛІЗ ДОСЛІДЖЕНЬ ТА ПУБЛІКАЦІЙ

Зазвичай під мобільним бізнесом розуміють традиційні послуги електронного бізнесу, який забезпечується бездротовими мережами. Однак, мобільний бізнес забезпечує і принципово нові сервіси, які не можуть бути реалізовані за допомогою звичайної мережі, тому він виходить за її межі.

Отже, актуальність розробки мобільного додатку для керування бізнес-процесами на підприємстві, організації чи установі не викликає сумнівів..

Не зважаючи на стрімкий розвиток мобільних платформ та, відповідно, зростаючий попит на методи та засоби стосовно розробки для них, об'єми наукової інформації у цій сфері все ще невеликі та, у більшості

випадків, неактуальні. Звідси й підвищений науковий інтерес до даної теми.

ВИКЛАД ОСНОВНОГО МАТЕРІАЛУ

Розглядаючи спосіб створення додатків iOS та Android, укорінюється думка, що єдиним вибором є споріднені мови Objective-C, Swift та Java. Однак цей підхід має серйозний недолік: при розробці відразу на дві платформи є необхідність двічі писати ті ж фрагменти коду. Розглядаючи різні мови програмування (Java і Swift), різні підходи до того, щоб робити те ж саме в iOS та Android, різні способи створення графічного інтерфейсу, важко зробити що-небудь, щоб двічі не писати той самий код. Ситуація ще гірша, якщо два різних розробники створюють програми для iOS і Android, оскільки дві людини, ймовірно, реалізують однакову функцію трохи по-різному. Проте за останні кілька років виникла ціла нова екосистема платформ для створення мобільних додатків.

Немає жодних сумнівів стосовно того, що коли мова йде про розробку мобільних додатків - лідерами швидкодії є продукти, написані нативними мовами [1,2,7]. Це мінімізує рівні абстракцій. Принцип «менше обміну даними - краща швидкодія» як ніде краще працює на мобільних платформах, де системні ресурси є сильно обмеженими. Проте, подекуди різниця у бюджеті та терміни розробки набагато серйозніші за різницю у швидкодії. Вихід на ринок таких гравців як React Native чи Xamarin зробили цю прірву ще більш значущою, тому питання стало гострішим.

Ionic, Cordova і PhoneGap - засоби для розробки гібридних додатків. Найбільша проблема з гібридними додатками - це їх швидкодія та ефективність. Спочатку веб-сайти створювались для простих веб-сторінок, а не складних додатків, які розробляються наразі. Гібридні програми працюватимуть без перебоїв на високотехнологічних пристроях, але все ще не так гладко, як хотілося б. Обмеженість системних параметрів телефонів є найбільшою проблемою.

Зазначені засоби дають змогу програмістам створювати програми для мобільних пристроїв за допомогою CSS3, HTML5 та JavaScript, замість того, щоб спиратися на API, специфічні для платформи Android, iOS або Windows Phone. Це дозволяє загорнути код CSS, HTML та JavaScript в додаток, залежно від платформи пристрою.

Такі засоби розширюють можливості HTML та JavaScript для роботи з пристроєм. Проте отримані додатки будуть гібридними, а це означає, що вони не є нативними мобільними додатками (оскільки рендер представлення здійснюється за допомогою WebView замість власних елементів інтерфейсу платформи). Не можна їх називати і просто веб-додатками (оскільки вони упаковані як додатки конкретної платформи для розповсюдження та доступу до API пристроїв).

Інша проблема полягає в тому, що при розробці гібридного додатку автоматично успадковуються всі проблеми, які має розробник веб-сайтів. Як сайти можуть працювати по різному на деяких браузерах - так і додаток працюватиме зі своїми нюансами на певних пристроях. Діагностування помилок та відлагодження стає просто нестерпним для розробника.

Статистика говорить про те, що власники смартфонів витрачають більшу частину свого часу, використовуючи лише кілька додатків, і вони очікують, що будь-який новий додаток, який вони беруть у користування, буде швидким як Facebook, YouTube чи Uber. З високими сподіваннями користувачів гібридні додатки не справляються, пропонуючи більш обмежені можливості для користувачів з повільними анімаціями, відсутністю розпізнавання жестів, подекуди неправильною поведінкою клавіатури чи недостатньою взаємодією із мобільною платформою та її операційною системою.

Всі вищевказані проблеми ведуть до того що популярність гібридних додатків знижується. PhoneGap і Ionic демонструють помітні недоліки у великих додатках. На відміну від цього, список додатків, які перейшли з гібриду до рідного інтерфейсу на React Native стрімко збільшується. До списку входять Facebook, Airbnb, Instagram, Skype, Tesla, Slack та багато інших.

Найбільша користь, яку пропонує React Native, - це набагато краща продуктивність, ніж у випадку гібридних додатків. Розробник отримує доступ до всієї платформи і всіх нативних елементів інтерфейсу. Це дозволяє створювати додаток зі складними анімаціями та великою кількістю елементів управління. Крім того, апаратні функції обробляються конкретною платформою. React Native працює без використання WebView, тому розробнику не потрібно турбуватися про проблеми із сумісністю браузера. Нарешті, React Native базується на React структурі з набагато кращою підтримкою

- як від Facebook, так і від активної та чисельної спільноти. Завдяки постійній підтримці, React Native залишається актуальним, забезпечує більш високу надійність та чудову базу знань, де можна дізнатись як розробляти додатки та вирішувати проблеми.

Порівнюючи засоби розробки під мобільні платформи було б неправильно не пригадати ще один. Xamarin - це один з найстаріших крос-платформних фреймворків такого типу. Проект був придбаний компанією Microsoft у 2016 році та став частиною їх IDE Visual Studio, що забезпечило платформі стабільний подальший розвиток. Це одна з ключових причин, чому великі компанії, такі як, наприклад, Pinterest, покладаються на Xamarin. Він пропонує єдину мову - C#, бібліотеку класів і runtime, що працює на трьох мобільних платформах: iOS, Android та Windows Phone (рідна мова Windows Phone і є C#). Швидкодія та продуктивність продукту на виході достатні навіть для вимогливих ігор.

Все ж вибір конкретного патерну залишається за розробником, який визначає більш кращий шаблон для конкретного випадку. Мобільні технології в бізнесі є логічним продовженням електронного документообігу і часто розуміється як його підмножина.

Стосовно популярності розглянутих засобів було зроблено вибірку статистичних даних сервісу Google Trends (табл.1).

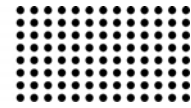
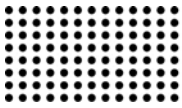
Таблиця 1

Вибірка статистичних даних сервісу Google Trends

	React Native	Xamarin	Ionic	Cordova	PhoneGap
В Україні	40%	25%	24%	10%	1%
У світі	18%	14%	44%	22%	2%

Статистика показує що React Native лідирує в Україні, Ізраїлі, Швеції та деяких інших країнах. Проте у світі більш популярним запитом в Google є Ionic. Загальна тенденція показує що React Native та Xamarin з часом впевнено відбирають популярність у технологій що працюють на WebView.

Так чи інакше, якщо порівнювати нативні та гібридні додатки то постає питання про перевагу того чи іншого виду у 2018 році. Для цього потрібно з'ясувати скільки додатків використовує звичайний користувач щодня



та скільки додатків він не відкривав з моменту встановлення.

Загалом же завантаження додатка займає кілька хвилин, а також кілька секунд, щоб видалити його, якщо він не відповідає вашим очікуванням. Користувачський інтерфейс дуже важливий для мобільних додатків: привабливі елементи інтерфейсу користувача, плавна прокрутка, розпізнавання жестів для платформ, розширені анімації та ефекти. На відміну від цього, гібридна розробка додатків повинна адаптуватися до специфіки користувацького інтерфейсу декількох платформ одночасно. Незважаючи на те, що крос-платформенні технології та системи активно вдосконалюються, гібридні додатки все ще можуть виглядати та некоректно працювати на різних пристроях. Таким чином, якщо є зацікавленість в користувацькому інтерфейсі, варто вибрати нативну розробку додатків. Щоб захистити гібридний підхід, можна сказати, що хороший веб-дизайнер може подолати недоліки UX і зробити додатки дуже схожими до нативного дизайну. Але є й випадки, такі як Google Gmail, коли розробники не приділяють великої уваги рідному користувацькому інтерфейсу та реалізують один дизайн для всіх платформ. Зрештою, користувачі не дуже піклуються про особливості вигляду і навіть не можуть заперечувати, що перед ними не нативний додаток (вони навряд чи думають про такі речі), якщо він відповідає всім своїм цілям.

Нативні додатки безпосередньо зв'язуються з апаратним забезпеченням пристрою, і, як правило, весь статичний вміст завантажується після встановлення додатка [2]. Таким чином, користувачі не залежать від швидкості інтернету, тому програма завжди плавно і надійно працює. На відміну від нативних додатків, гібридним спочатку потрібно пройти гібридну платформу і виконати код, а потім звернутися до пристрою. Як правило, вміст завантажується з сервера (хоча і не завжди), а продуктивність залежить від швидкості вашого інтернет-з'єднання. Крім того, можна відмітити деяке відставання, наприклад, переключивши iPad з вертикального в горизонтальне положення. Це є великим мінусом гібридного додатка. У нативному додатку зміна орієнтації екрана є швидкою та легкою, на відміну від гібридного (Facebook став швидшим з моменту переходу на нативний додаток).

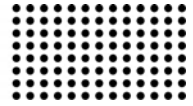
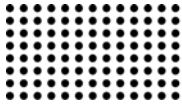
Однак, варто відмітити, що гібридні програми все ще повільніші, ніж їх конкуренти, але не так критично. Продуктивність - це індивідуальна річ, яка залежить від потреб проекту та можливостей пристрою. Розробнику та кінцевому користувачу потрібно звертати увагу на швидкість, якщо метою є створення додатку, що підтримує оновлення в режимі реального часу, високі навантаження (соціальні мережі, такі як Facebook), складну бізнес-логіку з кількома розрахунками (фото, музика або відеоредактори) та 3D-графіку (ігри). В інших випадках можна не звертати на це уваги. Наприклад, власник, може думати, що "прокрутка" на веб-сайті працює з деякими відставаннями, а користувачі навіть не звертають уваги на це [3]. Загалом же ефективність гібридних додатків, у останні роки, зросла завдяки наступним факторам:

1. Щороку пристрої стають набагато потужнішими та швидшими;
2. Дія ОС стає кращою від версії до версії;
3. Прогрес технологій;
4. Рівень кваліфікації та досвід розробників зростає.

Якщо говорять про використанні гібридних додатків з низькою продуктивністю, то це стосується старих пристроїв, застарілих ОС, недосвідчених розробників або поганої оптимізації використовуваних технологій.

Гібридні програми можуть перемагти свого конкурента лише в швидкості розробки програмного продукту. Маючи єдину кодову базу, ці додатки розробляються швидше, тому власник чи розробник може представити додаток на ринку вже через кілька місяців. Для пристосування функцій до різних платформ у гібридному підході потрібно менше часу, ніж створення дизайну для всіх можливих параметрів пристроїв (iPhone, планшетів, різних смартфонів Android тощо) у нативній розробці додатків. До речі, зміни в гібридах швидше доставляють користувачам, не заставляючи їх завантажувати оновлення.

Зазвичай розробники спеціалізуються лише на одній платформі (iOS або Android). Отже, щоб створити нативний додаток для різних платформ, знадобиться щонайменше дві команди розробників. Це також означає два бюджети. Будь-які подальші зміни та оновлення також дадуть кошти, помножені на кількість платформ. Розробка гібридних додатків є дешевшою, оскі-



льки деякі рішення швидше створюються за допомогою HTML та JS, аніж Objective-C або Java. Крім того, більша частина коду може бути повторно використана для різних платформ, хоча все ще існує необхідність адаптуватися до специфіки платформ. У будь-якому випадку знадобиться менше ресурсів і менше бюджету.

Далі подано переваги та недоліки нативних та гібридних додатків [4].

Переваги нативного додатку:

1. Набір багатих нативних інтерфейсів, який добре знайомий користувачам і допомагає їм інтуїтивно вичити додатки.

2. Повний потенціал платформи (багато готових та перевірених рішень).

3. Висока швидкість виконання.

4. Прямий доступ до функцій пристрою (камера, GPS, адресна книга, календар, push-сповіщення) через API.

Недоліки нативного додатку:

1. Тривалий розвиток, оскільки вимагає знання в різних мовах програмування.

2. Набагато дорожче, оскільки вам потрібно нагдувати більше фахівців для створення та підтримки різних платформ.

Переваги гібридного додатку:

1. Швидка розробка: одна кодова база для всіх платформ

2. Просте тестування та впровадження багатьох змін та масштабування на інші платформи.

3. Дешевше, адже оплата здійснюється лише однієї команді.

Недоліки гібридного додатку:

1. Не такий гладкий і інтуїтивно зрозумілий UX як у нативному додатку.

2. Більш низька продуктивність: може бути повільним для додатків із високою завантаженістю у випадку слабого підключення до Інтернету

3. Іноді збільшується витрата акумулятора для складних додатків.

Нативний тип розробки все ще знаходиться на вершині, оскільки набір функцій платформи постійно оновлюється, що робить гібридну розробку складнішою. Переваги нативних по відношенню до гібридних мобільних додатків - це краща продуктивність і UX. Нативна розробка краще підходить для високо інтерактивних

додатків з великою кількістю графіки та анімації або коли вміст потрібно швидко оновлювати. Тут ми маємо на увазі ігри, деякі типи соціальних мереж тощо. У випадку, якщо потрібні деякі ексклюзивні можливості, гібридна розробка може вимагати додаткових зусиль та консультації розробників мови платформи, щоб написати їх з нуля.

Гібриди з'явилися з метою пришвидшення розробки. Використання крос-платформ обмежується лише посібниками для мобільних платформ, які накладають свої конструкції та оновлення ОС. Велика перевага полягає в тому, що цей вид розробки дешевший. Гарно розроблений гібридний додаток не має видимих чи функціональних відмінностей для користувачів. Загалом же кінцевому користувачу не важливо як побудований додаток якщо він зручний та приємний у використанні.

Для розробки під ОС Android є три основні IDE: Eclipse (на сьогоднішній день використовується все рідше), IntelliJ idea, Android Studio. В якості інструмента збірки проекту є Ant. Дане IDE легко налаштовується, інтегрується з Neax для розробки з компонентами, такими як SDK Android, NDK та Java машиною. Після успішної інтеграції створити свій проект легко. В якості мови програмування використовується Java. Дане IDE дуже просте в роботі, тому воно підходить для початкового рівня розробки. IntelliJ IDEA - більш серйозний інструмент розробки. Надає такі корисні речі як закриття дужок після умови, груповий перезапис методів та автоматичне створення шаблонних класів (Interface, Singleton). Також тут є можливість зміни теми оформлення. Однак, варто відмітити, що даний проект не є відкритим, однак оновлення виходять дуже часто. Перша версія IntelliJ IDEA з'явилася у січні 2001 року й швидко здобула популярність, як перша Java IDE із широким набором інтегрованих інструментів для рефакторингу, що дозволяла програмістам швидко реорганізувати сирцевий код програм. Дизайн середовища орієнтовано на продуктивність праці програмістів, дозволяючи їм сконцентруватися на розробці функціональності, тоді як IntelliJ IDEA бере на себе виконання рутинних операцій.

Починаючи з шостої версії продукту IntelliJ IDEA надає інтегрований інструментарій для розробки графічного користувацького інтерфейсу.



З версії 9.0 є безкоштовний варіант Community Edition з відкритими кодами. Сирцеві коди відкритої версії IntelliJ IDEA Community Edition поширюються рамках ліцензії Apache 2.0. Бінарні пакунки підготовлені для Linux, Mac OS X і Windows.

До складу IntelliJ IDEA включені напрацювання, створені в результаті спільної роботи з компанією Google, яка використовувала IntelliJ IDEA як базис для своєї нового відкритого середовища розробки Android Studio. Завдяки співпраці істотно розширені штатні можливості IntelliJ IDEA з розробки застосунків для платформи Android.

Android Studio прийшло на зміну плагіну ADT для платформи Eclipse. Середовище побудоване на базі вихідного коду продукту IntelliJ IDEA Community Edition, що розвивається компанією JetBrains. Android Studio розвивається в рамках відкритої моделі розробки та поширюється під ліцензією Apache 2.0.

Середовище надає засоби для розробки застосунків не тільки для смартфонів і планшетів, але і для носимих пристроїв на базі Android Wear, телевізорів (Android TV), окулярів Google Glass і автомобільних інформаційно-розважальних систем (Android Auto). Для застосунків, спочатку розроблених з використанням Eclipse і ADT Plugin, підготовлений інструмент для автоматичного імпорту існуючого проекту в Android Studio.

Середовище розробки адаптоване для виконання типових завдань, що вирішуються в процесі розробки застосунків для платформи Android. У тому числі у середовищі включені засоби для спрощення тестування програм на сумісність з різними версіями платформи та інструменти для проектування застосунків, що працюють на пристроях з екранами різної роздільності (планшети, смартфони, ноутбуки, годинники, окуляри тощо). Крім можливостей, присутніх в IntelliJ IDEA, в Android Studio реалізовано кілька додаткових функцій, таких як нова уніфікована підсистема складання, тестування і розгортання застосунків, заснована на складальному інструментарії Gradle і підтримуюча використання засобів безперервної інтеграції.

Для прискорення розробки застосунків представлена колекція типових елементів інтерфейсу і візуальний редактор для їхнього компонування, що надає зручний попередній перегляд різних станів інтерфейсу застосунку (наприклад, можна подивитися як інтерфейс буде

виглядати для різних версій Android і для різних розмірів екрану). Для створення нестандартних інтерфейсів присутній майстер створення власних елементів оформлення, що підтримує використання шаблонів. У середовищі вбудовані функції завантаження типових прикладів коду з GitHub.

До складу також включені пристосовані під особливості платформи Android розширені інструменти рефакторингу, перевірки сумісності з минулими випусками, виявлення проблем з продуктивністю, моніторингу споживання пам'яті та оцінки зручності використання. У редакторі доданий режим швидкого внесення правок. Система підсвічування, статичного аналізу та виявлення помилок розширена підтримкою Android API.

Базовою IDE для iOS є Xcode. На відміну від відкритої ОС Android, iOS накладає своєрідну «монополію» на вибір засобів для розробника. Тим не менш, дана IDE по праву займає перше місце серед інструментів для розробки додатків. Це один із небагатьох інструментів, з яким приємно працювати. Xcode використовує єдине вікно робочого простору – workplace window, що містить більшу частину необхідних для роботи даних. З Xcode можна легко перейти від написання коду до його відлагодження та розробки користувацького інтерфейсу в межах одного робочого вікна. SDK iOS постійно розширює комплект інструментів Xcode, додаючи компілятори та фреймворки, необхідні для роботи з ОС. В якості мови для розробки виступає Objective-C – об'єктно-орієнтована мова для розробки всіх iOS-додатків.

ВИСНОВКИ

Виходячи з проведеного аналізу, можна стверджувати, що нативні додатки, які дають найкращу продуктивність, добре підходять для комплексних проектів, що працюють зі складною графікою (наприклад, відеоігри). Неправильний вибір засобу розробки у такому випадку точно дасть про себе знати у кінцевому продукті. У кращому випадку можна отримати невдоволеність користувачів, а у гіршому – взагалі не буде змоги впровадити такий застосунок.

Що стосується менш вимогливих додатків, при їх розробці на кілька платформ (наприклад, iOS та Android) доцільно звернути увагу на такі засоби як Xamarin чи React Native. Їх кінцевий продукт, що мати-

ме нативний користувацький інтерфейс, буде поводитись швидко та плавно при правильному застосуванні технологій. А користувач не помітить жодної втрати

продуктивності без спеціальних тестів. Крім того, це суттєво скорочує термін розробки, а відповідно і вартість.

ЛІТЕРАТУРА

1. Nader Dabit. React Native in Action. 2016. P. 64-70.
2. Stan Bershadskiy, Crysfel Villa. React Native Cookbook. 2018. P. 89-92.
3. Jonathan Peppers. Xamarin Cross-platform Application Development: Second Edition. 2015. P. 104.
4. Mahesh Panhale. Beginning Hybrid Mobile Application Development. 2015. P. 110.
5. Rahat Khanna, Sani Yusuf, Hoc Phan. Ionic: Hybrid Mobile App Development. 2017. P. 64-65.
6. Mark L. Murphy. The Busy Coder's Guide to Android Development. 2014. P. 157.
7. Christian Keur, Aaron Hillegass. iOS Programming: The Big Nerd Ranch Guide. 2015. P. 32.
8. Native whis hybrid mobile app development: what to choose in 2018 <https://greenice.net/native-vs-hybrid-mobile-app-development-choose/> (viewed on January 17, 2018).
9. React vs. Cordova, PhoneGap, Ionic etc.: <https://learnreact.design/2018/02/14/react-native-vs-cordova-phone-gap-ionic-etc/>.
10. React native VS Ionic:2 Comparision: <https://medium.com/swlh/react-native-vs-ionic-2-comparison-50aba900be6c> (viewed on April 13, 2017). – Title from the screen.

*Рецензент: д.т.н., проф. П.В. Сорокатий
Хмельницький національний університет*



КОНЦЕПТУАЛЬНЫЕ ОСНОВЫ ПОСТРОЕНИЯ ПРОДУКЦИОННЫХ СИСТЕМ ЭКСПЕРТНОГО ОЦЕНИВАНИЯ СОЦИАЛЬНО-ЭКОНОМИЧЕСКИХ СИСТЕМ

УДК 004.8

ХОДАКОВ Виктор Егорович

доктор технических наук, профессор, заведующий кафедрой информационных технологий факультета кибернетики и системной инженерии Херсонского национального технического университета

Научные интересы: системный анализ, информационные технологии.

e-mail: hodakov.victor@gmail.com.

ЯЦЮК Сергей Викторович

аспирант, кафедра информационных технологий факультета кибернетики и системной инженерии Херсонского национального технического университета

Научные интересы: информационные технологии.

e-mail: sergey.shambal@gmail.com.

ВВЕДЕНИЕ

На сегодняшний день как в мире в целом, так и в каждой стране в частности, сложилось неравномерное распределение доходов и вложений в экономику и развитие отдельно взятой территории. Украина не есть исключением: разница в экономическом развитии и благоустройстве регионов довольно высока даже по украинским меркам, что является следствием нескольких факторов: несовершенство системы распределения имеющихся ресурсов, неэффективное государственное управление, затянувшийся кризис в экономике страны. Результатом этого процесса есть всё более увеличивающаяся разница между развитием регионов и населённых пунктов, отток населения из менее комфортных для проживания и благоустроенных территорий к более комфортным, снижение социального и культурного уровня населения.

Для решения задач повышения эффективности развития регионов, более целесообразного распределения бюджетных средств и сокращения разрыва в комфортности проживания людей в разных регионах можно использовать экспертные системы. Экспертная система – это программное средство, использующее экспертные знания для обеспечения высокоэффективного решения неформализованных задач в узкой предмет-

ной области. Основу ЭС составляет база знаний о предметной области, которая накапливается в процессе построения и эксплуатации ЭС [1].

ПОСТАНОВКА ЗАДАЧИ

Цель работы – выработать концептуальные основы построения продукционных баз знаний и экспертных систем, реализация которых позволит повысить эффективность управления социально-экономическими системами.

Задачи работы:

- выделить факторы, влияющие на инвестиционную привлекательность региона;
- создать дерево целей для продукционной базы знаний;
- продемонстрировать применение методов, снижающих нагруженность и мощность базы знаний;
- рассмотреть алгоритм работы экспертной системы на основании продукционной базы знаний.

РЕШЕНИЕ ЗАДАЧИ

Инвестиционная привлекательность – совокупность объективных и субъективных характеристик объекта инвестирования, обуславливающих в совокупности потенциальный платежеспособный спрос на инвестиции в данную стра-

ну, регион, отрасль, предприятие (корпорацию). В зависимости от временного горизонта анализа, управления и прогнозирования могут быть выделены текущая и перспективная инвестиционная привлекательность. Также следует различать абсолютную инвестиционную привлекательность и сравнительную инвестиционную привлекательность (последнее предполагает сопоставление с показателями инвестиционной привлекательности других объектов инвестирования) [2, 3].

Оценка инвестиционной привлекательности – одна из важных составляющих эффективного управления социально-экономическими системами. Её можно рассматривать применительно к различным уровням, начиная от государства и региона, заканчивая районом или населённым пунктом, в зависимости от целей и объекта инвестирования, а также их масштабов.

Под термином «социально-экономическая система» (СЭС) в общепринятом смысле понимается совокупность ресурсов и экономических субъектов, образующих единое целое, взаимосвязанных и взаимодействующих между собой в сфере производства и потребления, обмена и распределения. Социально-экономическую систему, как и всякую другую систему, характеризует системное качество. Системой является совокупность объектов и процессов, называемых компонентами системы, взаимосвязанных и взаимодействующих между собой, которые образуют единое целое, обладающее свойствами, не присущими составляющим его компонентам, взятым в отдельности [4, 5].

Показатели СЭС, которые могут оказать влияние или быть интересными при оценке инвестиционной привлекательности, представлены в таблице 1 [6]:

Таблица 1

№	Группы факторов	Показатели
1	Экономико-географическое положение	площадь, млн. кв. км
		выход к морским границам
		наличие транспортных коридоров
		наличие зон приоритетного развития, шт.
		объекты природно-заповедного фонда
2	Ресурсно-сырьевой потенциал	выбросы загрязняющих веществ в атмосферу, млн. тонн
		топливо-энергетические ресурсы
		металлургические ресурсы
		строительное сырье
		водные ресурсы
		лесные ресурсы
3	Трудовой потенциал	рекреационные ресурсы
		площади плодородных почв, тыс. га

№	Группы факторов	Показатели
3	Трудовой потенциал	плотность населения, чел./кв. км
		численность населения, чел.
		стоимость рабочей силы, грн./час
		предложение рабочей силы, чел.
		миграционный прирост населения, чел.
		среднегодовая численность штатных работников, чел.
4	Экономический потенциал	индексы объема продукции промышленности, %
		индексы продукции сельского хозяйства, %
		оптовый товарооборот, млн. грн.
		розничный товарооборот, млн. грн.
		чистый доход, млн. грн.
		валовой региональный продукт, млн. грн.
		индекс промышленной продукции, %
		финансовый результат деятельности малых предприятий, млн. грн.
		количество активно работающих предприятий сферы услуг, шт.
		доход от реализованных услуг, млн. грн.
		количество предприятий и организаций, шт.
		оборот розничной торговли, млн. грн.
		уровень развития малого предпринимательства
		финансовый результат, млн. грн.
процент предприятий, получивших прибыль, %		
уровень рентабельности операционной деятельности, %		
5	Инфраструктурный потенциал	наличие рыбного хозяйства
		плотность дорог, км/кв. км
		наличие аэропортов
		наличие морских портов
		наличие межрегиональных транспортных узлов
6	Научно-технический потенциал	грузооборот транспорта, млн. тонн
		пассажирооборот транспорта, млн. пасс.
		численность специалистов научной деятельности, чел.
		общая сумма затрат на инновации, млн. грн.
		количество специалистов с докторской научной степенью, чел.
7	Здравоохранение	внедрение новых прогрессивных технических процессов
		освоено производств новых видов продукции
		количество специалистов с кандидатской научной степенью, чел.
		количество научных организаций, шт.
8	Инвестиционный потенциал	количество лечебных учреждений, шт.
		плановая емкость амбулаторно-клинических учреждений, кол. посещений за 1 смену
		количество больничных коек, шт.
9	Инвестиционные риски	объем капитальных инвестиций, млн. грн.
		объем прямых иностранных инвестиций, млн. грн.
		объем экспорта, млн. грн.
		объем импорта, млн. грн.
10	Природно-климатический потенциал	объем инвестиций в основной капитал, млн. грн.
		законодательные
		политические
		социальные
		экономические
		экологические
		среднегодовые изотермы (лето), °С
		среднегодовые изотермы (зима), °С
		среднегодовой уровень осадков, мм. рт. ст.
		влажность воздуха, %
среднегодовые показатели силы ветра, м/сек.		
глубина промерзания грунта, м		
количество солнечных дней		

Для оценки инвестиционной привлекательности регионов при помощи информационных систем можно воспользоваться экспертными системами и продукционными базами знаний. В реальной экспертной системе необходимо учитывать максимально возможное количество показателей, что даёт более точный и объёмный результат, однако добавление каждого нового узла увеличивает мощность базы знаний, поэтому для лучшего восприятия связей элементов экспертной системы опустим большинство показателей и построим простую базу знаний, которая отвечает на вопрос «Привлекателен ли регион для инвестиций?» (рис. 1). Истоками базы будут являться: наличие транспортных ко-

ридоров (транспорт), количество ресурсов (ресурсы), численность населения (население), стоимость рабочей силы (стоимость), предложение рабочей силы (предложение).

Пусть истокам соответствуют значения:

- 1) Транспорт – мало, есть;
- 2) Ресурсы – мало, умеренно, много;
- 3) Население – мало, умеренно, много;
- 4) Стоимость – низкая, средняя, высокая;
- 5) Предложение – низкое, высокое.

Целевой вершиной будет являться «Инвестировать в регион» со значениями – да, нет.

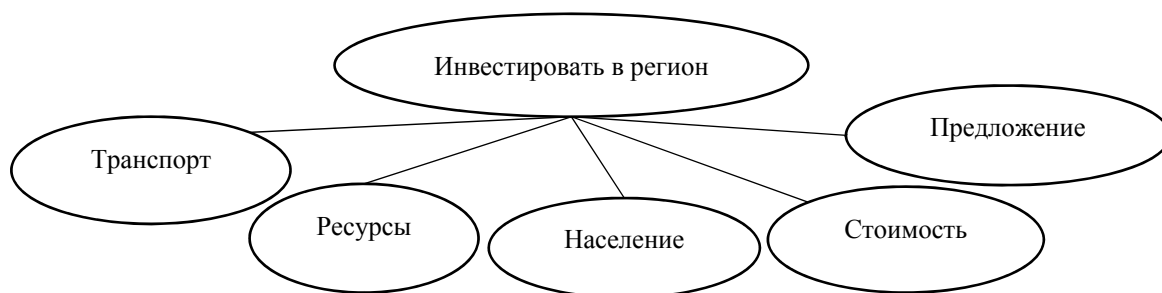


Рисунок 1 – Дерево целей для базы знаний «Инвестировать в регион»

Для структуры дерева целей, приведенного на рисунке 1, мощность считается путем перемножения количества всех возможных вариантов по каждому истоку. Для рассматриваемого примера количество вариантов по вершинам равны:

- 1) Транспорт – 2;

- 2) Ресурсы – 3;
- 3) Население – 3;
- 4) Стоимость – 3;
- 5) Предложение – 2.

Значит мощность базы знаний равна $= 2 * 3 * 3 * 3 * 2 = 108$ правил.

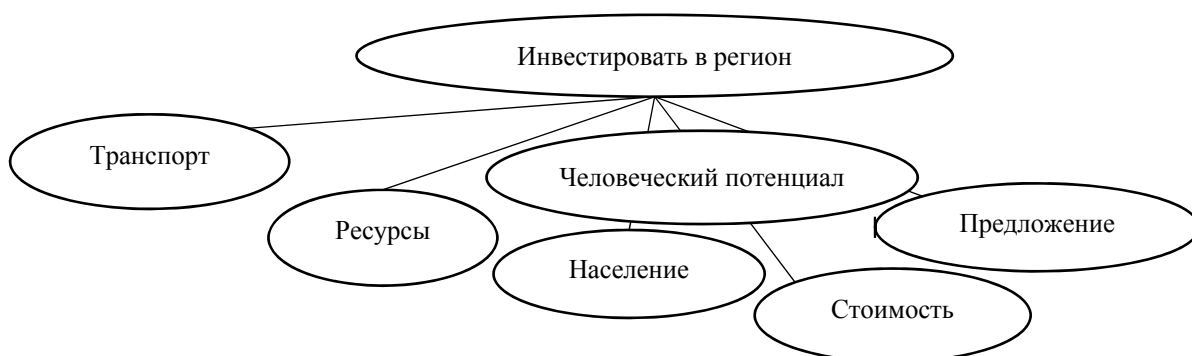


Рисунок 2 – Дерево целей с промежуточными вершинами

Если присмотреться к имеющемуся на рисунке 1 дереву целей, можно прийти к выводу, что вершины «население», «стоимость» и «предложение» логически

описывают один из аспектов вопроса «Инвестировать в регион?» (это также видно из таблицы 1, в которой все три вершины принадлежат к одной категории –

трудовой потенциал»), поэтому их можно объединить одной общей вершиной – «человеческий потенциал».

После добавления одной новой вершины дерево целей изменится (рис. 2).

Вершина «человеческий потенциал» может принимать значения: низкий, средний, высокий – 3.

Введение новой промежуточной вершины даёт возможность разбить базу знаний на две части, что позволяет снизить её мощность, тем самым снизив сложность процесса нахождения ответа. Первая объединит вершины «население», «стоимость», «предложение» и «человеческий потенциал»; вторая – «инвестировать в регион», «транспорт», «ресурсы», «человеческий потенциал». Каждая из частей представляет собой дерево, подобное изображённому на рисунке 1, и их мощности будут рассчитываться аналогично:

— Мощность первой части = $3 * 3 * 2 = 18$.

— Мощность второй части = $2 * 3 * 3 = 18$.

Общая мощность рассчитывается, как сумма мощностей, входящих в неё частей, то есть $18 + 18 = 36$.

Для дерева целей, изображённого на рисунке 2, дерево решений примет следующий вид (рис. 3 и 4). Если посмотреть на получившееся дерево решений (рис. 3) с точки зрения логики, можно прийти к следующим заключениям:

— Инвестировать в регион не стоит, когда человеческий потенциал низкий либо средний при остальных низких параметрах.

— Инвестировать в регион стоит, если человеческий потенциал высокий либо средний, но при этом присутствуют положительные остальные показатели.

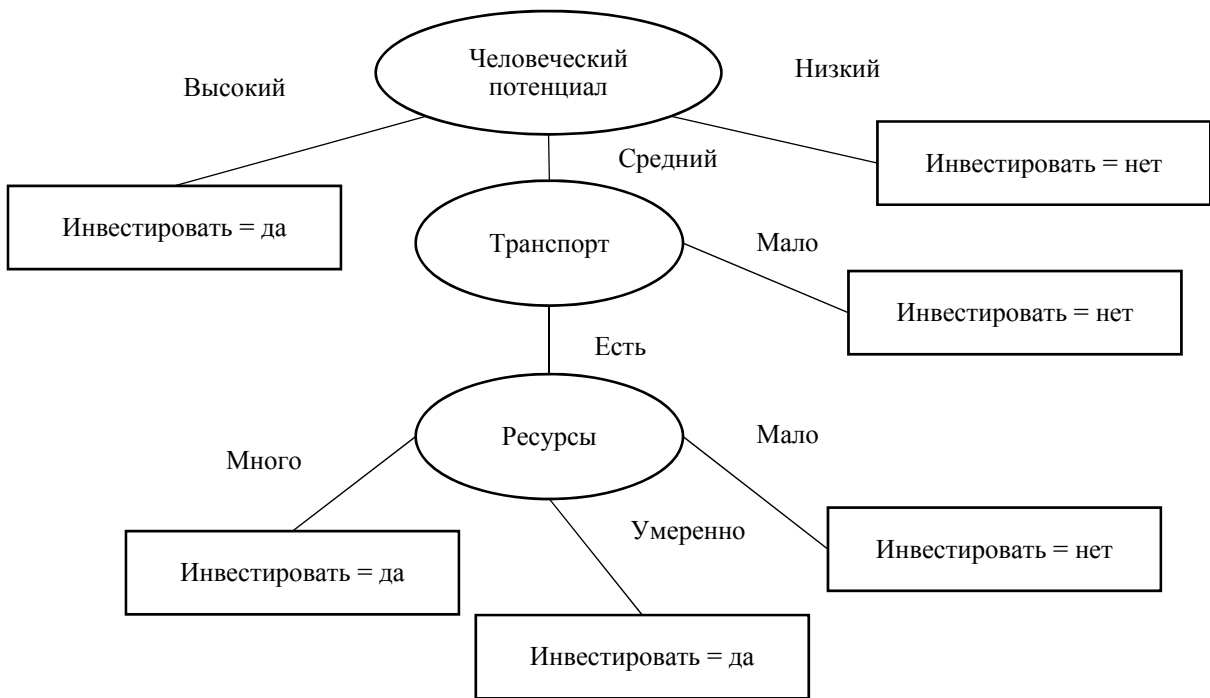


Рисунок 3 – Дерево решений для целевой вершины

Аналогичные выводы можно сделать, проанализировав дерево решений, изображённое на рис. 4:

- Человеческий потенциал является низким, если:
 - 1) мало населения;
 - 2) умеренное количество населения, однако высокая стоимость труда;

3) средняя или высокая стоимость труда при низком предложении и умеренном количестве населения.

— Человеческий потенциал является средним, если:

- 1) умеренное количество населения, высокое предложение рабочей силы, однако средняя её стоимость;

2) умеренное количество населения, низкое предложение, при этом низкая стоимость труда;

3) большое количество населения, низкое предложение с высокой или средней стоимостью рабочей силы.

— Человеческий потенциал является высоким, если:

1) умеренное количество населения, высокое предложение рабочей силы и низкая её стоимость;

2) большое количество населения, высокое предложение рабочей силы либо низкое, но при этом с низкой стоимостью оплаты труда.

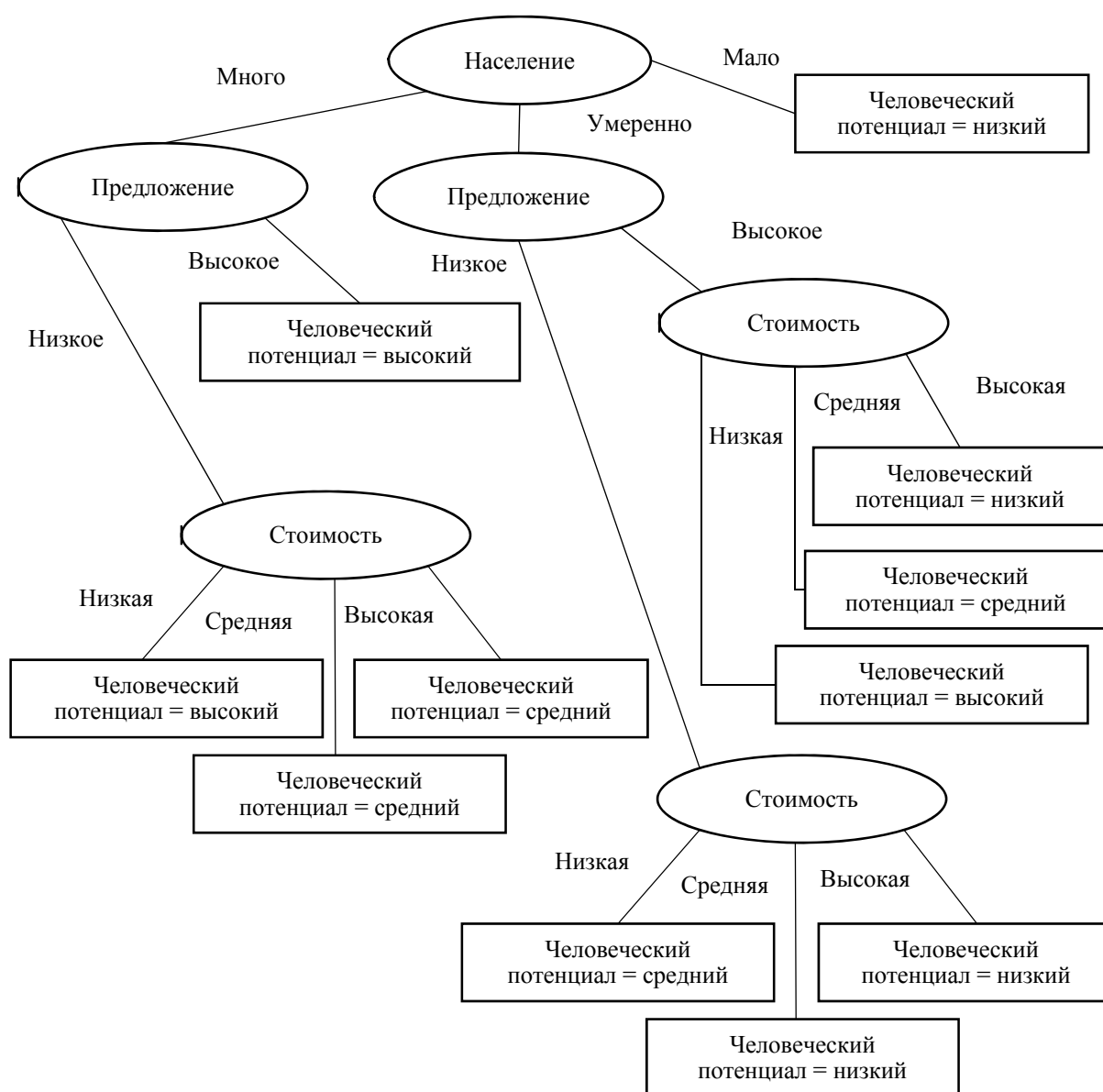


Рисунок 4 – Дерево решений для промежуточной вершины «Человеческий потенциал»

В результате, количество правил вместо $18 + 18 = 36$ получится $6 + 11 = 17$.

Построим дерево правил, которое предназначено для изображения всех правил базы знаний. Если для

базы знаний было выполнено сокращение дерева решений (отсечены ветви), то в дереве правил появятся правила (кружки), к которым линиями присоединены не все истоки (квадраты слева).

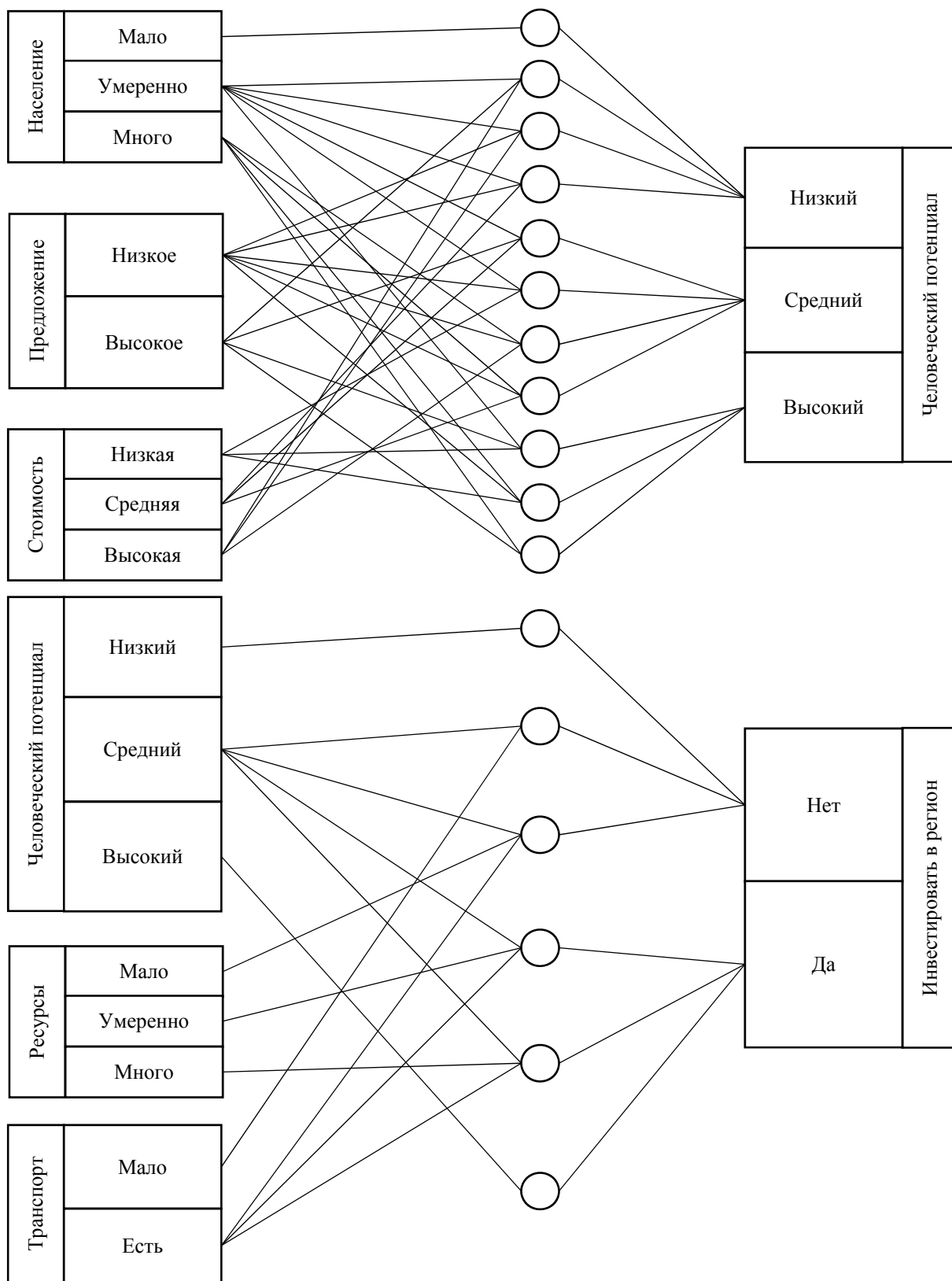


Рисунок 5 – Дерево правил



Рассмотрим последовательность действий системы на конкретном примере. Пусть для региона будут актуальны следующие параметры: Население – Много, Предложение – Низкое, Стоимость – Средняя, Транспорт – Есть, Ресурсы – Умеренно. На первом этапе система определяет, что переменная «Население» имеет значение «Много», отбрасывая остальные ветки правил. Согласно диаграммам на рисунках 3 и 4, если «Население» равно «Много», следует проверить «Предложение». «Предложение» определено, как «Низкое». В случае «Высокое», можно было бы задать «Человеческий потенциал» и перейти к определению основного вопроса, но для «Низкое» необходимо рассмотреть «Стоимость». В данной ветке правил для «Стоимость» равна «Средняя» определяется «Человеческий потенциал» как «Средний». Имея «Человеческий потенциал» равным «Высокий» или «Низкий», можно сразу выдать ответ на основной вопрос. Однако в нашем случае следует продолжить углубление в дерево правил и посмотреть значение переменной «Транспорт». Поскольку «Транспорт» равен «Есть», то снова ответ не может быть выдан. Остаётся углубиться на последний уровень и рассмотреть оставшуюся переменную «Ресурсы». Для данной ветки при значении «Ресурсы» равно «Умеренно»

выполняется набор минимально необходимых условий для выдачи положительного ответа. Результат: для выбранных исходных данных ответом на вопрос «Инвестировать в регион?» является значение «Да».

ОСНОВНЫЕ РЕЗУЛЬТАТЫ И ВЫВОДЫ

В работе представлены концептуальные основы построения продукционных баз знаний и экспертных систем: выделены факторы, влияющие на инвестиционную привлекательность региона, созданы деревья целей для продукционной базы знаний; продемонстрировано применение методов, снижающих мощность базы знаний; рассмотрен алгоритм работы экспертной системы на основании продукционной базы знаний. Данную последовательность действий можно использовать для реализации реальных экспертных систем, которые позволят повысить эффективность управления социально-экономическими системами, а в частности: определить территориальные и отраслевые направления денежных и других ресурсных инвестиций, рациональнее распределять бюджетные средства, выявлять и использовать преимущества и недостатки каждого региона.

ЛИТЕРАТУРА:

1. <http://www.habarov.spb.ru/> — Информационный портал Санкт-Петербургской лесотехнической академии. Автор – к.т.н., доцент Хабаров С. П.
2. <http://ru.science.wikia.com/wiki/> — Научный wiki-portal.
3. Filimonov V.S. Poniatie investitsionnoy privlekatelnosti predpriyatiya v sovremennyh rynochnykh usloviyakh // V sbornike "Aktualnye problemy nauki, ekonomiki i obrazovaniya XXI veka". – 2012 – s. 212 – 216.
4. Lapygin Yu.N. Teoriya organizatsii. M.: Infra-M. – 311s., 2007.
5. S.V. Silkin. Razvitie sotsialno-ekonomicheskikh sistem v sovremennykh usloviyakh na primere krupnogo goroda. Nauchnie zapiski Novosibirsk State University of Economics and Management. – 2009/4.
6. Prirodno-klimaticheskie faktory i sotsialno-ekonomicheskie sistemy: monografiya / V.E. Hodakov, N.A. Sokolova – Herson: HGMA, 2016. – 604s.
7. Abdikeev N.M. Proektirovanie intellektualnykh sistem v ekonomike: uchebник / Pod red. N.P. Tihomirova. – M.J Izdatelstvo "Ekzamen", 2004.
8. Gavrilova T.A., Horoshevskiy T.F. Bazy znaniy intellektualnykh sistem. – SPb.: Piter, 2001.
9. Telnov Yu.F. Intellektualnye informatsionnye sistemy v ekonomike. Uchebnoe posobie. Izdanie 3, rasshirennoe i dorabotannoe. – M.: SINTEG, 2002.
10. Yarushkina N.G. Osnovy teorii nechetkiy i gibridnykh sistem: Ucheb. posobie. – M.: Finansy i statistika, 2004.

*Рецензент: д.т.н., проф. В.А. Доровской
Профессор кафедры информатики и
социально-гуманитарных дисциплин
Криворожского филиала ЧВУЗ
«Европейский университет»*

АЛГОРИТМ ПОШУКУ ЗВ'ЯЗКІВ І ЗАЛЕЖНОСТЕЙ У ДАНИХ ВЕБ-СТОРИНОК

УДК 004.82 004.622 519.766

КАТЕРИНИЧ Лариса Олександрівна

кандидат фізико-математичних наук, доцент, Київський національний університет імені Тараса Шевченка

Наукові інтереси: штучний інтелект, експертні системи, веб-програмування.

e-mail: katerynych@gmail.com.

ПЕТЕЛЬКО Юрій Юрійович

аспірант, Київський національний університет імені Тараса Шевченка,

Наукові інтереси: веб-програмування.

e-mail: ypetelko@gmail.com.

На даний момент існує велика кількість пошукових систем, які здійснюють тією чи іншою мірою релевантний пошук у колекціях документів. Під терміном «документ» мається на увазі будь-який носій текстової інформації: бази даних, веб сторінки, системи електронного документообороту, тощо. Здійснюючи запит у тій чи іншій системі, користувач зазвичай отримує відповідь у вигляді списку документів, які пошукова система вважає релевантними відповідно до отриманого запиту. Існує декілька способів оцінки релевантності документа відповідно до отриманого запиту. Оцінка релевантності на пряму залежить від реалізації пошукової системи, яка, ґрунтуючись на деякому власному алгоритмі, вираховує відповідність вихідного документа отриманому запиту. Так, наприклад, для систем пошуку у колекціях систем електронного документообороту (зазвичай у таких системах зберігають строгі і конкретні документи юридичного чи економічного змісту) найбільше на релевантність впливає кількість входжень слів або словосполучень із запиту, а у системах веб-пошуку (Google, Bing, Yahoo), де інформація зберігається у неструктурованому вигляді, є неформалізованою, важливим фактором при визначенні релевантності документу є кількість гіперпосилань на даний документ з інших документів, відвідуваність веб-сторінки, кількість раніше здійснених запитів (схожих з

даним за тими чи іншими ознаками), які були здійснені і у яких даний документ був визначений як релевантний.

Здійснивши пошук у будь-якій з таких систем, на вихід користувач отримує список документів (посилань на веб-сторінки), які система вважає релевантними. Після цього, щоб отримати важливу для користувача інформацію, потрібно відкрити (переглянути) один чи декілька з отриманих документів і проаналізувати їх. При цьому, різна інформація про об'єкт пошуку може міститися в різних документах різної релевантності і часто для того, щоб дізнатися деякий факт про об'єкт пошуку доводиться переглянути велику кількість документів для пошуку взаємозв'язку між інформацією про об'єкт що міститься в кожному з цих файлів.

Тут постає питання пошуку таких взаємозв'язків. Як сказано вище, сучасні системи пошуку просунулись відносно далеко у питаннях визначення релевантності документів, проте такі системи не здійснюють аналізу інформації для розпізнавання перехресних зв'язків.

Метою алгоритму пошуку зв'язків і залежностей (АПЗЗ) даних веб-сторінок є саме такий аналіз. Отримавши на вхід запит про деякий об'єкт, явище чи особу, система, що реалізує АПЗЗ, на вихід надає інформацію про об'єкт, а також інформацію про зв'язки цього об'єктами з іншими.

• **Визначення понять.** Перед тим як надати формальний опис всього алгоритму опишемо визначення ключовим поняттям, які використовуються при описі.

Запит (далі позначатимемо «Зп») – набір слів (формалізованих або заданих природньою мовою), які тією чи іншою мірою описують об'єкт пошуку. Поняття запити є інтуїтивно зрозумілим. Наприклад: припустимо, що користувачу мережі інтернет (пошукової системи) потрібно знайти інформацію про «Києво-Могилянську академію». В такому разі запит може мати наступний вигляд: «Національний університет «Києво-Могилянська академія»» або «НаУКМА» або «університет Могилянка».

Експерт (користувач) – особа, яка здійснює пошук. Особа, яка здійснює пошук, зазвичай знає деякий об'єм інформації про об'єкт пошуку або цікавиться лише деякою частиною інформації про об'єкт. Для прикладу, при пошуку інформації про президента США, Біла Клінтона, після здійснення запити «Клінтон» до пошукової системи експерт може одразу визначити ті сторінки нерелевантними, на яких міститься інформація про Хіларі Клінтон (дружину біла Клінтона).

Неважливі слова (далі позначатимемо «Нс») – слова, словосполучення, терміни і символи, які не дають ніякою корисної інформації про об'єкт пошуку. До таких слів можна віднести усі знаки пунктуації, займенники, більшість дієслів, службові частини мови, спеціальні символи розмітки (HTML, XML, тощо).

Ключові слова (далі позначатимемо «Кс») – слова і терміни, які несуть смислове навантаження і описують об'єкт пошуку. Опис об'єкту пошуку здійснюється у запиті експертом, який здійснює пошук. Тоді множина ключових слів є найбільшою підмножиною слів з множини слів запити, яка не містить неважливих слів. Або більш формалізовано: $Kc = Zp \setminus Nc$.

Стоп-слова (далі позначатимемо «Сс») – слова та терміни, що визначаються експертом як такі, що не повністю не відповідають його запити і при будь-яких можливих зв'язках з запитом зменшують релевантність документу, у якому зустрічається стоп-слово, або експерту неважливі зв'язки об'єкту пошуку із словом позначеним як стоп-слово. З прикладу про Клінтона, слово «Хіларі» може бути позначене як стоп-слово, якщо експерта не цікавлять зв'язки Біла Клінтона із

дружиною, або слова «Афганістан» чи «війна» можуть бути позначеними як стоп-слова, якщо експерт шукає інформацію про благодійні вечори, у яких брав участь президент США.

Ресурс – документ, що містить деяку текстову інформацію. Ресурсом будемо називати будь-який текстовий документ, веб-сторінку (чи навіть веб-портал).

Пошукова система – система, що здійснює пошук ресурсів згідно заданого запити експерта без їхнього аналізу та пошуку зв'язків. Пошуковими системами є Google, Bing, різні вузькоспеціалізовані і корпоративні системи пошуку інформації у колекціях документів.

Веб-сторінки являють собою HTML документи. Важливо зрозуміти, як саме інформація розподілена у HTML документах. Сучасні веб-сторінки – це частинки деякого веб-порталу чи веб-застосунку, що несуть велику кількість інформації і оперують дуже об'ємними HTML документами, як динамічними так і статичним. Кожен такий документ, окрім корисної для користувача інформації містить дуже багато інформації, яка є неважливою для користувача. Крім того, оскільки HTML представляє інформацію в ієрархічному вигляді і кожен елемент структурно належить тому чи іншому тегові, можна зробити висновок, що інформація, яка має сенс, а також несе деякі знання розміщується в межах одного тегу або в деякій безперервній частині документу HTML розмітки. Тобто, важлива інформація про об'єкт пошуку міститься у безпосередній близькості від ключового слова, за яким здійснювався пошук. Таким чином, можна сказати, що для знаходження зв'язків об'єкта пошуку з іншими об'єктами чи явищами, необхідно проаналізувати об'єкти, явища і терміни, які знаходяться в деякому околі від ключових слів пошуку у документі HTML. Зазвичай такі об'єкти несуть додаткову інформацію чи приховані знання про об'єкт пошуку і можуть бути використані в уточнюючих запитах до сховища даних (у нашому випадку пошукової системи), для виявлення наступних зв'язків та прихованих знань.

РОБОТА З ПОШУКОВОЮ СИСТЕМОЮ

Пошукова система є одним з ключових елементів у роботі алгоритму пошуку зв'язків і залежностей у колекціях документів і служить сховищем даних. АПЗЗ не передбачає здійснення пошуку релевантних ресурсів. Цим займається пошукова система. Пошукова система

виконує функції очищення, інтеграції і вибору даних. Робота алгоритму починається з отримання системою, що реалізує АПЗЗ (далі «система»), запиту від експерта. Запит може бути заданий як природньою мовою, так і іншими, більш формалізованими способами. Отримавши такий запит система надсилає аналогічний (або

нормалізований) запит до пошукової системи, для отримання множини ресурсів, які позначаються пошуковою системою як релевантні документу. Схема взаємодії системи та пошукової системи зображена на рисунку нижче:

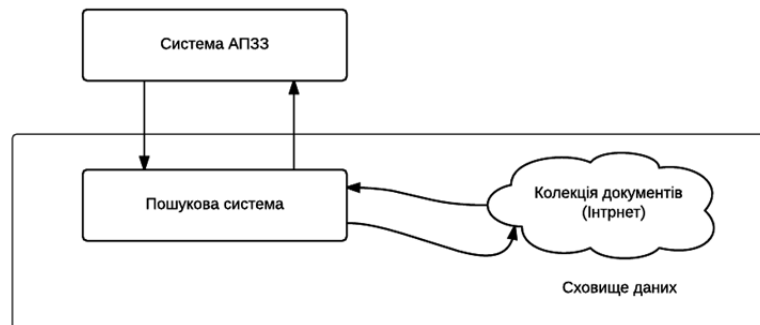


Рисунок 1.

Як видно на рисунку, робота системи (АПЗЗ) з пошуковою системою зводиться до наступного:

1. Система (АПЗЗ) отримує запит від експерта.
2. Здійснює первинну нормалізацію запиту (може не здійснювати, якщо таку нормалізацію здійснює сама пошукова система).
3. Здійснює отриманий після нормалізації (якщо нормалізація мала місце, якщо ні, то аналогічний тому, який прийшов на вхід системі) запит до пошукової системи.

У відповідь на наданий запит система отримує від пошукової системи список релевантних ресурсів (множину $D = \{T_1, T_2, \dots, T_n\}$), серед яких і буде здійснюватись подальший пошук зв'язків і залежностей.

Як сказано вище, якість роботи алгоритму і всієї системи прямопропорційно залежить від якості пошукової системи, яка використовується. При зменшенні релевантності пошуку множини ресурсів, серед яких здійснюється пошук взаємозв'язків, якість такого пошуку буде теж зменшуватися.

НОРМАЛІЗАЦІЯ ТЕКСТОВОЇ ІНФОРМАЦІЇ

При роботі з інформацією представленою у текстовому виді важливим для будь-якого алгоритму є процес нормалізації тексту. Як описано вище, система АПЗЗ отримує запит експерта (можливі запити природньою мовою) і список релевантних ресурсів (множина D), які

потрібно проаналізувати, для визначення взаємозв'язків. Оскільки запити і інформація, що зберігається на кожному ресурсі може бути представлена природньою мовою, процес нормалізації полягає у наступному:

1. Розбиття тексту на лексеми та побудова множини усіх лексем.
2. Лемматизація та стемматизація кожної лексеми.
3. Вилучення з отриманої множини усіх H_s .

Після такої нормалізації отримується множина термінів (без неважливих слів, знаків розмітки документів, знаків пунктуації), яку можна використовувати у подальшому аналізі. Важливо відзначити, що терміни у такій множині зустрічаються у такому ж порядку, як і у ресурсі, над текстом якого здійснювалася нормалізація.

Процес нормалізації є важливим, особливо, якщо пошук здійснюється мовами з розвинутою морфологією.

ПОБУДОВА МНОЖИНИ КЛЮЧОВИХ ТЕРМІНІВ

Наступним кроком після отримання списку ресурсів з релевантною інформацією, нормалізації і побудови множини термінів для кожного ресурсу є процес аналізу кожного ресурсу для пошуку взаємозв'язаних з об'єктом пошуку понять.

Перед тим, як перейти до пояснення методу відбору паттернів і зв'язаних понять, варто розібратися з тим,

як дані розміщені на веб-сторінках. Сучасні веб-сторінки мережі Інтернет містять велику кількість текстової інформації, більшість якої не несе ніякого смислового навантаження і є або рекламою, або з досить високої ймовірністю інформацією, що не має зв'язку з шуканою. Тому будувати набори частих паттернів (itemsets) для усіх термінів, що зустрічаються на ресурсі є малоефективним і часозатратним. З іншого боку, важлива смислова інформація, щодо об'єкту пошуку, зазвичай, знаходиться в деякому околі ($\pm 10-20$) термінів навколо ключових слів. Саме таким чином будується множина термінів, які вважаються асоційованими з термінами у запиті.

Побудова множини важливих слів здійснюється наступним чином:

1. Нехай S_{all} – множина пар <термін, кількість> усіх важливих термінів, порожня на початку аналізу.

2. Для кожного ресурсу T із множини D , будується множина термінів (позначимо її S), нормалізованих, як описано вище. Позиція кожного терміну у такій множині відома, тому до будь-якого терміну з побудованої множини можна звернутися за індексом.

3. Для кожного із термінів запиту, знаходимо його індекс у множині S .

4. Для кожного знайденого у пункту 3 індексу i , будуємо його околі ($i-n$, $i+n$), де n – деяка наперед задана константа.

5. Для кожного індексу з околу ($i-n$, $i+n$) з множини S , додаємо термін, що знаходиться за цим індексом до множини S_{all} таким чином, що якщо термін вже існує у множині S_{all} збільшуємо значення кількості на одиницю, якщо ні – додаємо пару (термін, 1).

Після такого аналізу множина S_{all} міститиме терміни, які зустрічаються з ключовими словами найчастіше у деякому околі, що вказує на взаємозв'язок між об'єктом пошуку та знайденими термінами.

Оскільки множина S_{all} містить пари (термін, кількість) її можна відсортувати за кількістю і отримаємо терміни у порядку найчастішого вживання з ключовими словами. Чим частіше зустрічається термін у такій множині, тим сильнішим є його зв'язок з об'єктом пошуку.

Як сказано вище, пошуком релевантних даних займається сховище даних, у нашому випадку це пошукова система. Використання побудованої множини

термінів S_{all} , які можуть мати зв'язок до об'єкта пошуку, для побудови уточнюючих запитів до сховища даних може мати якісний вплив на його релевантність і для виявлення нових зв'язків.

Уточнюючі запити – це повторно здійснений запит до сховища даних, у нашому випадку – до пошукової системи, у якому змінився набір ключових слів, щоб підвищити релевантність пошуку ресурсів. Як сказано вище, після першої ітерації роботи алгоритму, у розпорядженні експерта є множина термінів, пов'язаних із запитом користувача. Серед таких термінів з високою ймовірністю будуть такі, що матимуть якісний вплив на результати пошуку пошуковою системою при використанні цих термінів у запиті у комбінації з попередньо вказаними ключовими словами у якості нового набору ключових слів. Після уточнюючого запиту і здійснення аналізу ресурсів таким же чином, як і на першій ітерації, множина важливих термінів поповниться новими елементами, що матимуть зв'язок з об'єктом пошуку. Ітеративний процес можна продовжувати як завгодно довго, в залежності від потреб експерта. Чим більше ітерацій буде здійснено, тим глибші зв'язки вдасться виявити на кожній наступній ітерації.

ПОВНИЙ ОПИС АЛГОРИТМУ

Розглянемо реалізація алгоритму на псевдокоді:

```

procedure FindRelationsAlgo begin define Q; # Query
define S_all; # Results set define res_list;
define res_amount = 0; define continue = TRUE; input
(Q);
S_all = empty_set(); res_list = empty_list(); do
Q = normalize(Q); res_list = query_for_resources(Q
); res_amount = length(res_list);
for resource in res_list do clean(resource); define S;
define index_list; define around_set; define N;
N = 20;
S = build_term_set(resource); index_list =
query_index_list(S, Q);
around_set = build_around_set(S, index_list, N);
S_all = union_of_sets(S_all, around_set); end for
sort(S_all);
present_results(S_all);

```

```

    input(continue); if continue == TRUE define k_words;
    input(k_words);
    Q = union_of_sets(Q, k_words); end if
    while continue == TRUE;

    return S_all;
end
end procedure FindRelationsAlgo

```

Проаналізуємо найважливіші етапи і функції алгоритму. S_{all} - множина результатів, слів та термінів, які позначаються такими, що мають зв'язок до об'єкту пошуку. Елементами множини є пари <термін, кількість>. Q – запит експерта (користувача), за яким здійснюється пошук.

Функція **normalize**(Q) – функція, у якій здійснюється нормалізація запиту користувача, виділяються ключові слова, неважливі слова прибираються, при необхідності (в залежності від пошукової системи) кожен термін запиту лемматизується і стеммінгується. На вихід функція видає список ключових слів запиту.

Функція **query_for_resources**(Q) – одна з важливих функцій алгоритму, що здійснює комунікацію зі сховищем даних (пошуковою системою). Отриманий на вхід список ключових слів передається пошуковій системі, після обробки запиту пошуковою системою, функція отримує відповідь пошукової систем з списком релевантних ресурсів. Важливо зауважити, що відповідь пошукової системи може бути представлена у різних форматах (найпоширенішим є формат HTML), і дана функція повинна коректно опрацювати отриманий результат і подати на вихід лише список ресурсів.

Процедура **clean** (*resource*) – процедура, що здійснює початкову обробку ресурсу. Ресурси можуть бути представлені у різноманітних форматах (найпоширеніший – HTML), тому перед початком обробки і аналізу ресурсу його слід привести до правильного вигляду. У процедурі виділяється смислова (текстова) частина ресурсу, видаляються символи розмітки, знаки пунктуації, неважливі слова.

Функція **build_term_set**(*resource*) – функція, яка з отриманого на вхід тексту ресурсу подає на вихід множину (або список) термінів та слів.

Важливим є порядок термінів у множині. Слова в отриманій множині проіндексовані і розташовані в

такому ж порядку, в якому вони зустрічаються в оригіналі ресурсу.

Функція **query_index_list** (S , Q) – функція, яка отримавши на вхід дві множини (два списки) термінів подає на вихід список індексів входження кожного терміну з множини Q у множині S . Тобто, в отриманому на виході списку будуть міститися індекси, за якими у множині S знаходяться терміни з множини Q .

Функція **build_around_set** (S , *index_list*, N) – функція, яка отримавши на вхід множину, список індексів та значення радіусу околу N , подає на вихід множину стермінів (слів), які знаходяться в околі N кожного індексу із списку *index_list*.

Реалізація функції **build_around_set** (S , *index_list*, N) на псевдокодї:

```

function build_around_set(S, index_list, N)
begin define res_list;
for index in index_list do define i = index - N;
if i < 0 i = 0;
end if
while i < index + N AND i < length(index_list) do if i !=
index_add_to_list(res_list, S[i]);
end if
end while
end for return res_list;
end
end function build_around_set

```

В описані вище функції важливо звернути увагу на значення аргументу N . Як видно у реалізації алгоритму, аргумент ініціалізовано константою і не змінюється в процесі аналізу. Від підбору аргумента N залежить розмір результуючого списку. Надто мале N призведе до того, що кількість термінів, що потраплять у результуючу множину буде малим, і з високою ймовірністю, якість проведеного аналізу буде невисокою, оскільки велика кількість зв'язаних термінів проігнорується, не потрапивши в результуючу множину. З іншого боку, використання надто великого радіусу околу значно збільшить часові затрати роботи алгоритму, але результати суттєво не зміняться. Експериментально було визначено, що оптимальним є значення 20-35.

Функція **union_of_sets** (S_{all} , *around_set*) – функція, яка отримавши на вхід дві множини, повертає на вихід об'єднання цих множин. Варто зауважити, що

у реалізації алгоритму множина S_{all} має специфічну реалізацію. Множина S_{all} містить пари <термін, кількість>. При додаванні елемента до множини, якщо такий елемент вже існує, то у відповідній парі значення *кількість* збільшується на 1, якщо такого елемента не існує, додається нова пара <термін, 1>.

ПЕРЕВАГИ І НЕДОЛІКИ АЛГОРИТМУ, ШЛЯХИ УДОСКОНАЛЕННЯ

Описаний вище алгоритм дозволяє здійснювати ефективний пошук зв'язків між об'єктом пошуку і даними веб-сторінок. Як і кожен алгоритм у алгоритму АПЗЗ є свої переваги і недоліки.

Переваги алгоритму :

- Використання вже існуючого сховища даних. Будь-яка система, що використовує методи видобування даних і знань (Data Mining) повинна містити одним із своїх компонентів сховище даних. Сховища даних – складні інформаційні системи, розробка і підтримка яких вимагає значних часових і трудових затрат. Алгоритм АПЗЗ використовує як сховище даних будь-яку існуючу пошукову систему (або сховище даних), до яких існує доступ. Це дозволяє скоротити часові затрати і уникнути дублювання розробки уже існуючих систем.

- Мовна незалежність. Алгоритм АПЗЗ дуже мало залежить від мови, що використовується користувачем. Навіть якщо не здійснювати обробку природньої мови у процесі виконання алгоритму, результати роботи практично не змінюються. Часто обробку природніх мов здійснює сама пошукова система.

- Простота реалізації.

Недоліки алгоритму :

- Націленість на роботу з веб-сторінками. Деякою мірою алгоритм залежить від розміщення даних у ресурсі. Як сказано вище, на веб-сторінках інформація розміщується блоками і зв'язані частини інформації зазвичай знаходять недалеко одна від одної. Тому можна використовувати спосіб побудови множини важливих слів таким чином, як описано вище. Проте, інформація у інших сховищах даних може мати кардинально інше представлення, що може призвести до некоректної роботи алгоритму.

- Залежність від роботи пошукових інтернет-систем. Алгоритм залежить як від якості пошуку пошу-

кової системи так і від швидкодії. Низька релевантність пошуку спричинить низьку якість результатів. Швидкість роботи пошукової системи і час відклику впливають на час роботи всього алгоритму. Крім того, більшість пошукових інтернет систем не дозволяють здійснювати велику кількість автоматичних запитів, що, при дуже складних аналізах може призвести до неправильного результату.

- Особливості реалізації соціальних мереж та великих порталів, на яких зберігається корисна інформація, зокрема про їхніх користувачів, часто не дозволяють здійснити аналіз описаним вище методом. Часто у відповідь на запит до таких порталів отримується закодована сторінка (наприклад JavaScript код), який не піддається аналізу. З іншого боку, кожна з великих соціальних мереж надає спеціальні інструменти для роботи з даними, що розміщені у мережі. Використання таких інструментів може якісно вплинути на роботу алгоритму, особливо при пошуку зв'язків деякої особи з іншими об'єктами, особами і даними.

Шляхи удосконалення алгоритму:

- Покращення роботи з соціальними мережами. Як сказано вище, соціальні мережі – складні веб-портالي, що часто надають важливу інформацію лише з використанням певних специфічних для кожної пошукової системи інструментів. Використання таких інструментів дозволить не лише видобути множину деяких зв'язаних з об'єктом термінів, а й класифікувати її, виділити, наприклад, рік народження особи, чи її впадобання.

- Побудова дерева залежностей на кожному кроці. В описаному вище алгоритмі, кожна наступна взаємодія з пошуковою системою відкриває нові залежності. Побудова деякого дерева, на кожному рівні якого зберігатимуться нові знайдені зв'язки допоможе відповісти на питання: як саме, або через що один об'єкт має зв'язок з іншим.

ВИСНОВКИ

Алгоритм пошуку зв'язків і залежностей на веб-сторінках – ефективний метод виявлення прихованих даних і зв'язків. Описаний вище алгоритм дозволяє здійснювати інтелектуальний пошук зв'язків у мережі інтернет серед даних, що знаходяться у вільному доступі.

Основними перевагами описаного алгоритму є:

- Використання вже існуючого сховища даних
- Мовна незалежність.
- Простота реалізації.

Реалізація алгоритму не вимагає трудових і часових затрат на побудову сховища даних. Результати роботи практично не залежать від мови, якою здійснюється аналіз, користується експерт.

З іншого боку, до недоліків можна віднести:

- Алгоритм націлений на роботи з даними веб-сторінок, представлених у форматі HTML;
- Залежить від роботи пошукової системи;
- Не враховує особливостей реалізації соціальних мереж.

Алгоритм пошуку зв'язків і залежностей може мати широке практичне застосування у таких галузях:

- Пошук осіб правоохоронними органами;
- Пошук інформації про особу HR-відділами підприємств;
- Пошук зв'язків особи з іншими особами, подіями, об'єктами;
- Пошук зв'язків події, організації з іншими подіями і особами, тощо.

Використання такого алгоритму у автоматизованих системах значною мірою вплине на якість пошуку, адже на вихід отримуються не просто «сирі» веб-сторінки, а об'єкти, зв'язані з об'єктом пошуку.

ЛІТЕРАТУРА:

1. Data Mining. A Knowledge Discovery Approach / [Krystof J. Cios, Witold Pedrych, Roman W. Swiniarski, Lukasz A. Kurgan] – San Diego, USA. : “Springer”, 2007 – 606с.
2. Data Mining and Knowledge Discovery Technologies / David Tanar – New York, USA : “IGI publishing”, 2007 – 369с.
3. Data Mining Patterns: New Methods and Applications / [Pascal Poncelet, Maguelonne Tesseire, Florent Masseglia] – New York, USA: “IGI publishing”, 2008 – 307с.
4. 2008 – 307с.
5. Data Mining with Computational Intelligence / Lipo Wang, Xiuju Fu – Berlin, Germany: “Springer”, 2005 – 276с.
6. Data Mining Patterns: New Methods and Applications / [Pascal Poncelet, Maguelonne Tesseire, Florent Masseglia] – New York, USA: “IGI publishing”, 2008 – 307с.
7. Data Mining. Practical Machine Learning Tools and Techniques. Second Edition / Ian H. Witten, Eibe Frank – San-Francisco, USA: “Elsevier”, 2009 – 525с.
8. Learning Python, Fifth Edition / Mark Lutz – Sebastopol, USA : “O’Reilly Media Inc.”, 2013 – 1540с.
9. Making Sense of Data. A Practical Guide to Exploratory Data Analysis and Data Mining / Glenn J. Myatt – Canada: “John Wiley & Sons”, 2007 – 280с.
10. Pattern Recognition Algorithms for Data Mining. Scalability, Knowledge Discovery and Soft Granular Computing / Sankar K. Pal, Pabitra Mitra – Boca Raton, USA : “CRC Press”, 2004 – 200с.
11. Бібліографія : [Електронний ресурс] // Вікіпедія – вільна енциклопедія – Режим доступу : <http://uk.wikipedia.org/wiki/Python>

*Рецензент: д. ф-м.н., проф. О.І. Провотар
Київський національний університет імені Тараса Шевченка*



МОДЕЛЮВАННЯ КОРПОРАТИВНОЇ КОМП'ЮТЕРНОЇ МЕРЕЖІ

УДК 004.94

ЛЕПА Евгений Владимирович

к.т.н., доцент кафедри інформаційних технологій Херсонського національного технічного університета.

Научные интересы: системы принятия и поддержки решений, технологии интеллектуального анализа данных, моделирование.

e-mail: e.lepa@rambler.ru

ВСТУП

Сьогодні, в умовах постійного пошуку, спрямованого на поліпшення виробничих процесів, і швидкого розвитку обчислювальної техніки і прикладного програмного забезпечення, складність телекомунікаційних систем різко зростає. З'являються нові концепції, технології, архітектурні рішення побудови телекомунікаційних систем. Здійснюється перехід від статичної, централізованої структури телекомунікаційних систем до динамічної, гнучкої, що заснована на розподілі систем отримання і обробки інформації [1,6].

Сучасна комп'ютерна система – це складний технічний об'єкт. Тому розробка комп'ютерної системи і її впровадження неможливі без попереднього моделювання і дослідження моделі. Причому часто модель комп'ютерної мережі або модель телекомунікаційної системи функціонує паралельно мережі або системи, що реально працює, як її віртуальний образ. Такий підхід дозволяє прогнозувати поведінку мережі або системи в найближчому або віддаленому майбутньому, проводити якісну модернізацію [2,4].

На сучасному ринку інформаційних технологій існують спеціальні програмні системи, що орієнтовані на моделювання та дослідження комп'ютерних мереж. Програмні системи моделювання мереж – інструмент, який може стати в нагоді будь-якому адміністратору корпоративної мережі, особливо при проектуванні нової мережі або внесенні кардинальних змін в уже існуючу. Продукти цієї категорії дозволяють перевірити наслідки впровадження тих чи інших рішень ще до придбання обладнання: відповідність поточним вимо-

гам і вимогам найближчого часу, можливість і вартість подальшого розвитку.

Проте тема моделювання та оптимізації продуктивності мережі за допомогою програмних систем імітаційного моделювання у вітчизняних дослідженнях має фрагментарний характер.

МЕТА Й ЗАВДАННЯ ДОСЛІДЖЕННЯ

Поставлене завдання проектування корпоративної комп'ютерної мережі. Необхідно виконати імітаційне моделювання з метою визначення структури мережі, вибору її компонентів і визначення основних характеристик. Для моделювання використовується система імітаційного моделювання Riverbed Modeler [3,5]. Ця система є однією з найбільш використовуваних для моделювання комп'ютерних мереж. Крім того, вона вільно поширюється у вигляді академічної версії, що робить її доступною для широкого кола користувачів. Вона може бути використана студентами при проектуванні й дослідженні комп'ютерних мереж і систем [10].

ДОСЛІДЖЕННЯ ТА АНАЛІЗ РЕЗУЛЬТАТІВ

Підприємство має центральний офіс в Одесі та філії в Херсоні і Дніпрі. На кожній точці вже існує локальна мережа, побудована за стандартом 10BaseT, з виходом в Internet. Необхідно об'єднати всі точки в одну мережу та організувати доступ до серверів, що розташовані в Києві. Замовник планує орендувати два сервери, що розташовані в київському дата центрі, доступ до яких будемо організовувати через шифрований EoIP тунель.

Перш за все побудуємо концептуальну модель майбутньої мережі (рисунок 1) [7,9].

Основними складовими частинами є: сервер, маршрутизатор, комутатор, робоча станція, SIP-телефон та GSM-шлюз. Для зручності будемо об'єднувати робочі станції в групи. Насамперед побудуємо таблицю відповідності обладнання та модулів в Riverbed Modeler (таблиця 1).

Результатом побудови є імітаційна модель, що складається з чотирьох субмереж, що поєднані лініями зв'язків, які моделюють роботу відділень та віддаленого серверу.

Опис процесів, що моделюються, проходить шляхом конфігурації модуля Application Definitions, який визначає програми та групи додатків. Крім того, він описує, як створювати мітки вузлів і портів.

Таблиця 1

Відповідність обладнання та модулів в Riverbed Modeler

Найменування обладнання	Модуль
Сервер	ppp_server
Маршрутизатор	atm4_ethernet2_slip8_gtwy
Комутатор	ethernet16_switch
Робоча станція	ethernet_wkstn_adv
Мережа 10BaseT	10BaseT_LAN
Мережа 100BaseT	100BaseT_LAN
SIP-телефон	ip_phone
GSM-шлюз	sip_server

Для моделювання специфічних додатків, яких немає в стандартному переліку використовується модуль Task Definition, що дозволяє будувати користувацькі завдання на базі яких конфігуруються додатки.

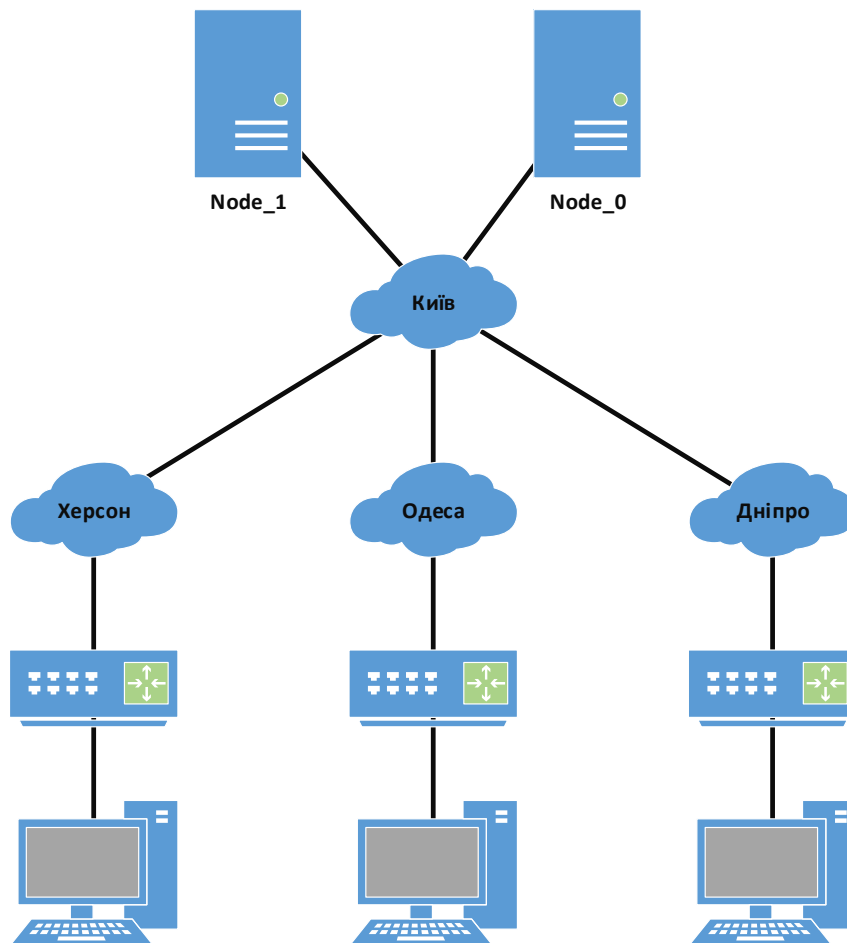


Рисунок 1 – Концептуальна модель мережі

Profile Definitions об'єднує додатки в профіля, що описують роботу модулів. Модель включає 4 профіля:

- Engineer_profile;
- E-commerce_profile;
- Oper_profile;
- Communication_profile.

Розглянемо детальніше структуру кожного профілю.

Engineer_profile описує модель використання робочої станції інженером.

Профіль включає 5 додатків, серед яких:

1. Web Browsing (Heavy);
2. Email (Light);
3. Telnet Session (Heavy);
4. File Transfer (Heavy);
5. Database Access (Heavy).

E-commerce_profile описує модель використання робочої станції офісним працівником. Профіль включає 4 додатки, серед яких:

1. Web Browsing (Heavy);
2. Email (Light);
3. File Print (Light);
4. Report_downloading_ap.

Oper_profile описує модель використання робочої станції офісним працівником. Профіль включає 5 додатків, серед яких:

- Web Browsing (Light);
- Email (Light);
- Video Conferencing (Light);
- Voice over IP call;
- Database Access (Light).

Субмережа Херсонського відділення складається з 5 блоків, а саме: головного роутера, міжповерхового комутатора, виділеного типового PC, та двох мереж, що агрегують по 10 робочих станцій.

Субмережа Дніпровського відділення складається з 4 блоків: головного роутера, міжповерхового комутатора та двох мереж, що агрегують 10 та 30 робочих станцій відповідно.

До складу субмережі Одеського відділення входять 2 міжповерхових комутатори, маршрутизатор, 3 блоки 10BaseT_LAN по 20 та 1 блок на 10 робочих станцій.

Два сервери, що розташовані в Київському Дата-центрі.

Після закінчення первинної побудови та конфігурації моделі можна переходити до перших прогонів.

Найбільш корисна характеристика мережі - це час доставки повідомлення. Це означає, що в першу чергу нас цікавлять затримки на різних ділянках. Будемо вважати, що сервера зовсім не затримують пакети при обробці, так як насправді час на перенаправлення був б мізерно малим. Тому, час на доставку кожного пакету буде складатися з часу пересилань між вузлами мережі і часу очікування в чергах[8]. Оцінимо завантаження серверів у вихідному сценарії (рисунок 2).

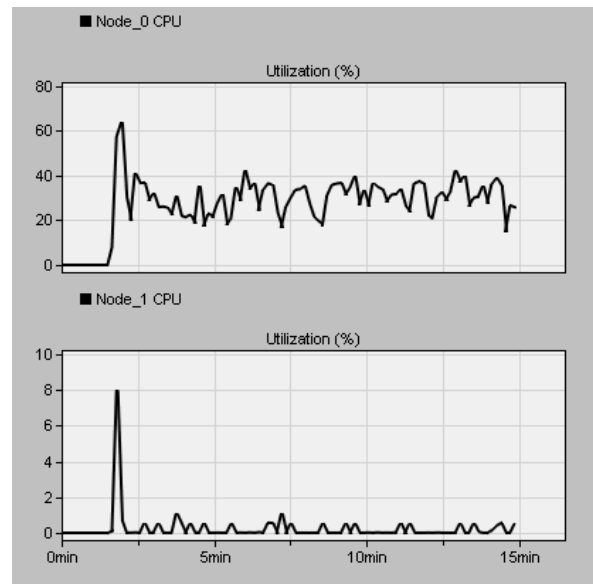


Рисунок 2 - Завантаження CPU для Node_0 та Node_1

Навантаження на Node_0 в середньому складає 35%, максимальна відмітка склала 63%. Для Node_1 результати склали 1% та 8% відповідно. З отриманих даних видно, що навантаження на Node_1 не суттєво порівняно з Node_0. З врахуванням наявності вільної розрахункової здатності, можна перенести всі операції на перший сервер, для цього переконфігуруємо список «Supported Services» для Node_0.

Після прогону нового сценарію отримано результати, що відображені на рисунку 3.

З отриманих даних видно, що навантаження на сервер суттєво не зросло. Тому всі процеси переносимо на Node_0, а сервер Node_1 будемо використовувати в якості резервного.

Наступним параметром, який необхідно оцінити є навантаження на канал передачі даних (КПД) зі сторо-

ни серверу Node_0. Для мережі, що побудована за стандартом 10BaseT, в момент пікового навантаження спостерігаємо повне завантаження каналу на прийом і відповідно збільшення затримки до 1 с.

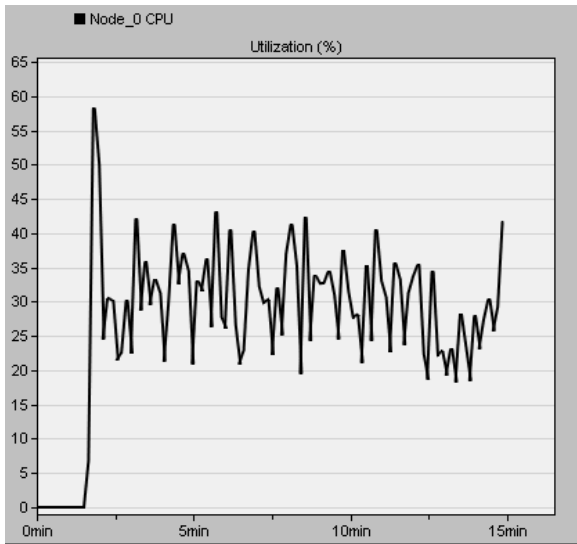


Рисунок 3 - Результуюче навантаження CPU для Node_0

Для підвищення пропускної здатності мережі та зменшення затримок проведемо модернізацію моделі за стандартом 100BaseT та розширимо пропускну здатність каналу Internet до 30Мбіт/с.

З отриманих результатів видно, що час повного завантаження каналу зменшився з 5 до 2 хвилин, а час максимальної затримки зменшився до 0,35 с.

При збільшенні пропускної здатності каналу Internet до 45Мбіт/с максимальна затримка зменшилась до 0,25 с, а час повного завантаження приблизно до 1 хв. Розширимо канал Internet до 100Мбіт/с.

Очікувано, що при розширенні каналу передачі даних до 100Мбіт/с зменшилась як максимальна затримка так і час повного пікового завантаження каналу. Об'єднані графіки навантаження на КПД представлені на рисунках 4 і 5. На основі аналізу отриманих результатів, можна зробити висновок, що оптимальною по відношенню затримка / вартість буде ширина каналу Internet в 45Мбіт/с.

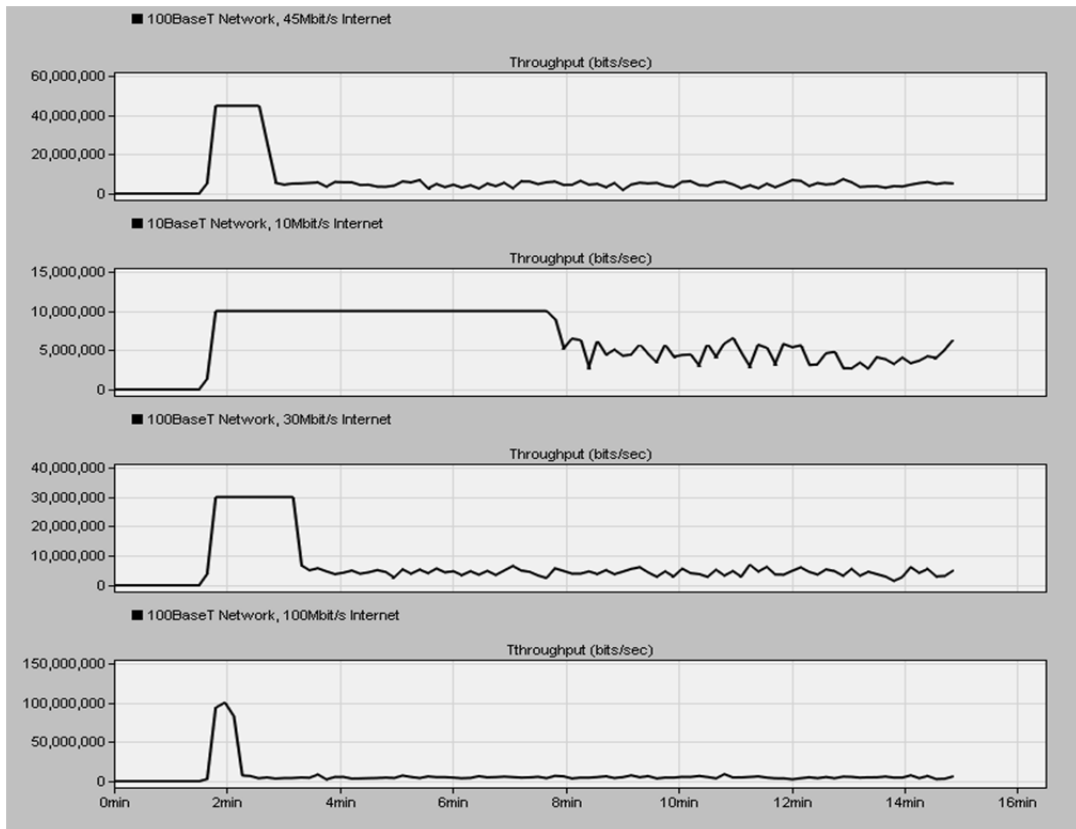


Рисунок 4 - Об'єднаний графік навантаження на порт Node_0 (прийом)

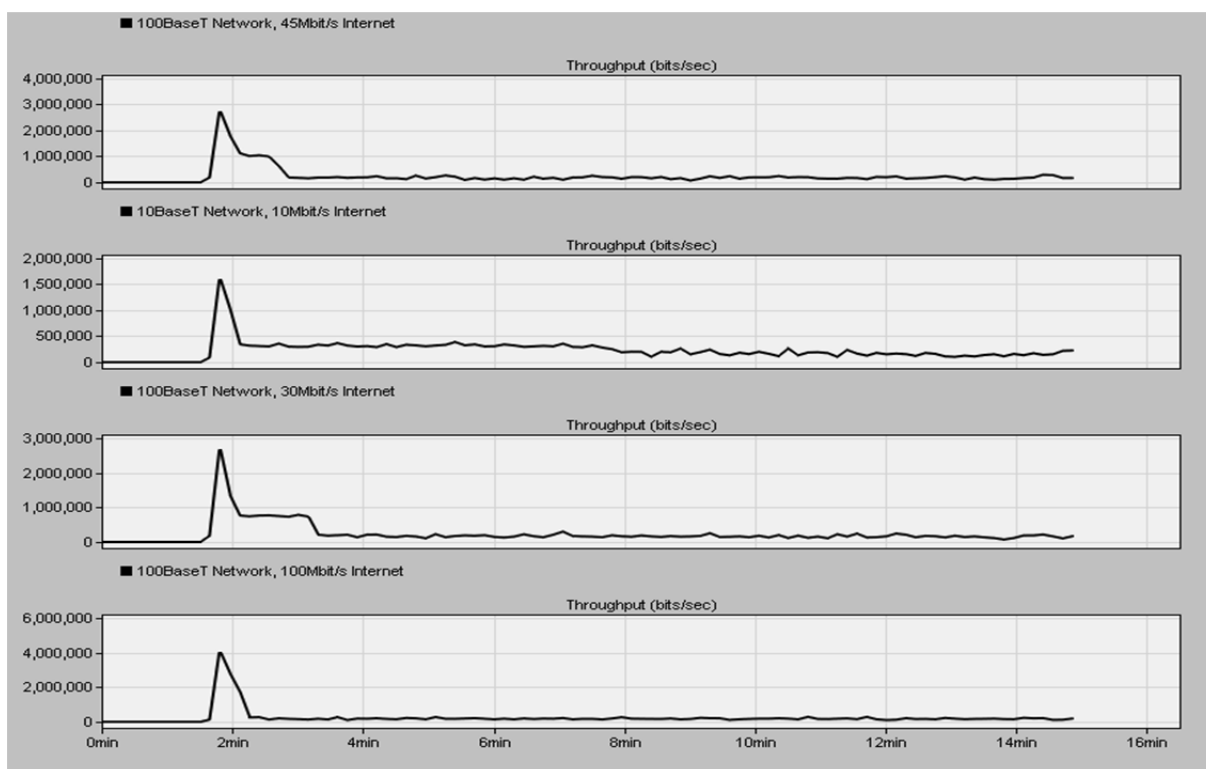
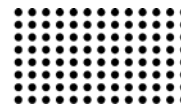


Рисунок 5 - Об'єднаний графік навантаження на порт Node_0 (передача)

Аналогічним чином може бути проаналізовано оптимальну пропускну здатність каналу Internet для кожного з відділень.

ВИСНОВКИ

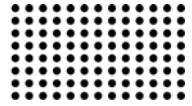
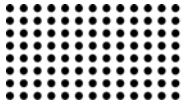
Розроблена структура й виконане моделювання корпоративної комп'ютерної мережі з використанням

системи моделювання Riverbed Modeler. Досліджена пропускну здатність мережі, виконаної по стандартах 10BaseT і 100BaseT. Визначена ширина каналу Internet для оптимального співвідношення затримка/вартість.

ЛІТЕРАТУРА:

1. Semyonov YU.A. Telekommunikacionnye tekhnologii. – M.: GNC ITEHF. – 2004. – 532 s.
2. Olifer N.A., Olifer V.G. Sredstva analiza i optimizacii lokal'nyh setej. Centr informacionnyh tekhnologij, 1998. – 261 s.
3. Veb-sajt Riverbed Modeler. – [Elektronnij resurs]. – Rezhim dostupu: <https://www.riverbed.com/gb/products/steelcentral/steelcentral-riverbed-modeler.html>
4. Klaus Wehrle, James Gross. Modeling and Tools for Network Simulation. Hardcover: 2010, – 256 p.
6. Kel'ton V., Lou A. Imitacionnoe modelirovanie. Klassika CS. 3-e izd. – SPb.: Piter; Kiev: Izdatel'skaya gruppa BHV, 2004. – 847 s.: il.
7. Sovetov B.YA., YAKovlev S.A. Modelirovanie sistem: Uchebnik dlya vuzov. – 4- e izd., ster. – M.: Vysshaya shkola, 2005. – 343 s.: il.
8. Stolings V. Sovremennye komp'yuternye seti. – SPb.: Piter, 2003. – 783 s.: il.
9. Sistema modelirovaniya Opnet [Elektronnij resurs] : oficial'nyj sajt proizvoditelya. — Rezhim dostupa : <http://www.opnet.com>.
10. Strogalev V. P., Tolkacheva I. O. Imitacionnoe modelirovanie. - MGТУ im. Baumana, 2008. – 378 s.: il.
11. G. V. Trusov Vvedenie v matematicheskoe modelirovanie M.: Universitetskaya kniga, Logos, 2007. – 354 s.: il

Рецензент: д.т.н., проф. В.Ф. Коваленко
Херсонський національний технічний університет



РОЗРОБКА ЗАСОБУ ПРОЕКТУВАННЯ ВИСОКОНАВАНТАЖЕНИХ РЕЛЯЦІЙНИХ СИСТЕМ ЗБЕРІГАННЯ ДАНИХ: ОПТИМІЗАЦІЯ СТРУКТУРИ ТА ЗАПИТІВ SQL

УДК 004.652.4 : 004.4'22

КОЛЕСНИК Людмила Володимирівна

кандидат технічних наук, доцент, кафедра системотехніки, Харківський національний університет радіоелектроніки

Наукові інтереси: проектування інформаційних систем, управління і моделювання інформаційних ресурсів.

e-mail: liudmyla.kolesnyk@nure.ua, kolesnik_l_v@ukr.net

КИРИЧЕНКО Наталія Андріївна

студентка, кафедра системотехніки, Харківський національний університет радіоелектроніки

Наукові інтереси: проектування систем зберігання даних.

e-mail: nataliia.kyrychenko@nure.ua

КОСТОГЛОТ Ігор Валерійович

студент, кафедра системотехніки, Харківський національний університет радіоелектроніки

Наукові інтереси: інформаційні системи, web-програмування.

e-mail: ihor.kostohlot@nure.ua

ВСТУП

У сучасному світі кожне програмне забезпечення (ПЗ), веб-сайт і в цілому практично кожна система працює з великими об'ємами інформації та даних, які вимагають ресурсів для зберігання, обробки та редагування. Нині виконання цих цілей покладено на спеціально призначені для цього засоби зберігання та обробки даних, що мають усталену назву систем управління базами даних (СУБД), різновидів яких у наш час існує вже кілька десятків. Дана стаття буде присвячена одній конкретній СУБД під назвою MySQL.

АНАЛІЗ ЛІТЕРАТУРНИХ ДЖЕРЕЛ

ТА ПОСТАНОВКА ПРОБЛЕМИ

СУБД MySQL є реляційною СУБД. Це означає те, що дана СУБД заснована на реляційних залежностях, а саме: «відношення», «залежність», «зв'язок». Простіше кажучи, в даній СУБД всі дані представляються не суцільним і безупорядоченим набором інформації, а

мають чітку залежність один від одного і від тієї предметної області, для якої була розроблена база даних (БД) в СУБД MySQL. СУБД MySQL має досить великий набір інструментів для створення та управління БД, підтримує велику кількість форматів таблиць БД, як з підтримкою повнотекстового пошуку (MyISAM, InnoDB), так і з підтримкою транзакцій та зовнішніх ключів (InnoDB). Також СУБД MySQL є вільно поширюваним ПЗ, що робить дану СУБД ще більш популярною [1].

Однак багатофункціональність та гнучкість СУБД MySQL також є і мінусом, оскільки найчастіше однозначного рішення у розумінні того, який тип таблиці вибрати, які індекси проставити, як оптимально зв'язати таблиці в БД – немає. Первісне планування структури бази даних здебільшого не враховує великої кількості необхідних таблиць і атрибутів та через це, як наслідок, спроектована база даних має більш громіздкий вигляд і займає більше часу на розробку.

СУБД MySQL, як і багато інших спеціалізованих засобів, має значну кількість налаштувань і особливостей, та якщо на початку розроблена БД працює досить швидко, то буде помилкою вважати, що вона збереже свою швидкість в режимі високої завантаженості та зростанні її об'ємів. Крім налаштувань самої СУБД та структури БД, важливою деталлю є безпосередньо самі SQL запити до бази даних, які також можуть бути оптимальні для однієї структури [2], налаштувань та навантаженості БД і можуть бути абсолютно не оптимальними при внесенні коректувань в структуру, налаштування, а також при підвищенні / зниженні навантаження на БД.

МЕТА І ЗАВДАННЯ ДОСЛІДЖЕННЯ

Метою роботи є розробка програмного засобу для автоматичного проектування / перепроектування високонавантаженої реляційної системи зберігання даних за допомогою оптимізації SQL запитів до БД [3], а також безпосередньо структури БД. Досягнення поставленої мети передбачає вирішення наступних завдань:

- дослідження методів оптимізації структури бази даних;
- дослідження методів оптимізації швидкості виконання SQL запитів;
- розробка програмного засобу для автоматизації процесу оптимізації бази даних і SQL запитів.

ПЕРЕДІСТОРІЯ ВИНИКНЕННЯ ІДЕЇ

Ідеєю для розробки даного програмного засобу виявився власний досвід роботи над веб-проектом, який за кілька років від низьковідвідуваного ресурсу перейшов в стадію високонавантаженого та масштабного. Вихідний код проекту був розроблений самостійно, використовуючи мову PHP в якості серверної частини і JS на фронтенді, в якості СУБД була обрана MySQL. За рахунок оптимально написаного коду проект спочатку мав високу швидкість роботи, час генерації однієї сторінки займав ~ 0.002-0.005 секунд, проте по мірі зростання кількості даних в БД та кількості запитів до БД веб-ресурс почав досить сильно гальмувати. Після проведення нескладних тестів було виявлено, що причина є саме в БД. Запити, які виконувалися за ~ 0.0003 секунди, тепер виконувалися

за ~ 0.04 секунди. Першим етапом оптимізації послугувало те, що були переглянуті та поліпшені SQL запити, однак належного ефекту без внесення змін до структури бази даних це не дало. Другим етапом послугувало те, що спільно зі зміною запитів SQL також була змінена і структура БД – великі таблиці були розбиті на більш дрібні, кількість зв'язків у таблиці зросла, проте це хоча і дало бажаний ефект щодо прискорення, але ненадовго. При подальшому зростанні проекту і, відповідно, його БД, потрібна була чергова оптимізація. Цього разу до вищевказаних дій було додано дублювання часто використовуваних даних у кілька таблиць, були застосовані індекси для багатьох стовпців, а також були оптимізовані формати та розмірності атрибутів даних в таблиці. Дані дії дозволили відновити початкову швидкість роботи бази даних. Також, заради експерименту, таблиці в БД були переведені з типу InnoDB до типу MyISAM, що також дозволило виграти додаткову швидкість у обробці SQL запитів, хоча спочатку саме з типом InnoDB запити виконувалися значно швидше, ніж з типом даних MyISAM.

Далі буде детально розглянута послідовність дій для оптимізації бази даних з рекомендаціями, які були отримані на особистому досвіді.

ВИБІР ТИПУ (ENGINE) БАЗИ ДАНИХ

На першому етапі оптимізації БД необхідно визначитися з вибором типу бази даних [5]. На даний момент існує два основних типи – це InnoDB [6] і MyISAM. Не будемо вдаватися в тонкощі відмінностей типів баз даних та відразу викладемо коротку суть кожної з них.

Тип бази даних MyISAM бажано обирати виключно у тому випадку, якщо немає необхідності реалізації транзакцій на рівні бази даних, немає необхідності організації зовнішніх ключів, а також в проекті переважає один з видів доступу – читання або запис, тобто новинний ресурс або, наприклад, логування. Також MyISAM менше займає місця на диску, однак не підтримує мультибайтове кодування.

У всіх інших випадках бажано обирати тип бази даних InnoDB, а також даний тип необхідно вибирати тоді, коли можна пожертвувати швидкістю виконання операції в обмін на підвищену безпеку даних,

наприклад, в додатках фінансового характеру та строгої звітності.

ПРОЕКТУВАННЯ СТРУКТУРИ

Коли визначилися з типом бази даних, далі необхідно безпосередньо спроектувати її структуру. При проектуванні структури необхідно завжди стежити та дотримуватися того, щоб структура відповідала третій нормальній формі [4], однак це не завжди буває доцільним. У деяких випадках можливо зменшити навантаження на базу даних за рахунок перенесення деяких даних з бази в мовні файли ПЗ, що розробляється. Зараз ми докладніше опишемо, що конкретно мається на увазі.

Припустимо, що є база даних у третій нормальній формі і є таблиця з логіном користувача, а також таблиця з коментарями.

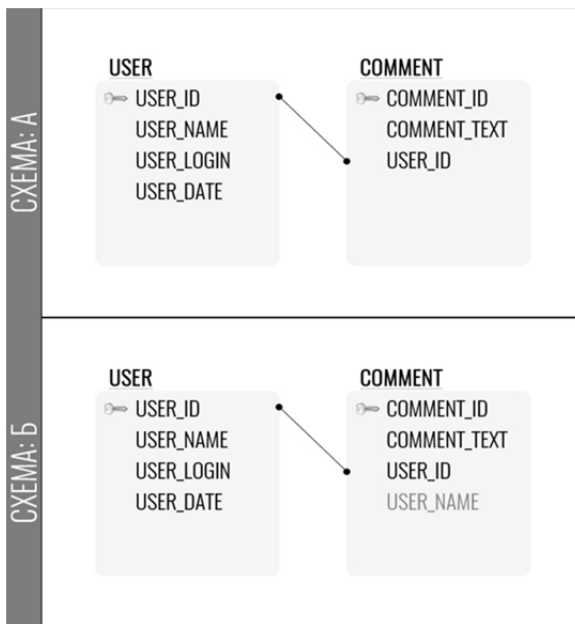


Рис. 1. Схема БД з таблицями користувачів і коментарів

На рисунку 1 зображені два варіанти схеми описаної БД. Припустимо, що спільно з текстом коментарів необхідно також виводити ім'я користувача. Як видно на схемах, зображених на рисунку 1, схема А містить виключно поле user_id, ID користувача, який залишив коментар, в той час, як схема Б, крім поля user_id, містить також і поле user_name, з ім'ям користувача.

У схемі А позначено, що у веб-додатку необхідно буде виконувати 1 складний запит з використанням

JOIN або виконувати 2 простих запити – перший запит на вибірку коментарів, другий запит – на вибірку імен авторів коментарів. Даний метод дозволяє зменшити розмір БД за рахунок того, що імена авторів не будуть дублюватися, проте даний варіант становить значно більше навантаження на базу даних.

Схема Б дозволяє за 1 запит зробити вибірку всіх необхідних коментарів відразу ж з іменами користувачів. В даному варіанті отримано незначне зростання розміру бази даних, однак швидкість виконання запитів та роботи бази даних значно виграє перед варіантом, зображеним на схемі А.

Тепер припустимо, що є база даних у третій нормальній формі та є таблиця з інформацією про користувача.

На рисунку 2 зображено дві схеми з варіантами зберігання інформації про стать користувачів. На схемі А зображена таблиця користувачів з полем gender типу TINYINT (1) та додаткова таблиця, в якій буде всього три сутності – «не вказано», «чоловік» і «жінка». На схемі Б зображена одна таблиця, де поле gender представлено типом ENUM з варіантами значень «0», «1», «2». У чому ж відмінності в даних схемах?

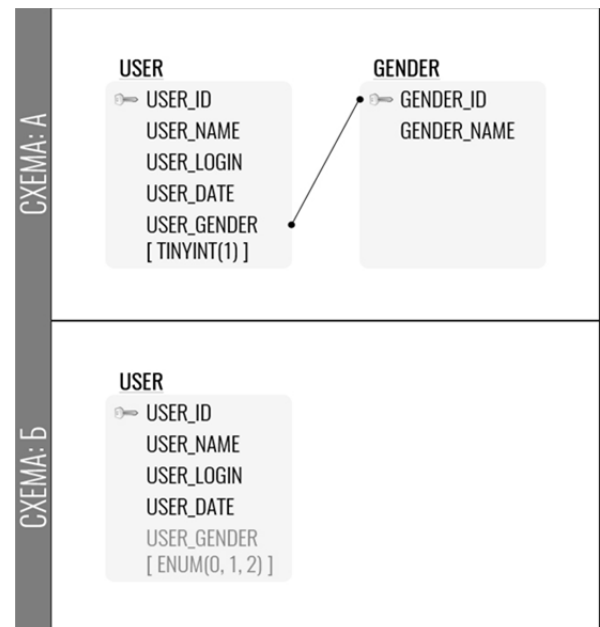


Рис. 2. Схема бази даних з таблицею інформації про користувача

Як і у попередньому випадку, на схемі А позначено, що у веб-додатку необхідно буде виконувати або 1

складний запит з використанням JOIN, або 2 простих – перший запит на вибірку самих користувачів, другий запит на вибірку варіантів значення поля gender користувачів за їх ID. Дана маніпуляція аналогічно дозволяє зменшити розмір бази даних за рахунок того, що значення поля gender не будуть дублюватися, проте у підсумку даний варіант становить значно більше навантаження на базу даних.

У схемі Б буде виконуватися усього один запит до бази даних. У результаті буде отримана вибірка, де у полі gender буде міститися одне зі значень – «0», «1» або «2». У такому випадку вже безпосередньо у ПЗ повинно бути чітко зазначено, що значення «0» відповідає значенню «не вказано», «1» – «чоловік» та «2» – «жінка».

Також можливий варіант, коли дані все ж важливо зберігати спочатку саме в базі даних, в цьому випадку можливе використання схеми А, але з кешуванням результатів вибірки з таблиці gender_table. В даному випадку ми отримуємо при першому запиті до ПЗ роботу за схемою А, а при всіх наступних – за схемою Б, оскільки значення в даному випадку вже будуть взяті не з бази даних, а з кешу.

Одним з важливих параметрів є обладнання, особливо при використанні кешування. Найбільш оптимальним буде використання кешування на серверах та пристроях з FLASH-пам'яттю та / або SSD-дисками.

РОЗБИТТЯ ГРОМІЗДКИХ ТАБЛИЦЬ

Наступний етап полягає у тому, що необхідно врахувати максимально можливий обсяг та розмір таблиці у базі даних [7]. Виключно з особистого досвіду можна зробити висновок, що найбільш оптимальне число записів у одній таблиці не повинно перевищувати 100 000 записів, при цьому обсяг таблиці не повинен перевищувати 100 Mb. Однак дані значення можуть відрізнятися в залежності від потужності обладнання, частоти запитів до таблиці, а також в залежності від того, наскільки важлива висока швидкість виконання запиту. Якщо час відповіді на запит немає великого значення або ж запити до даної таблиці виконуються раз на добу або навіть рідше, тоді даний етап оптимізації для поточної таблиці можна пропустити.

Однак, якщо важливо щоб запит виконувався максимально швидко, а таблиця може мати як 1 млн, так і більш записів або обсяг в кілька гігабайт, то вкрай рекомендовано її розділити. Розділити громіздку таблицю, котра зображена на рисунку 3 (А), можливо на логічні частини, наприклад – за країнами, як зображено на рисунку 3 (Б), за гендерною приналежністю, за інтересами, або за значеннями індексів «від ... до».

При поділі громіздких таблиць виникає необхідність використовувати додаткову таблицю з покажчиками на таблиці, що, в свою чергу, вимагає виконання додаткових запитів до бази даних, тому не слід розбивати даним чином таблиці, в яких не передбачається значний обсяг даних. Якщо ж в таблиці буде досить громіздкий обсяг даних, то навіть з урахуванням додаткових запитів швидкість отримання підсумкової вибірки буде набагато швидше, ніж у випадку одного запиту до громіздкої бази даних.

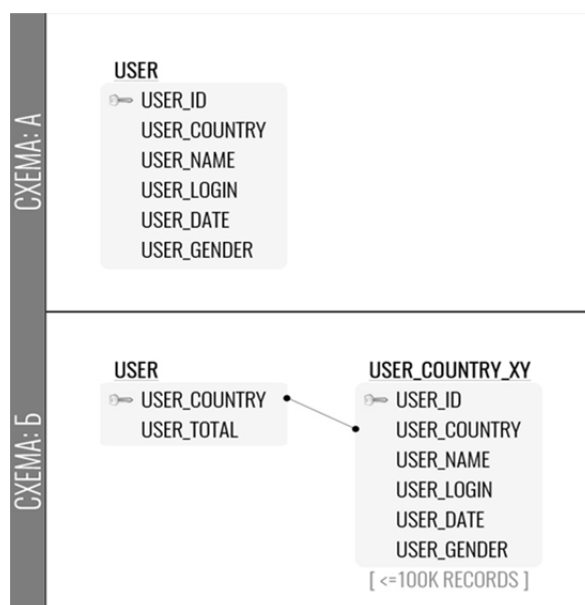


Рис. 3. Приклад поділу громіздких таблиць

ЗНАЧЕННЯ ТИПІВ ТА РОЗМІРНОСТЕЙ ПОЛІВ ДАНИХ (АТРИБУТІВ)

Кожне поле (атрибут) в таблиці бази даних повинен мати свій тип даних та при цьому розмірність поля повинна бути мінімальною. Розглянемо докладніше, що це означає.

Припустимо, що необхідно зберігати у базі даних інформацію про ім'я користувача та дату його реєстрації на сайті в UNIX форматі. Отже, ім'я користувача – це текст, дата в UNIX форматі, логічно, теж могла бути вказана у вигляді тексту. В результаті для імені користувача та дати можливо вказати тип даних TEXT, як показано на рисунку 4 (А). Однак це не є оптимальним, оскільки на тип TEXT база даних буде резервувати значно більше місця, ніж необхідно для зберігання даних про ім'я користувача та дату реєстрації. До того ж, ім'я користувача не буде містити значних обсягів тексту, для яких передбачено тип TEXT, а дата реєстрації в UNIX форматі і зовсім не буде перевищувати розмірності в 11 знаків.

Таким чином, можливо припустити, що у полі «ім'я користувача» може бути максимум 18 символів, відповідно для зберігання інформації у даному полі нам цілком буде досить типу VARCHAR з розмірністю у 18 символів. Поле «дата реєстрації», як вже було описано раніше, може містити усього 10-11 знаків. До того ж, дане поле числове та невід'ємне, а також, оскільки дата реєстрації нині не може бути раніше 1 січня 1970 року, то для зберігання дати реєстрації буде досить типу INT з розмірністю в 11 знаків та атрибутом unsigned. Атрибут unsigned означає, що значення поля може бути виключно додатним числом. Таблиця з новими розмірностями полів та атрибутами зображена на рисунку 4 (Б).

Вибір оптимального типу та розмірності полів даних є дуже важливим у проектуванні структури БД [8]. Чим більш оптимально буде підібраний тип даних та розмірність поля, тим менше буде у підсумку обсяг БД, а відповідно і швидкість роботи БД буде значно вище. У СУБД MySQL, як і в багатьох інших СУБД, існує оптимізація на рівні ядра для роботи з відповідними типами даних. Це означає те, що записати / зчитати числові дані з поля з числовим типом даних СУБД буде простіше, ніж з поля, наприклад, з текстовим типом даних.

ЗАСТОСУВАННЯ ІНДЕКСІВ ДО ПОЛІВ ДАНИХ (АТРИБУТІВ)

Проставлення індексів до полів даних є невід'ємним етапом в оптимізації БД. При створенні будь-якої таблиці в базі даних, незалежно від її призначення, рекомендується використовувати хоча б один індекс – індекс на поле з порядковим номером запису в таблиці, найчастіше це поле ID, який називається PRIMARY_KEY з атрибутом auto_increment [9]. PRIMARY_KEY – це первинний ключ, за яким СУБД буде ідентифікувати запис у таблиці БД, а атрибут auto_increment означає, що запис в це поле буде проводитися автоматично та кожне нове значення буде на одиницю більше попереднього, тобто це буде лічильник записів у таблиці, який буде автоматично заповнюватися.

Крім PRIMARY_KEY існують ще інші типи індексів, такі як UNIQUE та INDEX. Індекс UNIQUE означає, що дані у полі, до якого застосовано даний індекс, не можуть повторюватися, тобто вони завжди унікальні. Індекс INDEX дозволяє індексувати дані по типу бінарного дерева, що, в свою чергу, може дозволити прискорити швидкість виконання запитів більш ніж у два рази.

Однак слід зазначити той факт, що надлишок індексів, як і їх нестача, однаково погано позначаються на швидкості виконання запитів до бази даних. Також не рекомендується застосовувати індекси до полів, дані в яких часто змінюються, оскільки після кожної зміни такого поля СУБД буде необхідно перебудувати індекс спочатку. Перебудувати індекс СУБД буде потрібно і у разі видалення \ додавання нового запису у таблицю. У цьому випадку СУБД потрібно перебудувати індекс не тільки по одному полю, а по всіх полях, до яких застосовано індекс.

СХЕМА: А	Полі	Тип	Сравнение	Атрибути	
	user_id	int(10)		UNSIGNED	
	user_name	text	utf8_general_ci		
	user_date	text	utf8_general_ci		
				...	
СХЕМА: Б	Полі	Null	По умолчанию	Дополнительно	
	user_id	Her	Her	AUTO_INCREMENT	
	user_name	Her	Her		
	user_date	Her	Her		
	Полі	Тип	Сравнение		
	user_id	int(10)			
	user_name	varchar(18)	utf8_general_ci		
	user_date	int(11)			
				...	
	Полі	Атрибути	Null	По умолчанию	Дополнительно
	user_id	UNSIGNED	Her	Her	AUTO_INCREMENT
	user_name		Her		
	user_date	UNSIGNED	Her	0	

Рис. 4. Типи та розмірності полів даних (атрибутів)

ОПТИМІЗАЦІЯ SQL-ЗАПИТІВ

Після того, як структура бази даних була оптимізована, можна переходити до наступного етапу – на етап оптимізації SQL запитів. Відразу потрібно зазначити кілька моментів, яких бажано уникати при написанні SQL запитів, а саме – використовувати якомога менше:

- вкладених підзапитів;
- операторів JOIN та інших операторів конкатенації;
- функції групування на великих таблицях, наприклад таких, як COUNT();
- повних вибірок рядків, тобто не використовувати оператор *;
- заміників імен стовпців.

Також, для отримання оптимальної швидкості виконання SQL запиту необхідно пам'ятати, що порядок полів у SQL запиті бажано використовувати відповідно до порядку полів у таблиці баз даних [10]. У обов'язковому порядку рекомендується використовувати той порядок полів в SQL запитах, який був при призначенні індексів. На рисунку 5 зображено приклад з неправильним порядком полів у SQL запиті (А) та з правильним порядком полів у запиті, без урахування індексів (Б) та з урахуванням індексів (В).

Якщо все ж таки довелось використати оператор JOIN, слід пам'ятати, що порядок з'єднання таблиць повинен бути таким, щоб пошук відповідного індексу проводився у приєднуваній таблиці, тобто, FROM ГОЛОВНА LEFT JOIN ПРИЄДНУВАНА ON ПРИЄДНУВАНА.ID = ГОЛОВНА.SUBID. Таким чином пошук по ПРИЄДНУВАНІЙ таблиці буде проводитися в рази швидше [11].

СТРУКТУРА							
Ім'я	Тип	Сортимент	Атрибути	Null	По униканню	Допоміжний	
user_id	INT(11)		PRIMARY	NO	YES	INDEX	
user_name	VARCHAR(255)			NO	NO		
user_country	VARCHAR(255)			NO	NO		
user_date	VARCHAR(255)			NO	NO		
user_gender	ENUM('T','F')			NO	NO		
user_date	INT(11)			NO	NO		

ІНДЕКСИ							
Ім'я	Тип	Унікальний	Унікальний	Рівень	Унікальний	Сортимент	Null
PRIMARY	BTREE	NO	NO	0	A		
user_country	BTREE	NO	NO	0	A		
user_gender	BTREE	NO	NO	0	A		

SQL ЗАПИТИ

A) SELECT * FROM 'user' WHERE 'user_date' > 1516881979 AND 'user_country' IN ('UA', 'US')

B) SELECT 'user_id', 'user_name' FROM 'user' WHERE 'user_name' LIKE '%NICK%' AND 'user_country' IN ('UA', 'US')

B) SELECT 'user_id' FROM 'user' WHERE 'user_country' IN ('UA', 'US') AND 'user_GENDER' = 1 AND 'user_name' LIKE '%NICK%' AND 'user_date' > 1516881979

Рис. 5. Приклади правильних та неправильних SQL запитів

Вибірку та сортування бажано робити за полями індексів, а в найкращому варіанті і зовсім за полем з PRIMARY KEY. Вставку INSERT даних в базу краще робити не поштучно, а об'єднуючи її в невеликі групи до 100 записів за 1 раз [12]. Оновлення даних UPDATE також практичніше робити не за всією кількістю полів, а тільки за тими полями, які оновилися. Наприклад, коли користувач редагує свою анкету та при цьому змінює тільки одне поле з десяти – необхідно на рівні ПО визначити, яке саме поле було змінено та тільки це поле оновити в базі даних.

Якщо після всіх дій запит так і залишається ресурсоємним та довготривалим – рекомендується кешувати результат виконання даного SQL запиту.

ОПИС ПРИНЦИПІВ РОБОТИ ПЗ

Програмний засіб (ПЗ) для оптимізації та побудови нової структури бази даних MySQL функціонує у двох режимах – напівавтоматичний та автоматичний.

У напівавтоматичному режимі під час виконання оптимізації та побудови нової структури ПЗ дозволяє користувачеві підтверджувати або відхиляти в ручному режимі кожне нове оптимальне рішення, а також, в разі кількох варіантів, вибрати найкращий з них на думку користувача.

В автоматичному режимі весь процес оптимізації ПЗ проводить самостійно, без участі користувача.

Незалежно від режиму ПЗ, до початку роботи користувачеві необхідно вказати в якості вхідних параметрів доступи до існуючої бази даних, доступи до бази даних для проведення тестування та доступи до бази даних, в якій буде сформована остаточна структура нової бази даних, список частовиконуваних та проблемних SQL запитів. Також користувачеві необхідно вибрати режим роботи програми та кроки оптимізації структури бази даних, тобто користувач може сам вказати, що треба оптимізувати, а що не треба.

Після завершення роботи ПЗ користувач отримує нову структуру бази даних, оптимізовану спеціально під його запити, отримує виправлені відповідно до нової структури бази даних SQL запити та отримує список рекомендацій, при наявності таких для його бази даних. Також користувачеві відображається безпосередньо результат виконання оптимізації у вигляді звіту про виконану роботу – які таблиці були злиті або розбиті, де

та які індекси були проставлені і т.д. Після кожного кроку оптимізації у звіті додається результат прискорення виконання SQL запитів на новій структурі бази даних в процентному еквіваленті.

Результат прискорення виконання SQL запитів зображено на (1) і вираховується як різниця між сумою часу виконання середнього значення по 100 тестів на новій та на старій структурі бази даних.

$$xTime = \frac{\sum_{n=1}^{100} SQLtOld}{100} - \frac{\sum_{n=1}^{100} SQLtNew}{100}, \quad (1)$$

де $SQLtOld$ – час виконання одного запиту до MySQL зі старою структурою бази даних у секундах;

$SQLtNew$ – час виконання одного запиту до MySQL з новою структурою бази даних у секундах;

$xTime$ – різниця між часом виконання запитів до MySQL зі старою структурою бази даних та новою структурою бази даних у секундах. Якщо $xTime$ більше 0, тоді знайдено більш оптимальний варіант; якщо $xTime$ менше 0, тоді – менш оптимальний варіант запиту; якщо 0, тоді обидва варіанти рівні між собою.

Мова реалізації програмного засобу – PHP. Для оформлення інтерфейсу користувача також були використані такі технології для веб-програмування, як – HTML, CSS, JavaScript (Ajax).

ДЕМОНСТРАЦІЯ ПРОГРАМНОГО ЗАСОБУ ТА РЕЗУЛЬТАТІВ ЙОГО РОБОТИ

Інтерфейс програмного засобу зображено на рисунку 6.



Рис. 6. Інтерфейс програмного засобу

Під час роботи ПЗ у напівавтоматичному режимі користувачеві відображаються проміжні діалогові фрейми, у котрих він може затвердити результат, переглянути інші запропоновані варіанти оптимізації або ж зовсім пропустити цей крок. Також у ході

виконання користувачеві відображається вікно з логами роботи ПЗ, на основі яких і буде пізніше надано звіт про виконані дії. Приклад проміжного діалогового фрейму, а також вікна з логами роботи ПЗ зображено на рисунку 7.



Рис. 7. Вікно з проміжним діалоговим фреймом та логами

Після завершення роботи ПЗ користувачеві надаються результати оптимізації структури бази даних та SQL запитів. Також користувач має можливість

завантажити на свій ПК: звіт про оптимізацію, структуру нової бази даних у SQL форматі, нові SQL запити, які будуть відповідати новій структурі бази даних.

ВИСНОВКИ

У даній статті були розглянуті основні проблеми СУБД MySQL та організації з її використанням високонавантаженої реляційної системи зберігання даних. Виходячи з особистого досвіду авторів даної статті, безпосередньо по кожному з можливих напрямків оптимізації роботи СУБД MySQL були дані рекомендації. Таким чином, у статті були описані кроки вибору типу (engine) бази даних MySQL, були розглянуті проблеми, пов'язані з проектуванням структури бази даних та надано рекомендації щодо їх усунення, був запропонований варіант вирішення проблеми громіздкості таблиць у базі даних. Також були описані дії та надано рекомендації щодо проставлення типів та розмірностей полів даних (атрибутів), були розглянуті індекси полів бази даних і те, як оптимально їх використовувати. Були дані рекомендації зі складання та використання SQL запитів.

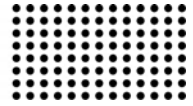
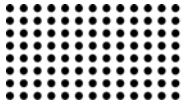
У результаті розгляду вищевказаних проблем та на підставі наданих рекомендацій щодо оптимізації структури та запитів у базах даних MySQL було описано принцип роботи програмного засобу, який в автоматичному режимі буде проводити перепроєктування вже існуючої бази даних з метою прискорення її роботи.

У статті були наведені результати експериментів, отриманих дослідним шляхом з використанням бета версії програмного засобу та бази даних діючого високонавантаженого сервісу. За підсумками експериментів були отримані позитивні результати, а саме – в автоматичному режимі програмний засіб надав новий варіант структури бази даних, яка виявилася на 8.43% швидше попереднього варіанту, отриманого шляхом самостійного ручного перепроєктування.

ДЖЕРЕЛА:

1. SQL. Zbirny`k receptiv / E. Molinaro. – М. : Vy`d. Sy`mvol-Plyus, 2009. – 672 s.
2. Advanced Data Structures / P. Brass. – С. : Vy`d. Cambridge University Press, 2014. – 474 s.
3. MySQL. Opty`mizaciya produkty`vnosti / B. Shvarcz, P. Zajcev, V. Tkachenko, D. D. Zoodnaj, D. Dzh. Balling, A. Lencz. – М. : Vy`d. Sy`mvol-Plyus, 2010. – 832 s.
4. Zabezpechennya vy`sokoyi dostupnosti sy`stem na osnovi MySQL / Ch. Bell, M. Ky`ndal, L. Talmann. – М. : Vy`d. Russkaya Redakcy`ya, 2012. – 624 s.
5. Effective MySQL Optimizing SQL Statements / R. Bradford. – NY. : Vy`d. McGraw-Hill Education, 2011. – 184 s.
6. Instant InnoDB: short, fast, focused / M. Reid. – В. : Vy`d. Packt Publishing, 2013. – 88 s.
7. Database Systems: A Practical Approach to Design, Implementation, and Management / T. Connolly, C. Begg. – L. : Vy`d. Pearson, 2014. – 1440 s.
8. Bazy`dany`x. Proektuvannya, realizaciya i suprovid. Teoriya ta prakty`ka / T. Konnelly`, K. Begg. – М. : Vy`d. Vy`l`yams, 2017. – 1440 s.
9. MySQL: vy`kory`stannya ta administruvannya / V. Vasvani. – SPb. : Vy`d. Py`ter, 2011. – 368 s.
10. Beginning Database Design: From Novice to Professional / C. Churcher. – NY. : Vy`d. Apress, 2012. – 252 s.
11. Vvedennya do sy`stem baz dany`x / K. Dzh. Dajt. – М. : Vy`d. Vy`l`yams, 2017. – 1328 s.
12. Relational Database Design and Implementation, Fourth Edition / J. L. Harrington. – В. : Vy`d. Morgan Kaufmann, 2016. – 712 s.

*Рецензент: д.т.н., проф. К.Е. Петров
Професор кафедри штучного інтелекту ХНУРЕ*



РОЗРОБКА ІНФОРМАЦІЙНОЇ МОДЕЛІ КОМП'ЮТЕРНОЇ СИСТЕМИ КОНТРОЛЮ ЗАВАНТАЖЕННЯ КОНТЕЙНЕРОВОЗА

УДК 629.544:531.7

НАКУЛ Юрій Олександрович

аспірант «Одеської морської академії», м. Одеса, Україна

Наукові інтереси: логістика транспортування контейнерних вантажів морським сполученням, автоматизація процесів завантаження/відвантаження вантажів з контейнеровозів, контроль стійкості судна в залежності від завантаженого вантажу.

e-mail: Nak.01@ukr.net

Процес завантаження контейнеровозів є складним завданням, що вимагає координації багатьох складових. Зазвичай кранові керуються планом завантаження, в якому вказані координати розташування контейнерів в системі координат, прив'язаній до судна. Місця встановлення контейнерів вказуються в цифрах за допомогою системи координат номер ряду-відсіку-ярусу в судновій системі координат. Недбалість під час цього процесу може привести до дуже серйозних наслідків, до яких відноситься неправильний розподіл маси на судні, що може призвести до втрати контейнерів під час транспортування чи затоплення всього судна, також можуть бути здійснені перестановки контейнерів на палубі з втратою інформації, або неконтрольоване вивантаження їх на беріг. Оскільки крановий працює в безперервному режимі, керуючись виключно текстовою та табличною інформацією – зі збільшенням об'ємів робіт зростає кількість можливих помилок, що призводить до надлишкових фінансових витрат. Для здійснення повного контролю за переміщеннями контейнерів під час завантаження та отримання можливості автоматичної перевірки параметрів стійкості судна потрібно мати можливість автоматичної реєстрації їх переміщень та ваги.

Для вирішення поставленого завдання необхідно провести систематизацію та аналіз основних понять, технологій, обладнання, норм і вимог, що пред'являються до системи завантаження, як з боку замовника

перевезення, так і з боку оптимізації процесу завантаження та його відповідності до критеріїв розміщення вантажів та подальшої перевозки. При цьому необхідно розглянути можливості об'єднання звичайної мережі передачі інформації з нетиповими мережами, такими як сенсорну мережу, корабельну, портову та інші, пошук їх місця в загальній структурі інформаційного середовища, розробки нової концептуальної моделі апаратно-програмних комплексів відповідно до введених змін.

В ході досліджень визначено, що найбільш універсальними інформаційними моделями є моделі, що використовуються при побудові промислових інформаційних систем. Використання інформаційних моделей промислових систем дозволяє будувати універсальні комп'ютерні комплекси, здатні вирішувати різні дослідницькі або виробничі завдання. При цьому в разі відсутності будь-яких функцій в системі – в моделі просто не буде використовуватися відповідний рівень. Тому актуальним є розгляд і вдосконалення інформаційних систем для промислових комплексів шляхом визначення в них місця нетипових комп'ютерних систем, а також розробки методів динамічної конфігурації моделі в залежності від розв'язуваних завдань.

На підставі проведеного дослідження існуючих систем планування та контролю завантаження визначено, що у всіх розроблених системах на сьогоднішній день питання розміщення контейнерів визначається досить

суб'єктивно, керуючись лише досвідом капітана судна, чи іншої відповідальної особи. Загальним недоліком існуючих моделей комп'ютерних систем є відсутність можливості контролювати вагу та переміщення контейнерів в реальному часі з можливістю автоматичної генерації звіту для подальшої корекції процесу завантаження. Системи планування та завантаження завжди розглядалися як окремі інформаційні системи, не пов'язана єдиними інформаційним простором з корабельними системами.

Рішення завдання побудови об'єднаної гетерогенної комп'ютерної системи контролю завантаження є актуальною, так як дозволить істотно скоротити час на планування та сам процес завантаження і економічні витрати загалом в процесі завантаження/розвантаження судна.

Метою роботи є розробка інформаційної моделі гетерогенної комп'ютерної системи контролю завантаження контейнеровоза та журналювання його процесу.

ОСНОВНІ РЕЗУЛЬТАТИ

Рішення поставленого завдання на прикладі об'єднання системи планування та контролю процесу завантаження найкраще здійснити методами системного

аналізу, використовуючи логічну схему проектування, базовану на формалізованих елементах (M, S, A, C, R, T), а також – графі (для наочності при топологічному аналізі) [1-13].

Тут M – модель об'єкта проектування;

S – непорожня множина. Елементи множини – завдання проектування;

A – кінцева множина. Елементи множини – початкові дані;

C – кінцева множина. Елементи множини – обмеження, вимоги або інші залежності;

T – кінцева множина. Елементи множини – проектні рішення;

R – відображення, яке ставить кожній парі виду (a_{si}, c_{si}) , $a_{si} \in A$, $c_{si} \in C$ у відповідність непорожню підмножину множини T, що позначається як $R(a_{si}, c_{si})$

Вся множина завдань проектування S вважається повністю вирішуваною, якщо $|R(a_{si}, c_{si})| = |S|$. При цьому повинно виконуватися рівність $|R(a_{si}, c_{si})| = 1$ [1].

Таким чином, для візуального представлення завдання можна побудувати блок-схему відображення всіх вище перерахованих елементів на об'єкт дослідження (рис. 1).

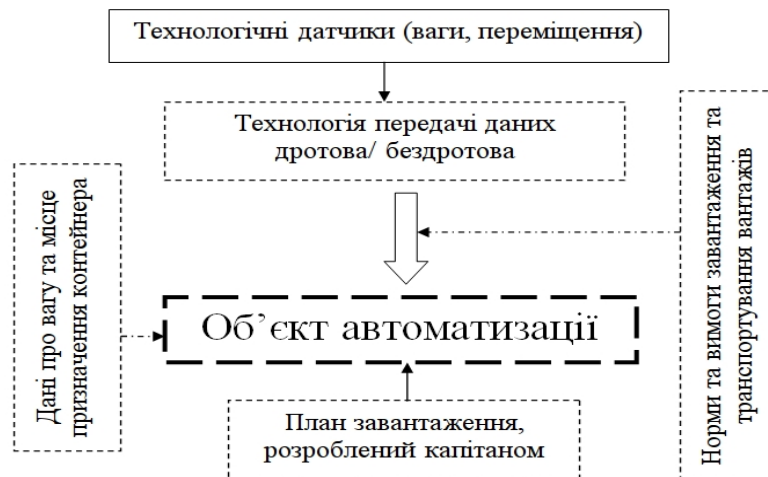


Рис. 1. Блок-схема рішення задачі проектування об'єднаної гетерогенної комп'ютерної системи контролю завантаження

Далі, на основі отриманих умов для вивчення структури взаємозв'язків елементів системи можна вже використовувати топологічний аналіз або аналіз зв'язності. Цей аналіз визначає зв'язність підсистем в системі [4, 9, 12].

Визначимо систему у вигляді $S = \{X, R\}$, де X – множина елементів; R – деяке відношення між елементами. Визначимо для відносини R матрицю інцидентів r, яка влаштована за принципом «хто перед ким має перевагу, або хто кому передає інформацію» [3, 7, 8].

Якщо один елемент матриці має перевагу перед іншими, або другий передає інформацію першому, тобто відношення R виконується, то в клітку (i, j) записується 1; якщо ж відношення R не виконується, то в клітку $(i,$

$j)$ записується 0 [4, 8, 10]. Отже, матриця інцидентції складається з нулів і одиниць (табл.1). Матриця в табл. 1 побудована на основі узагальненої інформації про структуру процесу роботи портових завантажень.

Таблиця 1

Матриця інцидентності r для структури комп'ютерної системи та контролю завантаження контейнеровозу

	Черговість доставки	Вага контейнера	Вартість вантажу	Журналювання процесу завантаження	План завантаження, розробл. капітаном	Черговість надходження контейнерів	Розподіл ваги на палубі
Черговість доставки	0	0	0	1	0	1	0
Вага контейнера	1	0	1	1	1	1	0
Вартість вантажу	1	0	0	1	1	1	0
Журналювання процесу завантаження	0	0	0	0	0	0	0
План завантаження, розроблений капітаном	0	0	0	1	0	1	0
Черговість надходження контейнерів	0	0	0	1	0	0	0
Розподіл ваги на палубі	1	1	1	1	1	1	0

Це завдання на даному етапі можна представити у вигляді неструктурованого графу (рис. 2). Виходячи з неструктурованого графа (рис. 2) побудувати інформаційну модель можна тільки після його структуризації за допомогою топологічного аналізу [4, 11].

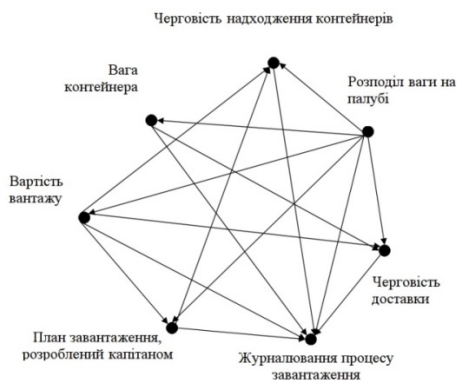


Рис. 2. Неструктурований граф взаємозв'язку елементів комп'ютерної системи контролю завантаження контейнеровозу

Рішення поставленого завдання здійснимо покроково, на кожному кроці визначаючи приналежність кожного з елементів певного рівня, починаючи з найвищого [5]. При цьому на кожному кроці рішення елементи певного рівня будуть виключатися з множини можливих рішень, тим самим спрощуючи вирішуване завдання.

Крок 1. Складаємо вектор-рядок $A0$, що дорівнює сумі рядків вихідної матриці r : $A0 = (3\ 1\ 2\ 6\ 3\ 5\ 0)$. Нульові елементи в рядку $A0$ мають найбільшу перевагу по даному відношенню. Зазначені елементи утворюють порядковий рівень $N0$. У нашому випадку це розподіл ваги на палубі $\{PB\}$ (в табл.2.2 закреслено хвилястою лінією).

Крок 2. Перетворимо рядок $A0$, а саме:

а) нулі замінимо знаком "X";

б) виключимо з рядка $A0$ значення, відповідні "нульовим" елементам, тобто розподіл ваги на палубі (також закреслимо цей рядок в матриці інцидентності).

В результаті отримаємо рядок $A1 = (2\ 0\ 1\ 5\ 2\ 4\ X)$. Нові нулі в рядку $A1$ дають елементи, які мають більшу перевагу перед іншими (крім уже виділених елементів). Вага контейнера $\{BK\}$ (в табл.2.2 закреслено штриховою лінією) утворюють новий порядковий рівень $N1$.

Крок 3. Перетворимо рядок $A1$ аналогічно кроку 2, в підсумку отримаємо рядок $A2 = (1\ X\ 0\ 4\ 1\ 3\ X)$. Утворений новий нуль відповідає елементам вартість вантажу $\{BV\}$, (в табл.2.2 закреслено суцільною лінією), що створює порядковий рівень $N2$.

Крок 4. Перетворимо рядок $A2$, виключаючи значення, відповідні "нульовим" елементам і замінюючи попередні нулі іксом.

В результаті отримаємо рядок $A3 = (0\ X\ X\ 3\ 0\ 2\ X)$. Нові нулі відповідають елементам черговість доставки {ЧД} та план завантаження, розроблений капітаном {ПЗРК} (в табл.2.2 закреслено штрих-пунктирними лініями).

Крок 5. Після перетворення $A3$ отримуємо:

$A4 = (X\ X\ X\ 1\ X\ 0\ X)$, що дає нам рівень $N4$ (в табл.2.2 закреслено двома суцільними лініями): черговість надходження контейнерів {ЧНК}.

Крок 6. Після перетворення $A4$ отримуємо:

$A5 = (X\ X\ X\ 0\ X\ X\ X)$, що дає нам заключний рівень $N5$: журналювання процесу завантаження {ЖПЗ}.

Результати показують, що елементи множини розташовуються за рівнями порядку, починаючи з найвищого – рівень $N0$, і закінчуючи нижчим рівнем – $N5$.

Підсумки розрахунків можна представити в такий спосіб:

{РВП}, {ВК}, {ВВ}, {ЧД, ПЗРК}, {ЧНК}, {ЖПЗ}
 N0 N1 N2 N3 N4 N5

Підсумковий результат можна представити у вигляді порядкового графа (рис. 3), в якому на порядкові рівні накладаються внутрішні зв'язки елементів.

На рис. 2.3 введені наступні позначення: 1 – розподіл ваги на палубі, 2 – вага контейнера, 3 – вартість вантажу, 4 – черговість доставки, 5 – план завантаження, розроблений капітаном, 6 – черговість надходження контейнерів, 7 – журналювання процесу завантаження.

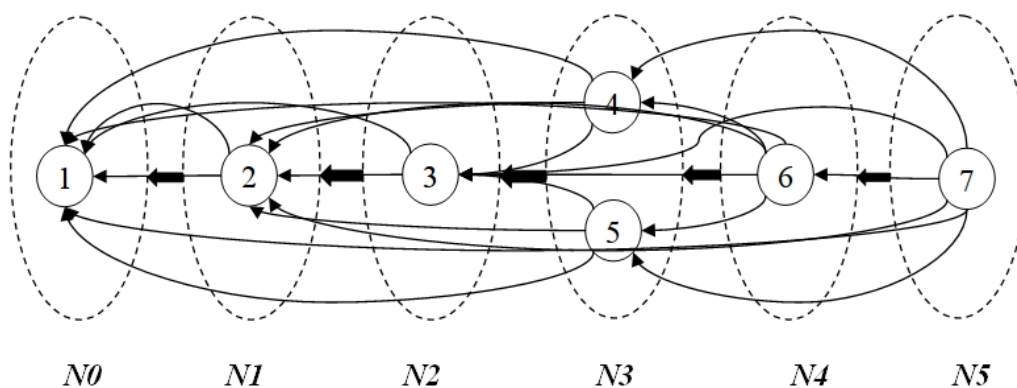


Рис. 3. Структурований граф взаємозв'язку елементів комп'ютерної системи контролю завантаження контейнеровозу

Як видно з графа на рис. 3 інформаційна модель комп'ютерної системи завантаження розділилася на 6 рівнів, кожен зі своїм колом завдань. За допомогою даної моделі в подальшому матимемо змогу розробити автоматизовану систему прийняття рішення при завантаженні контейнеровоза для отримання максимального економічного ефекту та підвищення безпеки транспортування вантажів за рахунок інтелектуалізованої системи прийняття рішення при позиціонуванні контейнера з врахуванням часу його доставки до кінцевого місця призначення, його ваги та розподілу наявної маси вантажів на палубі.

Подання системи з використанням UML-діаграм [2] наведено на рис. 4.

Система реалізується в такий спосіб.

Передбачається, що при заході судна в порт, суднова комп'ютерна система з'єднується з портовою та на час стоянки утворюють єдину систему з сумісним інформаційним полем даних. На борту судна капітан після процесу верифікації в судновій системі вносить інформацію про план подальшого розвантаження / завантаження. Вся внесена інформація вноситься в судову базу даних і є пріоритетною при подальшій роботі. В свою чергу, адміністрація порту має інформацію про підготовлені контейнери для завантаження та їх черговість надходження до порталу крану-завантажувача. Ще до моменту швартування судна, вказані системи при входженні контейнеровоза в порт синхронізуються і розпочинаються підготовчі роботи. Крановий отримує свій план завантаження, користуючись інформацією з вказаних баз даних та починає здійснювати відповідні маніпуля-

ції. Після закінчення процесу швартування машиністи крану розпочинають працювати в повну силу. Згідно завдань з баз даних здійснюють маніпуляції над контейнерами, при цьому система в реальному часі визначає вагу завантажуваних контейнерів та журналює їх

реальні переміщення. При 4 – точковій системі кріплення контейнера до крана система матиме можливість дати оцінку рівномірності розподілу ваги в контейнері та дати рекомендації для додаткового огляду фахівцем.

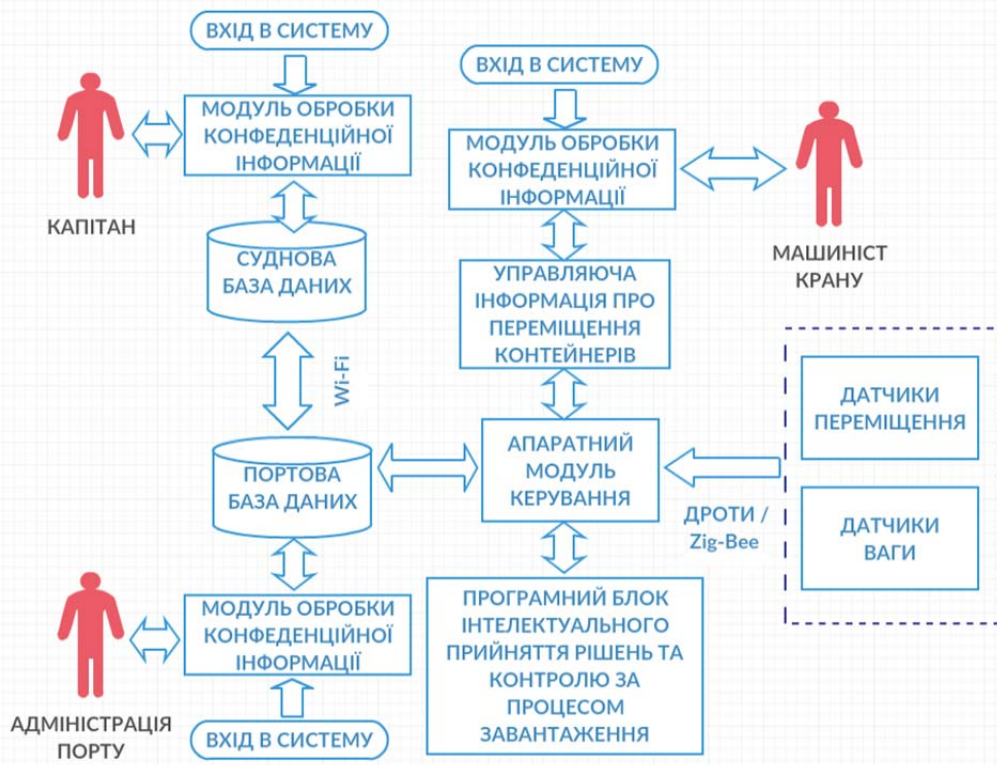


Рис. 4. UML-діаграма комп'ютерної системи контролю завантаження контейнеровозу

Таким чином, розроблена система дозволяє в автоматичному режимі формувати звітність про реальні переміщення контейнерів в середині порту та при завантаженні їх на контейнеровоз. Дається оцінка відповідності розподілу ваги в контейнері та проводиться додаткова перевірка його ваги у відповідності до заданої в базі даних. Структура системи дозволить автоматично спрогнозувати розподіл ваги на контейнеровозі, що важливо для подальшої стійкості судна, спланувати оптимальне розташування контейнерів відповідно до їх черговості доставки та цінності вантажу. Система дозволяє чітко розмежувати права доступу та забезпечує від хибних даних від сторонніх користувачів, зменшити час на завантаження/розвантаження судна, підготовку звітної документації у відповідності до нормативних вимог. Система, наведена в цій формі, є лише концепцією і може інтерпретуватися кожним розробником по-

своєму. Наприклад, при побудові невеликих систем автоматизації, наприклад для річкових портів з великим трафіком судів та обмеженим колом судноплавців, можна всі існуючі бази даних об'єднати в один банк даних з використанням єдиної інформаційно-пошукової системи. Але при побудові систем, що оперують великими масивами інформації, дане спрощення не застосовується.

ВИСНОВКИ

Виконані дослідження показали можливість побудови комп'ютерних систем та мереж, дадуть можливість при заході судна в порт інтегрувати його комп'ютерну систему в портову для підвищення якості процесу завантаження. Приведена структура інформаційної моделі комп'ютерної системи контролю завантаження контейнеровоза. Розроблена модель дозволяє



істотно спростити процес завантаження за рахунок введення цифрових точок контролю та впровадження єдиного інформаційного поля портової зони та судна. Отримані результати дають можливість впроваджувати запропоновану модель у вже існуючі мережі, що значно скорочує час розгортання проекту та економічні витрати при її інтеграції.

Таким чином, вперше отримано інформаційну модель комп'ютерної системи завантаження контейнеровозу, що дозволило оптимізувати та прискорити процес завантаження/розвантаження судна за рахунок об'єднання суднових та портових систем планування і контролю за завантаженням контейнеровозів.

ЛІТЕРАТУРА:

1. Tymchenko A.A. Osnovy systemnoho proektuvannia ta systemnoho analizu skladnykh ob'ektiv [Tekst]: navchalne vydannia, knyha 1 / A.A. Tymchenko. – K. Lybid, 2000. – 270 s.
2. Arlou D. UML 2 y Unyfytsyrovannii protsess. Praktycheskyi ob'єktno-oryentirovannii analiz y proektyrovanye, 2-oe yzdanye [Tekst] / Dzhym Arlou, Aila Neishtadt, – Sankt-Peterburh: Symvol-Plius, 2007. – 624 s.
3. Volkova V.N. Osnovy teoryy system y systemnoho analiza [Tekst]: uchebnyk / V.N. Volkova, A.A. Denysov, yzd. 2. – Spb.: Yzd-vo SPbHTU, 1997. – 510 s.
4. Zghurovskiy M. Z. Osnovy systemnoho analizu [Tekst] / M. Z. Zghurovskiy, N. D. Pankratova, – K.: Vydavnycha hrupa BHV, 2007. – 544 s.
5. Ladaniuk A. P. Osnovy systemnoho analizu [Tekst] / A. P. Ladaniuk Navchalnyi posibnyk. – Vinnytsia, Nova knyha, 2004. – 176 s.
6. Levytyn A. V. Alhorytmy: vvedenye v razrabotku y analiz / A. V. Levytyn; per. s anhl. pod red. Y. V. Krasnykova. – M.: Yzdatelskyi dom «Vyliame», 2006. – 576 s.
7. Myrotyn L.B. Systemnyi analiz v lohystyke [Tekst]: uchebnyk / L.B. Myrotyn, bl.Э. Тышбаев. – M.: Экzamen, 2002. – 480 s.
8. Moiseev N.N. Matematycheskye zadachy systemnoho analiza [Tekst] / N.N. Moiseev. – M.: Nauka, 1981. – 488 s.
9. Perehudov F.Y. Vvedenye v systemnyi analiz [Tekst] / F.Y. Perehudov, F.P. Tarasenko. – M.: VSh, 1989. – 363 s.
10. Romanov V.N. Systemnyi analiz dlia ynzhenerv [Tekst] / V.N. Romanov. M-vo obrazovaniya RF, severo-zapadnyi gosudarstvennyi zaочnyi tekhnicheskyy unyversytet. – yzd. 2-e, dopolnennoe. – SPb: SZHZTU. – 2006. – 186 s.
11. Surmyn Yu.P. Teoryia system y systemnyi analiz [Tekst]: uchebnoe posobyе / Yu.P. Surmyn. – K.: MAUP, 2003. – 368 s.
12. Spytynadel V.N. Osnovy systemnoho analiza [Tekst] / V.N. Spytynadel. – M.: SPb.: Yzd. dom «Byzness-prensa», 2000. – 326 s.
13. Khoroshevskiy V. H. Ynzhenernyy analiz funktsyonirovaniya vychyslytelnykh mashyn y system / V. H. Khoroshevskiy. – M.: Radyo y svyaz, 1987. – 256 s.

*Рецензент: д.т.н., проф. М.П. Мусієнко
Професор кафедри комп'ютерної інженерії
Чорноморського національного університету
імені Петра Могили*



АНОТАЦІЇ

УДК 539.2 / Ходаков В.Є., Соколов О.Є., Веселовська Г.В. / Навчання як процес зміни послідовностей станів того, який навчається, в системі "той, який навчає – той, який навчається" // Проблеми інформаційних технологій. – 2018. – № 1 (023). – С.6-12: Бібл.: 15 назв, рос.

Виконано аналіз процесу навчання того, якого навчають (учня) в системі "той, який навчає – той, якого навчають". Навчання в найзагальнішому вигляді представлено як процес, тобто хід, розвиток явища послідовної зміни станів того, якого навчають. Завдання побудови оптимальної траєкторії навчання відноситься до класу задач управління інформаційними системами та дозволяє знайти часткове рішення рівняння Белмана для побудови оптимального управління інформаційними потоками підсистеми того, який навчає (вчителя). *Ключові слова:* система, стан, процес зміни, навчання, той, який навчає, той, який навчається.

УДК 004.3(075) / Веселовська Г. В., Соколов А. Е. / Удосконалення методів управління складними комп'ютеризованими системами технічних засобів навчання на засадах концепцій методологій систем штучного інтелекту // Проблеми інформаційних технологій. – 2018. – № 1 (023). – С.13-20: Бібл.: 15 назв, укр.

Здійснені дослідження особливостей сучасних комп'ютеризованих систем технічних засобів навчання як складних систем. Запропоновані концептуальні підходи до вдосконалювання методів управління технічними засобами навчання на базі інтелектуалізованих механізмів. *Ключові слова:* комп'ютерні системи, методи управління, складні системи, системи штучного інтелекту, технічні засоби навчання.

УДК 004.891.3 / Чалий С.Ф., Лещинський В.О., Лещинська І.О. / Моделювання контексту в рекомендаційних системах // Проблеми інформаційних технологій. – 2018. – № 1 (023). – С.21-26: Бібл.: 15 назв, укр.

Досліджено проблему формування контексту прийняття рішень як сукупності локальних контекстів споживачів в рекомендаційних системах. Такі системи формують рекомендації відносно вподобань користувачів щодо нових об'єктів на основі існуючих рейтингів. Дана проблема пов'язана з тим, що вибір споживачів виконується в умовах обмежень поточного локального контексту. Для підвищення точності рекомендацій необхідно побудувати узагальнений контекст на основі об'єднання реалізацій локальних контекстів користувачів. Сформульовано принципи локальності та зв'язності для локальних контекстів споживачів. Принцип локальності передбачає локальність представлення одних і тих же власти-

УДК 539.2 / Ходаков В.Е., Соколов А.Е., Веселовская Г.В. / Обучение как процесс изменения последовательности состояний обучаемого в системе "обучающий-обучаемый" // Проблеми інформаційних технологій. – 2018. – № 1 (023). – С.6-12: Бібл.: 15 назв, рос.

Выполнен анализ процесса обучения обучаемого (ученика) в системе "обучаемый – обучающий". Обучение в самом общем виде представлено как процесс, то есть ход, развитие явления последовательной смены состояний обучаемого. Задача построения оптимальной траектории обучения относится к классу задач управления информационными системами и позволяет найти частное решение уравнения Белмана для построения оптимального управления информационными потоками подсистемы обучающего (учителя). *Ключевые слова:* система, состояние, процесс изменения, обучение, обучающий, обучаемый.

УДК 004.3(075) / Веселовская Г. В., Соколов А. Е. / Совершенствование методов управления сложными компьютеризованными системами технических средств обучения на основе концепций методологии систем искусственного интеллекта // Проблеми інформаційних технологій. – 2018. – № 1 (023). – С.13-20: Бібл.: 15 назв, укр.

Выполнены исследования особенностей современных компьютеризованных систем технических средств обучения как сложных систем. Предложены концептуальные подходы к совершенствованию методов управления техническими средствами обучения на базе интеллектуализованных механизмов. *Ключевые слова:* компьютерные системы, методы управления, сложные системы, системы искусственного интеллекта, технические средства обучения.

УДК 004.891.3 / Чалый С.Ф., Лещинский В.А., Лещинская И.А. / Моделирование контекста в рекомендательных системах // Проблеми інформаційних технологій. – 2018. – № 1 (023). – С.21-26: Бібл.: 15 назв, укр.

Исследована проблема формирования контекста принятия решений как совокупности локальных контекстов потребителей в рекомендательных системах. Такие системы формируют рекомендации относительно предпочтений пользователей относительно новых объектов на основе существующих рейтингов. Данная проблема связана с тем, что выбор потребителей выполняется в условиях ограничений текущего локального контекста. Для повышения точности рекомендаций необходимо построить обобщенный контекст на основе объединения реалізацій локальных контекстов потребителей. Сформулированы принципы локальности и связности для локальных контекстов потребителей. Принцип локальности

UDC 539.2 / Hodakov V. Ye., Sokolov A. Ye., Veselovskaya G. V. / The teaching as a changing process the sequence of trainee states in the system of "training-trainee" // The problems of information technologies. – 2018. – № 1 (023). – P.6-12: Bible: 15 titles, rus.

The analysis of the learning process for the learner (student) in the training-trainee system is performed. Teaching in the most general form is presented as a process, that is, a course, the development for the phenomenon of a successive change in the trainee states. The task of constructing the optimal teaching trajectory is related to the class of information systems control problems and allows to find a particular solution of the Bellman equation for constructing optimal control on information flows of the training (teacher) subsystem. *Keywords:* system, state, change process, teaching, training, trainee.

UDC 004.3 (075) / Veselovskaya G. V., Sokolov A. Ye. / The control methods improvement for complex computerized systems of technical means in training on the basis of artificial intelligence systems methodology concepts // The problems of information technologies. – 2018 – № 1 (023). – P.13-20: Bible: 15 titles, ua.

The researches in modern computerized systems features of technical means of training as complex systems are carried out. Conceptual approaches to improving the methods in control for technical means of learning based on intellectualized mechanisms are proposed. *Key words:* computer systems, control methods, complex systems, artificial intelligence systems, technical means of training

U.D.C. 004.891.3 / Chalyi S.F., Leshchynskiy V.O., Leshchynska I.O. / Modeling the context in recommender systems // Problems of Information Technologies. – 2018. – № 1 (023). – С.21-26: Bible: 15 titles, ua.

The problem of forming the context of decision making as a set of local contexts of consumers in the reference systems is researched. Such systems formulate recommendations regarding user preferences for new objects based on existing ratings. This problem is due to the fact that the choice of consumers is carried out in conditions of limitations of the current local context. In order to improve the accuracy of the recommendations, it is necessary to construct a generalized context based on the integration of implementations of local contexts of users. The principles of locality and connectivity for local contexts of consumers are formulated. The principle of locality assumes the location of the representation of the same properties of objects by different consumers, which

востей об'єктів різними споживачами, що може призвести до відмінностей локальних контекстів.. Принцип зв'язності передбачає упорядкування сукупності локальних контекстів споживачів на основі встановлення співвідношень між цими контекстами. На основі представлених принципів запропоновано модель контексту прийняття рішень рекомендаційної системи. Модель містить у собі послідовності реалізації локального контексту користувачів, що пов'язані відношеннями еквівалентності, схожості та сумісності, що дає можливість підвищити точність ранжування об'єктів та споживачів в рекомендаційних системах шляхом врахування поточного контексту прийняття рішень. *Ключові слова:* рекомендаційні системи, ранжування результатів; контекст прийняття рішень, рекомендаційні системи, локальність, зв'язність.

UDK 004.9:510.635 / Дорошенко А.Ю., Шаронова Н.В., Єна Б.О., Янголенко О.В. / Розробка програмних компонентів інформаційної системи екстракції фактографічних даних з веб-ресурсів // Проблеми інформаційних технологій. – 2018. – №1 (23). – С.27-35: Библ.: 9 назв.

В дослідженні проведено аналіз проблеми та особливості практичної реалізації вирішення задачі екстракції фактографічних даних. Проаналізовано підходи та інформаційні технології вирішення задач парсингу на базі існуючих інформаційних систем. Створено специфікацію вимог до програмного забезпечення, що дозволить у подальшій роботі над проектом чітко розуміти вимоги та обмеження до реалізації. Розроблено еталонну архітектуру та запропоновано варіант розгортання програмної системи. Розроблено програмні компоненти серверної частини програмної системи, що дозволяє проводити екстракцію даних з торговельних площадок на основі використання гнучкого конфігурування та предикатної моделі видобування даних. Розроблено та імплементовано програмні компоненти для збору даних на прикладі збору характеристик моделей мобільних телефонів. Проведено тестування розроблених компонентів та доведено їх працездатність для збору даних з трьох різних торговельних площадок. Результати тестування дозволяють стверджувати, що отримані результати можуть бути базисом створення програмної системи для видобування фактографічної інформації. Запропоноване архітектурне рішення може бути розвинуто за рахунок імплементування моделі екстракції необхідної інформації. Додаткові джерела даних можуть бути додані в систему за допомогою YAML конфігурації.

предполагает локальность представления одних и тех же свойств объектов различными потребителями, что может привести к различиям локальных контекстов.. принцип связности предусматривает упорядочение совокупности локальных контекстов потребителей на основе установления соотношений между этими контекстами. На основе представленных принципов предложена модель контекста принятия решений рекомендательной системы. Модель включает в себя последовательности реализации локального контекста пользователей, связанные отношениями эквивалентности, сходства и совместимости, что позволяет повысить точность ранжирования объектов и потребителей в рекомендательных системах путем учета текущего контекста принятия решений. *Ключевые слова:* рекомендательные системы, ранжирование результатов; контекст принятия решений, рекомендательные системы, локальность, связность.

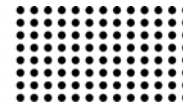
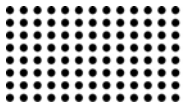
UDK 004.9:510.635 / Дорошенко А.Ю., Шаронова Н.В., Єна Б.О., Янголенко О.В. / Разработка программных компонентов информационной системы экстракции фактографических данных из веб-ресурсов // Проблеми інформаційних технологій. – 2018. – №1 (23). – С. 27-35: Библ.: 9 назв., укр.

В исследовании проведен анализ проблемы и особенностей практической реализации решения задачи экстракции фактографических данных. Проанализированы подходы и информационные технологии решения задач парсинга на базе существующих информационных систем. Создано спецификацию требований к программному обеспечению, что позволит в дальнейшей работе над проектом четко понимать требования и ограничения к реализации. Разработана эталонная архитектура и предложено вариант развертывания программной системы. Разработаны программные компоненты серверной части программной системы, позволяющей проводить экстракцию данных с торговых площадок на основе использования гибкого конфигурирования и предикатной модели добычи данных. Разработаны и имплементированы программные компоненты для сбора данных на примере сбора характеристик моделей мобильных телефонов. Проведено тестирование разработанных компонентов и доказана их работоспособность для сбора данных с трех разных торговых площадок. Результаты тестирования позволяют утверждать, что полученные результаты могут быть базисом создания программной системы для добычи фактографической информации. Предложенное архитектурное решение может быть развито за счет имплементации модели экстракции необходимой информации. Дополнительные источники данных могут быть добавлены в систему с помощью YAML конфигурации.

may lead to differences in local contexts. The principle of connectivity involves arranging the totality of local contexts of consumers based on establishing relationships between these contexts. Based on the presented principles, a model of the decision making context of the advisory system is proposed. The model contains sequences of implementations of the local context of users that are related to the equivalence, similarity and compatibility, which makes it possible to improve the accuracy of the ranking of objects and consumers in advisory systems by taking into account the current context of decision-making. *Keywords:* recommender systems, ranking of results; decision making context, reference systems, locality, connectivity.

UDC 004.9:510.635 / Doroshenko A.Yu., Sharonova N.V., Yena B.O., Yanholenko O.V. / Development of software components for information extraction of factual data from web resources // Problems of Information Technology. – 2018. – №1 (23). – P. 27-35: Bibl.: 9 names, ukr.

The analysis of the problem and features of the practical implementation of solution for factual data extraction problem was carried out in the research. Approaches and information technologies of solving parsing problems on the basis of existing information systems were analyzed. A specification of software requirements was created, which will clearly allow understanding the requirements and restrictions of the implementation on the further work on the project. The standard architecture was developed and the variant of deployment of the software system was proposed. The server part software components of the software system were developed, which allows the extraction of data from marketplaces, based on the use of flexible configuration and predicate model of data mining. Software components for data collection on the example of collecting characteristics of mobile phone models were developed and implemented. The testing of the developed components has been carried out and their efficiency has been proved, which can allow collecting data from three different marketplaces, the results of which allow asserting that the obtained results can be the basis for the creation of a software system for the extraction of factual information. The proposed architectural solution can be developed by implementing a model of extraction of necessary information. Additional data sources can be added to the system using the YAML configuration.



УДК 004 /Бродський Ю.Б., Гришук Р.В. / Аналіз ролі та місця, сутності та змісту інформаційних технологій на сучасному етапі розвитку науки і техніки // Проблеми інформаційних технологій. – 2018. – №1 (023). – С.36-42. Бібл.: 26 назв., укр.

Проведено системний аналіз поняття “інформаційна технологія” з урахуванням особливостей сучасного етапу розвитку суспільства, науки і техніки. Показано, що інформаційна технологія є смисловою системною категорією і відображає такі домінуючі як мета, процес, оптимальність. Визначено роль та місце інформаційних технологій в сучасних системах геофізичного моніторингу. *Ключові слова:* інформаційна технологія, системний аналіз, інформація, система геофізичного моніторингу.

УДК 685.34.01 / В.І. Чупринка, Г.Ю. Зелінський, Н.В.Чупринка / Автоматичне проектування схем розкрою листових матеріалів на плоскі геометричні об'єкти / Проблеми інформаційних технологій. – 2018. – № 1 (023). – С.43-53: Бібл.: 5 назв, англ.

В роботі був запропонований метод автоматичного проектування раціональних схем щільного розміщення плоских геометричних об'єктів в прямокутній області заданих розмірів. Запропоновано метод автоматичного проектування раціональних схем щільного розміщення плоских геометричних об'єктів в прямокутній області заданих розмірів, який більш ефективний за існуючі. Результати теоретичних досліджень були реалізовані в програмний модуль для автоматичного проектування раціональних схем щільного розміщення плоских геометричних об'єктів в прямокутній області заданих розмірів. Розроблений програмний продукт може бути використаний в легкій промисловості для автоматичного проектування раціональних схем розкрою листових матеріалів на деталі взуття. *Ключові слова:* щільне розміщення, структурні компоненти, функція цілі, секція, схема розкрою, план розкрою, програмне забезпечення.

УДК 519.7: 681.3 / П.І. Бідюк, І.В. Пудло, Є.О. Демківський, Т.І. Демківська / Прогнозування дохідності банківських продуктів з використанням скорингового підходу / Проблеми інформаційних технологій. – 2018. – №1 (023). – С. 54-60. Бібл.: 10 назв, англ.

В даній роботі було розглянуто проблему кредитного ризику, визначені причини його виникнення та підходи до оцінки. Проаналізовано супутні ризики в процесі кредитування на прикладі револьверних карток з грейсовим (пільговим) періодом. Було розглянуто основні етапи побудови скорингової моделі: формування вибірки, сегмен-

УДК 004 /Бродский Ю.Б., Гришук Р.В. / Анализ роли и места, сущности и содержания информационных технологий на современном этапе развития науки и техники // Проблемы информационных технологий. – 2018. – №1 (023). – С.36-42. Библ.: 26 назв., укр.

Проведен системний аналіз поняття “інформаційна технологія” з урахуванням особливостей сучасного етапу розвитку суспільства, науки і техніки. Показано, що інформаційна технологія представляє собою системну категорію і відображає такі домінуючі як мета, процес, оптимальність. Определена роль и место информационных технологий в современных системах геофизического мониторинга. *Ключевые слова:* информационная технология, системный анализ, информация, система геофизического мониторинга.

УДК 685.34.01 / В.І. Чупринка, Г.Ю. Зелінський, Н.В.Чупринка / Автоматическое проектирование схем раскрою листовых материалов на плоские геометрические объекты / Проблемы информационных технологий. – 2018. – № 1 (023). – С.43-53: Библ.: 5 назв, англ.

В работе был предложен метод автоматического проектирования рациональных схем плотного размещения плоских геометрических объектов в прямоугольной области заданных размеров. Предложенный метод автоматического проектирования рациональных схем плотного размещения плоских геометрических объектов в прямоугольной области заданных размеров, который более эффективен существующих. Результаты теоретических исследований были реализованы в программный модуль для автоматического проектирования рациональных схем плотного размещения плоских геометрических объектов в прямоугольной области заданных размеров. *Ключевые слова:* плотное размещение, структурные компоненты, функция цели, секция, схема раскроя, план раскроя, программное обеспечение.

УДК 519.7: 681.3 / П.И. Бидюк, И.В. Пудло, Е.А. Демковский, Т.И. Демковская / Прогнозирование доходности банковских продуктов с использованием скорингового подхода / Проблемы информационных технологий. – 2018. – № 1 (023). С. 54-60. – Библ.: 10 названий, англ

В данной работе была рассмотрена проблема кредитного риска, определены причины его возникновения и подходы к оценке. Проанализированы сопутствующие риски в процессе кредитования на примере револьверных карт с грейсовым (льготным) периодом. Были рассмотрены основные этапы построения скоринговой модели:

UDC 004 /Brodsky Yu.B., Hryshchuk R.V. / Analysis of the role and place, essence and content of information technologies at the present stage of science and technology development // The problems of Information Technologies. – 2018. – №1 (023). – P.36-42. Ref.: 26 titles, ukr.

We conducted a system analysis of the concept of “information technology” taking into account the peculiarities of the current development of society, science and technology. It is shown that information technology is a system category and displays such dominants as purpose, process, and optimality. We have also defined the role and place of information technologies in modern geophysical monitoring systems. *Key words:* information technology, system analysis, information, geophysical monitoring system.

UDC 685.34.01 /V.I. Chuprinka, G.U. Zelinsky, N.V. Chuprinka /Automatic planning cutting schemes of sheets on flat geometric object/ – 2018. –№ 1 (023). P. 43-53: Bibl.: 5 titles, eng.

In the paper, a method was proposed for the automatic design of rational schemes for the dense placement of planar geometric objects in a rectangular region of specified dimensions. A method is proposed for automatic design of rational schemes for the dense placement of planar geometric objects in a rectangular region of specified dimensions that is more efficient than existing ones. The results of theoretical studies were implemented in a software module for the automatic design of rational schemes for the dense placement of planar geometric objects in a rectangular region of specified dimensions. The developed software product can be used in the light industry for the automatic design of rational schemes for cutting sheet materials into parts of shoes. *Key words:* dense placement, structural components, target function, section, cutting layout, cutting plan, software.

UDC 519.7: 681.3 / P.I. Bidiuk, I.V. Pudlo, E.A. Demkivsky, T.I. Demkivska / Profitability Prediction Of Banking Products Using Scoring Approach. – 2018. – №1 (023). P. 54-60. Bibl.: 10 titles, eng.

This paper examined the problem of credit risk, determined causes and approaches to assessment. Analyzed related risks in the granting of credit process by the example of revolving cards grace with period. It was examined the main stages of scoring model construction: sampling, segmentation characteristics and so on. The result of work is developed application

тації характеристик тощо. Результатом роботи стала побудована аплікаційна скорингова модель, яка, на основі визначених параметрів, дозволяє провести сегментацію клієнтів таким чином, щоб банк не зазнав збитків від недоотримання прибутку. *Ключові слова:* кредитування; кредитний ризик; скоринг; кредитоспроможність; револьверні картки; грейсовий період; регресія.

УДК 004.23 / М.В. Деркач, І.С. Скарга-Бандурова / Дослідження методів розрахунку відстані для окремих сегментів маршруту // Проблеми інформаційних технологій. – 2018. – № 1 (023). – С.61-66: Бібл.: 12 назв., укр.

У роботі розглянута проблема точності розрахунку відстані між двома географічними координатами з використанням методів заснованих на формулах гаверсинусів, Вінсенті та Евклідової відстані. Запропоновано методу розрахунку з використанням лінеаризації маршруту шляхом суми відстаней окремих сегментів для кожного методу. В якості вхідних даних для проведення дослідження були взяті географічні координати троллейбусних зупинок вздовж маршрутів міста Северодонецьк. Також проведена оцінка ефективності застосування різних методів розрахунку відстані між початковою і кінцевою точкою на основі відношення отриманого результату до фактичної відстані для розглянутих способів, отримана середня помилка відхилення на кожному діючому маршруті міста. *Ключові слова:* формули гаверсинусів, метод Вінсенті, Евклідова відстань, маршрут, лінеаризація, сегмент.

УДК 004.75, 65.011.56 / Дідук В.А. / Методика прийняття рішення по місцю розташування продукції в складських приміщеннях // Проблеми інформаційних технологій. – 2018. – № 1 (023). – С.67-72: Бібл.: 10 назв., укр.

Планування розміщення продукції в складських приміщеннях – відповідальний процес, від якого в значній мірі залежить ефективність діяльності всього підприємства. Існуючі системи автоматизації роботи в складських приміщеннях переважно орієнтовані на звичайний облік ресурсів, та не передбачають можливості проводити автоматичну оптимізацію розміщення продукції на складі. В роботі пропонується методика підбору оптимального місця розташування продукції по секторам у відповідності до значення їх критеріального показника, що є комплексною величиною їх часу зберігання на складі, частоті запитів клієнтів, ваги, вибухонебезпечності та інше... Метод дає можливість модернізації існуючих WMS систем, чи лягти в основу створення нових систем. *Ключові слова:* склад, продукція, інформаційна система, комп'ютерна система, WMS система, методика.

формирование выборки, сегментации характеристик и тому подобное. Результатом работы стала построена формуляр скоринговая модель, на основе определенных параметров, позволяет провести сегментацию клиентов таким образом, чтобы банк не понес убытки от недополучения прибыли. *Ключевые слова:* кредитование; кредитный риск; скоринг; кредитоспособность; револьверные карты; грейсовый период; регрессия.

УДК 004.23 / М.В. Деркач, І.С. Скарга-Бандурова / Исследование методов расчета расстояния для отдельных сегментов маршрута // Проблеми інформаційних технологій. – 2018. – № 1 (023). – С.61-66: Бібл.: 12 назв., укр.

В работе рассмотрена проблема точности расчета расстояния между двумя географическими координатами различными методами, а именно были использованы метод, основанный на формулах гаверсинусов, методы Винсенти и Евклидова расстояния. Предложена методика расчета с использованием линейаризации маршрута путем суммирования отдельных сегментов для каждого метода. В качестве входных данных для проведения исследования были взяты географические координаты троллейбусных остановок вдоль маршрутов города Северодонецк. Также проведена оценка эффективности применения различных методов расчета расстояния между начальной и конечной точкой на основе отношения полученного результата к фактическому расстоянию для рассматриваемых способов, получена средняя ошибка отклонения на каждом действующем маршруте города. *Ключевые слова:* формулы гаверсинусов, метод Винсенти, Евклидово расстояние, маршрут, линейаризация, сегмент.

УДК 004.75, 65.011.56/ Дідук В. А. / Методика прийняття рішення по месту расположения продукции в складских помещениях // Проблеми інформаційних технологій. – 2018. – № 1 (023). – С.67-72: Бібл.: 10 назв., укр.

Планирование размещения продукции в складских помещениях - ответственный процесс, от которого в значительной степени зависит эффективность деятельности всего предприятия. Существующие системы автоматизации работы в складских помещениях преимущественным образом ориентированы на обычный учет ресурсов, и не предусматривают возможности проводить автоматическую оптимизацию размещения продукции на складе. В работе предлагается методика подбора оптимального места расположения продукции по секторам в соответствии со значением их критеріального показателя, является комплексной величиной их времени хранения на складе, частоте запросов клиентов, веса, взрывоопасности и другое... Метод дает возможность модернізації WMS систем, или лечь в основу создания новых систем. *Ключевые слова:* склад, продукция, информационная система,

scoring model, which on base of specified parameters, allows to make segmentation of customers in order to prevent bank from losses of revenue. *Keywords:* credit; credit risk; scoring; creditworthiness; revolving card; grace period; regression.

UDC 004.23 / M.V. Derkach, I.S. Skarga-Bandurova / A study on distance calculation methods for the local segments of the route // Problems of Information Technologies. – 2018. – № 1 (023). – С.61-66: Bible: 12 titles, ua

The paper considers the problem of the accuracy of calculating the distance between two geographical coordinates for the city routes. Different methods, namely, the Haversinus formulas, Vincenty and Euclidean distance methods were utilized. A calculation technique is proposed using the linearization of the route by summing the distances of individual segments for each method. As input to the study, geographic coordinates of trolleybus stops along the routes of the city Severodonetsk were taken. In addition, an assessment was made of the effectiveness of the application of various methods for calculating the distance between the start and ends points, based on the ratio of the result obtained to the actual distance for the methods in question, an average error of deviation was obtained for each current route of the city. *Key words:* Haversinus formulas, Vincenty method, Euclidean distance, route, linearization, segment.

U.D.C. 004.75, 65.011.56/ Diduk V. A. / Method of making a decision on the location of products in the warehouse // Problems of Information Technologies. – 2018. – № 1 (023). – С.67-72: Bible: 10 titles, ua.

Planning of product placement in warehouses is a responsible process, on which the effectiveness of the entire enterprise depends to a large extent. Existing automation systems in the warehouse are predominantly oriented to the usual inventory of resources, but do not provide for the possibility of automatic optimization of product placement in the warehouse. The paper proposes a methodology for selecting the optimal location of products by sectors in accordance with the value of their criterion, which is the complex value of their storage time in the warehouse, the frequency of customer requests, weight, explosion hazard, etc. The method allows for the modernization of existing WMS systems, whether to lie in the basis for creating new systems. *Keywords:* warehouse, products, information system, computer system, WMS system, methodology.

компьютерная система, WMS система, методика.

УДК 004.946. / Доровська І.О., Доровський Д.В. / Сценарно-процедурна модель функціонування віртуальних тренажерів для підготовки менеджерів // Проблеми інформаційних технологій. - 2018. - № 1 (23). С.73-80: Бібл.: 5 назв., Рос.

У роботі використані методи теорії множин, теорії графів, теорії штучного інтелекту, а також технології об'єктно-орієнтованого програмування. У статті показано використання моделей і методів створення та управління віртуальним тренажером, які дозволять підвищити якість професійної підготовки менеджерів.

УДК 004.946. / Доровская И.А., Доровской Д.В. / Сценарно-процедурная модель функционирования виртуальных тренажеров для подготовки менеджеров // Проблемы информационных технологий. - 2018. - № 1(23). С.73-80: Библ.: 5 назв., рус.

В работе использованы методы теории множеств, теории графов, теории искусственного интеллекта, а также технологии объектно-ориентированного программирования. В статье показано использование моделей и методов создания и управления виртуальным тренажером, которые позволят повысить качество профессиональной подготовки менеджеров.

UDC 004.946. / Dorovskaya IA, Dorovskoy DV / Scenario and procedural model of virtual simulators for managers training // Problems of Information Technologies. - 2018. - No. 1 (23). C. 73-80: Bibliography: 5 titles, Russian.

The methods of set theory, graph theory, the theory of artificial intelligence, as well as object-oriented programming technology were used in the work. The article shows the use of models and methods for creating and managing a virtual simulator, which will improve the quality of professional training of managers.

УДК 519.6 / Димов В.С., Боскін О.О. / Використання людино-машинного інтерфейсу для управління технологічними процесами // Проблеми інформаційних технологій. - 2018. - № 1 (023). - С.81-86: Бібл.: 12 назв, укр.

Розглянута система побудови нейроінтерфейсного маніпулятора для управління технічними пристроями (в тому числі за допомогою комп'ютерної техніки) для людей з будь-якими фізичними можливостями (або при їх відсутності). Пристрій, що розробляється, може знайти застосування в будь-яких системах управління для різних об'єктів та співпрацівників, які мають право керувати зазначеними об'єктами. *Ключові слова:* нейрокомп'ютерний інтерфейс, система управління, датчик, управління, зворотній зв'язок.

УДК 62.505 / Дымов В.С., Боскин О.О. / Использование человеко-машинного интерфейса для управления технологическими процессами // Проблемы информационных технологий. - 2018. - № 1 (023). - С.81-86: Библ.: 12 назв., укр.

Рассмотрена система построения нейроинтерфейсного манипулятора для управления техническими устройствами (в том числе с помощью компьютерной техники) для людей с любыми физическими возможностями (или при их отсутствии). Устройство разрабатываемого может найти применение в любых системах управления для различных объектов и сотрудников, имеющих право управлять указанными объектами. *Ключевые слова:* нейрокомпьютерный интерфейс, система управления, датчик, управление, обратная связь.

UDC 62.505 / Dymov V.S., Boskin O.O. / The use of human-machine interface for process control // The problems of information technologies. - 2018. - № 1 (023). - P.81-86: Ref.: 12 titles, ukr.

A system for constructing neurointerfeysnogo manipulator technical devices (including computer-assisted) for people with any physical capabilities (or lack thereof). The device developed can be used in all control systems for a variety of facilities and staff with the right to manage specified objects. *Keywords:* brain-computer interface, control system, sensor, control, feedback.

УДК 681.5 / О.С. Герасин, О.В. Козлов, Г.В. Кондратенко, Huang Mingxin / Синтез та дослідження математичної моделі гусеничного мобільного робота для вертикального переміщення // Проблеми інформаційних технологій. - 2018. - № 1 (023). - С.87-97: Бібл.: 23 назв., англ.

В статті розглянуто синтез та дослідження математичної моделі гусеничного мобільного робота (МР) з окремими основними притискними магнітами для переміщення та виконання заданих технологічних операцій на похилих та вертикальних феромагнітних поверхнях. Авторами розроблені математичні моделі окремих елементів МР та робота в цілому з урахуванням основних особливостей його конструктивної побудови. Отримана математична модель дозволяє досліджувати поведінку робота в різних виробничих умовах при просторовому русі по похилій поверхні та може бути застосована для подальшого синтезу системи автоматичного керування позиціонуванням подібних роботів. *Ключові слова:* гусеничний мобільний робот; магнітний притискний пристрій; переміщення по феромагнітній поверхні; математичне моделювання; система автоматичного керування.

УДК 681.5 / А.С. Герасин, А.В. Козлов, Г.В. Кондратенко, Huang Mingxin / Синтез и исследование математической модели гусеничного мобильного робота для вертикального перемещения // Проблемы информационных технологий. - 2018. - № 1 (023). - С.87-97: Библ.: 23 назв., англ.

В статье рассмотрены синтез и исследование математической модели гусеничного мобильного робота (МР) с отдельными основными прижимными магнитами для перемещения и выполнения заданных технологических операций на наклонных и вертикальных ферромагнитных поверхностях. Авторами разработаны математические модели отдельных элементов МР и робота в целом с учетом основных конструктивных особенностей. Полученная математическая модель позволяет исследовать поведение робота в различных производственных условиях при пространственном движении по наклонной поверхности и может быть использована для дальнейшего синтеза системы автоматического управления позиционированием подобных роботов. *Ключевые слова:* гусеничный мобильный робот; магнитное прижимное устройство; перемещение по ферромагнитной поверхности;

UDC 681.5 / Gerasin O.S., Kozlov O.V., Kondratenko G.V., Mingxin H. / Synthesis and study of the mathematical model of a caterpillar mobile robot for vertical movement // Problems of information technologies. - 2018. - No. 1 (023). P.87-97: Ref.: 23 titles, eng.

The article considers the synthesis and study of a mathematical model of a caterpillar mobile robot (MR) with separate main clamping magnets for moving and performing specified technological operations along inclined and vertical ferromagnetic surfaces. The authors have developed the mathematical models of MR's individual elements and the robot as a whole, taking into account the basic features of its structural design. The resulting mathematical model allows studying the MR's behavior in different production conditions under spatial motion on an inclined surface and can be applied for further synthesis of the automatic control system for positioning such robots. *Keywords:* caterpillar mobile robot; magnetic clamping device; motion along ferromagnetic surface; mathematical modeling; automatic control system.

математическое моделирование; система автоматического управления.

УДК 510.6 / Захарченко Л.М., Захарченко Р.М., Кірюшатов Т.Г., Кірюшатов К.В. / Підготовка користувачів нової версії програми 1С: Підприємство 8.3 для України // Проблеми інформаційних технологій. – 2018. – № 1 (023). – С.98-103: Бібл.: 10 назв., укр.

Проведено дослідження нових можливостей версії програми 1С: Підприємство 8.3 для України та необхідність підготовки кадрів для роботи з нею. Запропоновано використання нової версії програми 1С: Підприємство 8.3 на прикладі описаної методики обліку по нарахуванню та виплаті зарплати. *Ключові слова:* інформаційні технології, інтерфейс, рефакторинг.

УДК 510.6 / Захарченко Л.Н., Захарченко Р.Н., Кірюшатов Т.Г., Кірюшатов Е.В. / Подготовка пользователей новой версии программы 1С: Предприятие 8.3 для Украины // Проблемы информационных технологий. – 2018. – № 1 (023). – С.98-103: Библ.: 10 назв., укр.

Проведено исследование новых возможностей версии программы 1С: Предприятие 8.3 для Украины и необходимость подготовки кадров для работы с ней. Предложено использование новой версии 1С: Предприятие 8.3 на примере описанной методики учета по начислению и выплате зарплаты. *Ключевые слова:* информационные технологии, интерфейс, рефакторинг.

УДК 510.6 / Zaharchenko L. N., Zaharchenko R. N., Kiryushatova T.G., Kiryushatova K.V. / Preparation of users of the new version of 1C: Enterprise 8.3 for Ukraine // Problems of Information Technologies. – 2018. – № 1 (023). – P. 98-103: Bibl.: 10 titles. UKR.

The study of new features of the 1C: Enterprise 8.3 software version for Ukraine is the need for training personnel to work with it. The use of the new version 1C: Enterprise 8.3 is suggested using the example of the described accounting procedure for calculating and paying salaries. *Keywords:* information technology, interface, refactoring.

УДК 681.3.06 / Лобода О.М., Димов В.С. / Модели та методи інформаційної технології управління аграрного сектору економіки за допомогою достатніх умов оптимальності // Проблеми інформаційних технологій. – 2018. – № 1 (023). – С.104-110 Бібл.: 10 назв, укр.

Досліджено комплексний метод ідентифікації пов'язаний з побудовою оптимізаційної моделі, кінцевим результатом якого при використанні знайдених виробничих функцій буде виробіток рекомендацій для прийняття рішень по розподілу засобів між галузями. Встановлено необхідність створення, на основі достатніх умов оптимальності, моделі оптимального розвитку сільськогосподарського підприємства. Розроблена основна характеристика збалансованого зросту (магістраль) сільськогосподарського підприємства та розглянута задача оптимізації моделі з урахуванням запізнення введення основних виробничих засобів. *Ключові слова:* система управління, ідентифікація системи, виробничі функції, оптимізаційні моделі.

УДК 681.3.06 / Лобода Е.Н., Дымов В.С. / Модели и методы информационной технологии управления аграрного сектора экономики с помощью достаточных условий оптимальности // Проблемы информационных технологий. – 2018. – № 1 (023). – С.104-110: Библ.: 10 назв., укр.

Исследован комплексный метод идентификации связанный с построением оптимизационной модели, конечным результатом которого при использовании найденных производственных функций будет выработка рекомендаций для принятия решений по распределению средств между отраслями. Установлена необходимость создания на основе достаточных условий оптимальности, модели оптимального развития сельскохозяйственного предприятия. Разработана основная характеристика сбалансированного роста (магистраль) сельскохозяйственного предприятия и рассмотрена задача оптимизации модели с учетом запаздывания ввода основных производственных средств. *Ключевые слова:* система управления, идентификация системы, производственные функции, оптимизационные модели.

UDC 62.505 / Loboda O.M., Dymov V.S. / Models and methods of information technologies management of the agrarian sector of economics by sufficient optimum conditions // The problems of information technologies. – 2018. – № 1 (023). – P.104-110 Ref.: 10 titles, ukr.

The article investigates the complex method of identification associated with the construction of an optimization model. While using the found production functions its final result will be the development of recommendations for decision-making on the distribution of funds between industries. The necessity of creating an optimal model of an agricultural enterprise development is established on the basis of sufficient optimality conditions. The main characteristic of the balanced growth (artery) of the agricultural enterprise is developed. The problem of model optimization is considered taking into account the delay in the input of the main production assets. *Key words:* control system, system identification, production functions, optimization models.

УДК 681.5.017 / Мацуї А.М., Кондратець В.О. / Математичне моделювання автоматичного контролю енергоефективності подрібнення руди кульовими млинами при спрацюванні футеровки // Проблеми інформаційних технологій. – 2018. – №1 (023). – С.111-117: Бібліогр.: 19 назв., укр.

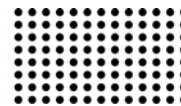
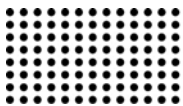
В статті представлені результати моделювання автоматичного контролю енергетичної ефективності подрібнення руди кульовими млинами при спрацюванні футеровки, який передбачає визначення параметра безпосередньо в барабані технологічного агрегату. Для проведення досліджень запропоновані математичні моделі розвантаженого і звантаженого пружного елемента. Встановлено, що в результаті спрацювання футеровки

УДК 681.5.017 / Мацуї А.Н., Кондратец В.А. / Математическое моделирование автоматического контроля энергоэффективности измельчения руды шаровыми мельницами при износе футеровки // Проблемы информационных технологий. – 2018. – №1 (023). – С.111-117: Библиогр.: 19 назв., укр.

В статье представлены результаты моделирования автоматического контроля энергетической эффективности измельчения руды шаровыми мельницами при износе футеровки, который предусматривает определение параметра непосредственно в барабане технологического агрегата. Для проведения исследований предложены математические модели разгруженного и загруженного упругого элемента. Установлено, что в результате износа футеровки возникает система-

UDC 681.5.017 / Matsui A. N., Kondratets V. A. / Mathematical modeling of automatic control of energy efficiency of ore grinding by ball mills with wear of lining // The problems of information technologies. – 2018. – №1 (023). – P.111-117: Ref.: 19 titles., ukr.

The article presents the results of modeling the automatic control of the energy efficiency of grinding ore by ball mills with wear of the lining, which involves determining the parameter directly in the drum of the process unit. To conduct research, mathematical models of an unloaded and loaded elastic element are proposed. It is established that as a result of deterioration of the lining a systematic relative error arises, which reaches 1.826%. The calibration of the monitoring device is proposed to be carried out with a lining



виникає систематична відносна похибка, яка досягає 1,826%. Запропоновано тарування контролюючого пристрою здійснювати при умовно зношеній наполовину футеровці. Тоді систематична похибка в процесі експлуатації не буде виходити за межі 0,983%, що допустимо в даних технологіях. *Ключові слова:* кульовий млин, моделювання, автоматичний контроль, енергоефективність, подрібнення руди, спрацювання, точність.

тическая относительная погрешность, которая достигает 1,826%. Предложено тарирование контролирующего устройства осуществлять при условно изношенной наполовину футеровке. Тогда систематическая ошибка в процессе эксплуатации не будет выходить за пределы 0,983%, что допустимо в данных технологиях. *Ключевые слова:* шаровая мельница, моделирование, автоматический контроль, энергоэффективность, измельчение руды, износ, точность.

that is conventionally worn out. Then the systematic error in the operation will not exceed 0.983%, which is acceptable in these technologies. *Key words:* ball mill, simulation, automatic control, energy efficiency, ore grinding, wear, precision.

УДК 004.412:519.237.5 / С.Б. Приходько, Н.В. Приходько, О.В. Спинов / Побудова рівняння нелінійної регресії для оцінювання розміру програмного забезпечення інформаційних систем з відкритим кодом на PHP // Проблеми інформаційних технологій. — 2018. — № 1 (23). — С. 118-125. Бібл.: 13 назв., англ.

УДК 004.412:519.237.5 / С.Б. Приходько, Н.В. Приходько, А.В. Спинов / Построение уравнения нелинейной регрессии для оценки размера программного обеспечения информационных систем с открытым кодом на PHP // Проблемы информационных технологий. — 2018. — № 1 (23). — С. 118-125. Библ.: 13 назв., англ.

UDC 004.412:519.237.5 / S.B. Prykhodko, N.V. Prykhodko, A.V. Spinov / Constructing the non-linear regression equation to estimate the software size of open source PHP-based information systems // Problems of information technologies. — 2018. — № 1 (23). — P. 118-125. Bibl: 13 nom., eng.

Рівняння, довірчі інтервали і інтервали прогнозування багатовимірної нелінійної регресії для оцінювання розміру програмного забезпечення інформаційних систем з відкритим кодом на PHP побудовані на основі багатовимірного нормалізуючого перетворення Джонсона. Виконано порівняння побудованого рівняння з лінійним та двома нелінійними регресійними рівняннями, що базуються на десятковому логарифмі і одновимірному перетворенні Джонсона.

Уравнение, доверительные интервалы и интервалы прогнозирования многомерной нелинейной регрессии для оценки размера программного обеспечения информационных систем с открытым кодом на PHP построены на основе многомерного нормализующего преобразования Джонсона. Выполнено сравнение построенного уравнения с линейным и двумя нелинейными регрессионными уравнениями, которые базируются на десятичном логарифме и одномерном преобразовании Джонсона.

The equation, confidence and prediction intervals of multivariate non-linear regression for estimating the software size of open-source PHP-based information systems are constructed on the basis of the Johnson multivariate normalizing transformation. Comparison of the constructed equation with the linear and two non-linear regression equations based on the decimal logarithm and the Johnson univariate transformation is performed.

УДК 004.942: 621.3.013/ Рябенський В.М., Чудайкін І.І., Таргунакова Ю.Д./Розробка програмного комплексу для розв'язання прямих задач морської магнітометрії//Проблеми інформаційних технологій. — 2018. — № 1 (023). — С.126-133: Бібл. 2 1 назв., укр.

УДК 004.942: 621.3.013/ Рябенский В.М., Чудайкин И.И., Таргунакова Ю.Д./Разработка программного комплекса для решения прямых задач морской магнитометрии//Проблемы информационных технологий. — 2018. — № 1 (023). — С.126-133: Библ. 21 назв., укр.

UDC 004.942: 621.3.013/ Ryabenkiy V.M., Chudaykin I.I., Targunakova J.D./Development of software for solving direct marine magneto-metric problems/ The problems of information technologies. — 2017. — № 1 (023). — P.126-133: Ref.: 21 titles, ukr.

У статті приводиться аналіз методів розв'язку польових задач для розрахунку магнітних полів. На підставі аналізу наявних проблем розрахунку магнітних полів тонкостінних конструкцій сформульовано та обґрунтовано необхідність розробки програмного комплексу для розв'язку прямої задачі магнітометрії — розрахунок магнітних полів типових збуджувальних тіл (наборами яких можна апроксимувати реальні об'єкти). У статті продемонстровано розв'язання ПЗММ за допомогою запропонованого програмного комплексу MBEM. Перераховано переваги MBEM. У статті приводиться чисельний розрахунок магнітного поля поблизу підводного човна, отриманий за допомогою розробленого комплексу MBEM. *Ключові слова:* чисельні методи для розрахунку магнітних полів; вектор магнітної напруженості; модифікований метод граничних елементів; програмний комплекс MBEM.

В статье приведен анализ методов решения полевых задач для расчета магнитных полей. На основании анализа имеющихся проблем расчета магнитных полей тонкостенных конструкций сформулирована и обоснована необходимость разработки программного комплекса для решения прямой задачи магнитометрии — расчет магнитных полей типовых возбуждающих тел (наборами которых можно аппроксимировать реальные объекты). В статье продемонстрировано решение ПЗММ при помощи предлагаемого программного комплекса MBEM. Перечислены преимущества MBEM. В статье приводится численный расчет магнитного поля вблизи подводной лодки, полученный с помощью разработанного программного комплекса. *Ключевые слова:* численные методы для расчета магнитных полей; вектор магнитной напряженности; модифицированный метод граничных элементов; программный комплекс MBEM.

The article gives an analysis of the methods for solving field problems for the calculation of magnetic fields. Based on the analysis of the existing problems of calculating the magnetic fields of thin-walled structures, formulated the necessity of developing a software package for solving the direct problem of magnetometry — calculation of magnetic fields of typical excitatory bodies (sets of which you can approximate real objects). The article demonstrates the solution direct marine magneto-metric problems with the help of the MBEM software package. Listed advantages MBEM. The article gives a numerical calculation of the magnetic field near the submarine, obtained with the help of the developed software package. *Keywords:* numerical methods for calculating magnetic fields; vector of magnetic intensity; modified boundary element method; MBEM software package.

УДК 004.386 / Яковенко В.О., Ульяновська Ю.В. / Аналіз задачі побудови автоматизованої системи виявлення та попередження надзвичайних ситуацій на підприємствах підвищеної небезпеки // Проблеми інформаційних технологій. – 2018. – № 1 (023). – С.134-139. Бібл. 11 назв., укр.

Розглядається задача розробки автоматизованої системи виявлення та запобігання надзвичайних ситуацій на підприємствах підвищеної небезпеки. Особливу увагу приділено схемі передачі даних від джерел первинної інформації до респондентів сповіщення з урахуванням інтелектуальної складової. Сформульовані вимоги та функціональні задачі системи. Запропонована структурна схема інтелектуальної системи сповіщення.

УДК 004.386 / В.А. Яковенко, Ю.В. Ульяновская / Анализ задачи построения автоматизированной системы обнаружения и предупреждения чрезвычайных ситуаций на предприятиях повышенной опасности // Проблеми інформаційних технологій. – 2018. – № 1 (023). – С.134-139. Бібл. 11 назв., укр.

Рассматривается задача разработки автоматизированной системы обнаружения и предотвращения чрезвычайных ситуаций на предприятиях повышенной опасности. Особое внимание уделено схеме передачи данных от источников первичной информации к респондентам оповещения с учетом интеллектуальной составляющей. Сформулированы требования и функциональные задачи системы. Предложена структурная схема интеллектуальной системы оповещения.

UDC 004.386 / V.Yakovenko, Yu. Ulianova / Analysis of the task of constructing automated detection and prevention of emergencies in enterprises of increased danger // Problems of information technologies. – 2018. – № 1 (023). – С.134-139. Bible: 11 titles, ua.

The task of developing an automated system for detecting and preventing emergencies in enterprises of high danger is considered. Particular attention is paid to the scheme of data transmission from the sources of primary information to the respondents of the notification, taking into account the intellectual component. Formulated requirements and functional tasks of the system. The proposed scheme of the intellectual system of notification is proposed.

УДК 504.7.064.3:614 / Козуля Т.В., Ведь О.В. / Огляд моделей гетерогенного каталізу // Проблеми інформаційних технологій. – 2018. – № 1 (023). – С.140-147. Бібл.: 12 назв., рос.

Математичне моделювання хімічних процесів і реакторів на даний час сформувалося в систему понять і методів, застосування яких дозволяє вирішити ряд важливих екологічних і науково-технічних проблем при розробці технологічних процесів і апаратів. Основою виробництва більшості хімічних і нафтохімічних продуктів є катализ. Проектування хімічних реакторів каталітичного типу передбачає розуміння процесів, що проходять в них. У даній роботі представлений огляд ряду моделей гетерогенного каталізу заснованих на структурованому, ієрархічному підході, суть якого в послідовному розв'язанні задач на різних рівнях опису процесів. *Ключові слова:* катализ, математичне моделювання, каталітичні процеси, моделі каталітичних систем.

УДК 504.7.064.3:614 / Козуля Т. В., Ведь Е. В. / Обзор моделей гетерогенного катализа // Проблеми інформаційних технологій. – 2018. – № 1 (023). – С.140-147. Бібл.: 12 назв., рус.

Математическое моделирование химических процессов и реакторов в настоящее время сформировалось в систему понятий и методов, применение которых позволило решить ряд важных экологических и научно-технических проблем при разработке технологических процессов и аппаратов. Основой производства большинства химических и нефтехимических продуктов является катализ. Проектирование химических реакторов каталитического типа предусматривает понимание процессов, проходящих в них. В данной работе представлен обзор ряда моделей гетерогенного катализа основанных на структурированном, иерархическом подходе, суть которого в последовательном решении задач на различных уровнях описания процессов. *Ключевые слова:* катализ, математическое моделирование, каталитические процессы, модели каталитических систем.

UDC 504.7.064.3:614 / Kozulya T. V., Ved O. V. / Models for Heterogeneous Catalysts // Problems of Information Technologies. – 2018. – № 1 (023). – С.140-147. Bible: 12 titles, rus.

Mathematical modeling of chemical processes and reactors has now been formed into a system of concepts and methods, the application of which has made it possible to solve a number of important environmental and scientific and technical problems in the development of technological processes and apparatuses. The basis for the production of most chemical and petrochemical products is catalysis. The design of catalytic-type chemical reactors involves understanding the processes that take place in them. In this paper we present an overview of a number of models of heterogeneous catalysis based on a structured, hierarchical approach, the essence of which is in the sequential solution of problems at different levels of the description of processes. *Keywords:* warehouse, products, information system, computer system, WMS system, methodology.

УДК 004.42+004.436 / Галкин О.В., Верес М.М., Ларин В. О., Бантыш О.В./ Використання предметно – орієнтованої мови і візуальних підходів для проектування системи акторів(АККА) // Проблеми інформаційних технологій. – 2018. – №1 (023). – С. 148-153. Бібл.: 10 назв., укр

Розглядаються особливості застосування Akka Framework для систем обробки транзакцій в реальному часі. Розроблено концептуальну модель системи Visual Akka, яка поєднує використання предметної мови та візуального підходу для створення програмних систем в рамках Akka. *Ключові слова:* модель актора, Akka, DSL.

УДК 004.42 + 004.436 / Галкин А.В., Верес Н.Н., Ларин В. А., Бантыш А.В. / Использование предметно - ориентированного языка и визуальных подходов для проектирования системы акторов (АККА) // Проблеми інформаційних технологій. – 2018. – №1 (023). – С.148-153.. Бібл.: 10 назв., рус

Рассматриваются особенности применения Akka Framework для систем обработки транзакций в реальном времени. Разработана концептуальная модель системы Visual Akka, которая сочетает использование предметного языка и визуального подхода для создания программных систем в рамках Akka. *Ключевые слова:* модель актора, Akka, DSL.

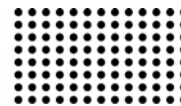
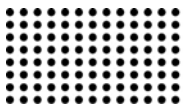
Galkin O.V., Veres M.M, Larin V.O., Bantysh O.V. / Use of domain-specific and visual approaches for actor system development(AKKA) // Problems of information technologies. – 2018. – No. 1 (023). – С. 148-153 Bible: 10 titles, eng

The features of the Akka Framework application for real-time transaction processing systems are considered. The conceptual model of Visual Akka system, which combines use of a domain-specific and a visual approach for creating program systems in Akka framework, was formulated as part of the project. *Keywords:* model actor, Akka, DSL.

УДК 004.94 / Везумський О.К. / ХАОС ФРАКТАЛ БИРЖА // Проблеми інформаційних технологій. – 2018. – №1 (023). – С.154-159. Бібл.: 6 назв., рос.

УДК 004.94 / Везумский А.К. / ХАОС ФРАКТАЛ БИРЖА // Проблеми інформаційних технологій. – 2018. – №1 (023). – С.154-159. Бібл.: 6 назв., рус.

UDK 004.94 / Vezumskiy A.K. / CHAOS FRACTAL EXCHANGE // Problems of information technologies. – 2018. – No. 1 (023). – С.154-159. The bible: 16names, rus



У статті розглядаються поняття фракталу та його атрибутів, питання використання фрактального аналізу та фрактальної геометрії що до моделювання та прогнозування біржевих котировок з метою підвищення прибутку гравця. Досліджені методи технічного та фундаментального аналізу та альтернативні їм фрактальні методи.

UDK 004.91 / Шевчук М.М., Юсин Я.О., Заболотня Т.М., Рыбачок Н.А., Дичка А.І. / Підходи до оброблення «синтаксичного цукру» при пошуку плагиату в програмному коді // Проблеми інформаційних технологій. – 2018. – №1(023). – С.160-165. Бібл.: 10 назв, укр.

Розглянуто питання оброблення «синтаксичного цукру» при пошуку плагиату в програмному коді. Запропоновано два нових підходи до попереднього оброблення «синтаксичного цукру», що можуть бути застосовані перед виконанням пошуку та не потребують структурного чи машинного подання програмного коду. Наведено перелік конструкцій «синтаксичного цукру» у мові C#. *Ключові слова:* пошук плагиату, програмний код, «синтаксичний цукор», C#.

UDK 681.3.093:044.3 / Зуев Д.О., Дос Е.В., Кропачев А.В., Бабкин О.В., Варламов О.О. / Менеджмент электропитания облачных серверных кластеров // Проблеми інформаційних технологій. – 2018. – №1(023). – С.166-172. Бібл.: 16 назв, англ.

Проаналізовано управління електроживленням фізичних ресурсів серверних кластерів. Було показано, що низька продуктивність роботи інфраструктури центру обробки даних пов'язана з диспропорцією використання серверів. Проблема надмірного споживання може бути вирішена шляхом мінімізації числа активних серверів в рамках процедури консолідації сервера. При забезпеченні консолідації серверів, необхідно підтримувати прийнятний рівень продуктивності роботи інфраструктури серверів. Консолідація серверів може привести до погіршення продуктивності центру обробки даних через конфлікт використання загальних ресурсів віртуальними машинами. Було продемонстровано, що пороговий рівень режиму використання мережевих ресурсів повинен ґрунтуватися на тому, щоб досягти компромісу між стабільною роботою центру обробки даних і можливістю економії електроенергії, що пов'язано з виключенням з аналізу нехарактерних випадків пікового навантаження серверів. Базова схема віртуалізації з підтримкою зіставлення на основі кластерів включає в себе рівень відстеження використання фізичних ресурсів серверів центрів обробки даних, перетворення отриманої залежності в двійкову послідовність, кластеризацію віртуальних машин і

В статье рассматриваются понятие фрактала и его атрибуты, вопросы использования фрактального анализа и фрактальной геометрии для моделирования и прогнозирования биржевых котировок с целью повышения прибыли игрока. Исследованы методы технического и фундаментального анализа и альтернативные им фрактальные методы.

UDK 004.91 / Шевчук М.М., Юсин Я.А., Заболотня Т.Н., Рыбачок Н.А., Дычка А.И. / Подходы к обработке «синтаксического сахара» при поиске плагиата в программном коде // Проблемы информационных технологий. – 2018. – №1(023). – С.160-165 Библ.: 10 назв, укр.

Рассмотрен вопрос обработки «синтаксического сахара» в процессе поиска плагиата в программном коде. Предложены два новых подхода к предварительной обработке «синтаксического сахара», которые могут быть применены перед выполнением поиска и не требуют структурного или машинного представления программного кода. Приведен перечень конструкций «синтаксического сахара» языка C#. *Ключевые слова:* поиск плагиата, программный код, «синтаксический сахар», C#.

UDK 681.3.093:044.3 / Зуев Д.О., Дос Е.В., Кропачев А.В., Бабкин О.В., Варламов А.А. / Менеджмент электропитания оборудования серверных кластеров // Проблеми інформаційних технологій. – 2018. – №1(023). – С.166-172: Бібл.: 16 назв, англ.

Проанализировано управление электропитанием физических ресурсов серверных кластеров. Было показано, что низкая производительность работы инфраструктуры центра обработки данных связана с диспропорцией использования серверов. Проблема избыточного потребления может быть решена путем минимизации числа активных серверов в рамках процедуры консолидации сервера. При обеспечении консолидации серверов, необходимо поддерживать приемлемый уровень производительности работы инфраструктуры серверов. Консолидация серверов может привести к ухудшению производительности центра обработки данных из-за конфликта использования общих ресурсов виртуальными машинами. Было продемонстрировано, что пороговый уровень режима использования сетевых ресурсов должен основываться на том, чтобы достичь компромисса между стабильной работой центра обработки данных и возможностью экономии электроэнергии, что связано с исключением из анализа нехарактерных случаев пиковой нагрузки серверов. Базовая схема виртуализации с поддержкой сопоставления на основе кластеров включает в себя уровень отслеживания использования физических ресурсов серверов центров обработки данных, преобразование

The article deals with the concept of fractal and its attributes, the use of fractal analysis and fractal geometry for modeling and forecasting stock quotes in order to increase the player's profit. Research methods of technical and fundamental analysis and alternative fractal methods.

UDC 004.91 / Shevchuk M., Yusyn Y., Zabolotnia T., Rybachok N., Dychka A. / The approaches to the "syntactic sugar" processing in the search for plagiarism in the program code // The problems of information technologies. – 2018 – №1 (023). – P.160-165. Bibl.: 10 titles, ukr.

The issue of «syntactic sugar» processing in the search for plagiarism in the program code is considered. Two new approaches to the preliminary processing of «syntactic sugar», which can be applied before the search is performed and don't require a structural or machine representation of the program code, are proposed. The list of «syntactic sugar» constructions of C# language is given. *Keywords:* search for plagiarism, program code, syntactic sugar, C#.

UDC 681.3.093:044.3 / Zuev D.O., Dos E.V., Kropachev A.V., Babkin O.V., Varlamov A.A. / Power management for server clusters hardware // Problems of information technologies. – 2018. – №1(023). P.166-172: Ref.: 16 titles, eng.

Power management for server clusters physical resources was analyzed. It was shown that low performance of data center infrastructure work refers to disproportion of servers' utilization. Overconsumption problem could be solved by minimization of the active servers' number within the bounds of the server consolidation procedure. In order to provide server consolidation implementation it is necessary to maintain acceptable performance level of the servers room infrastructure work. Server consolidation may cause performance degradation due to the conflict of using shared resources by virtual machines. It was demonstrated that threshold level of utilization regime analysis should be based in order to get a compromise between stable work of data center and opportunity for power savings which is associated with skipping of rare cases of servers' peak load. Basic scheme of clustering-based correlation-aware virtualization includes trace data center servers' physical resources utilization level, transformation of utilization traces into binary sequence up to the utilization threshold value, clustering of virtual machines up to the binary sequence in order to maintain not overlapping of different clusters and virtual machines allocation at physical servers in order to minimize the possibility of the service performance degradation at peak period. It was develop power management procedure which

розподіл віртуальних машин на фізичних серверах, щоб звести до мінімуму можливість зниження продуктивності служби в піковий період. При цьому важливо оцінити коефіцієнт кореляції між використанням фізичного ресурсу серверів віртуальними машинами. Було показано, що коефіцієнт кореляції Пірсона є оптимальним інструментом оцінки кореляції використання фізичних ресурсів віртуальними машинами серверів центру обробки даних. Розроблена на його основі модель дозволяє зберігати всі вибірки і рівномірно розподіляти обчислювальні ресурси, а також показує кореляційну залежність в межах певного періоду часу. *Ключові слова:* центр обробки даних, енергоспоживання, віртуальна машина, кластеризація, коефіцієнт кореляції Пірсона, погіршення продуктивності, пікове навантаження.

полученной зависимости в двоичную последовательность, кластеризацию виртуальных машин и распределение виртуальных машин на физических серверах, чтобы свести к минимуму возможность снижения производительности службы в пиковый период. При этом важно оценить коэффициент корреляции между использованием физического ресурса серверов виртуальными машинами. Было показано, что коэффициент корреляции Пирсона является оптимальным инструментом оценки корреляции использования физических ресурсов виртуальными машинами серверов центра обработки данных. Разработанная на его основе модель позволяет хранить все выборки и равномерно распределять вычислительные ресурсы, а также показывает корреляционную зависимость в границах определенного периода времени. *Ключевые слова:* центр обработки данных, энергопотребление, виртуальная машина, кластеризация, коэффициент корреляции Пирсона, ухудшение производительности, пиковая нагрузка.

consists from user-interactive and fast changing service, maintaining of the minimal performance degradation caused by physical resources sharing conflict and high correlation level of virtual machines. Thereby it is important to estimate proper measure to quantify the correlation coefficient between virtual machines to overcome the inefficiency of the conventional correlation metric. Pearson's correlation coefficient was proved to be optimal instrument of the correlation of used data center virtual machines physical resources utilization quantifying. Developed model allows storing all samples and evenly distributing computational utilization as well as correlation between the events in the bounds of certain time period. *Keywords:* data center, power consumption, virtual machine, clustering, Pearson's correlation coefficient, performance degradation, peak load.

УДК 004.658: 652.3 / Дворецкий М. Л., Дворецкая С. В., Давыденко Е. О. / Розробка системи управління знаннями організації на базі CMS WORDPRESS // Проблеми інформаційних технологій. – 2018. – №1 (023). С. 173-180: Бібл.: 19 назв., укр.

УДК 004.658: 652.3 / Дворецкий М. Л., Дворецкая С. В., Давыденко Е. А. / Разработка системы управления знаниями организации на базе CMS WORDPRESS // Проблемы информационных технологий. – 2018. – №1 (023). С. 173-180: Библ.: 19 назв., укр.

UDC 004.658: 652.3 / Dvortskiy M. L., Dvortskaya S. V., Davydenko Ye. O. / Development of the organization knowledge management system based on CMS WORDPRESS // The problems of information technology. – 2018. – No.1 (023). P. 173-180: Ref.: 19 titles, ukr.

Дослідження розглядає проблеми представлення, зберігання і подальшого використання знань у межах організації, набутих її співробітниками завдяки фактичному досвіду виконання поточних завдань. Розглянуто загальні питання управління та обміну знаннями, підходи щодо створення інформаційних порталів та обґрунтовано необхідність у структурованні знань (матеріалів) для подальшої ефективної обробки та пошуку. Додаткова увага приділяється зворотному зв'язку із кінцевими користувачами системи та наявності підсистеми мотивації. *Ключові слова:* управління знаннями, веб-застосунок, AJAX, веб-сервер, веб-фреймворк, система управління контентом, система керування базами даних, WordPress, WAMP, PHP, Apache, MySQL.

Исследование рассматривает проблемы представления, хранения и дальнейшего использования знаний в рамках организации, приобретенных ее сотрудниками благодаря фактическому опыту выполнения текущих задач. Рассмотрены общие вопросы управления и обмена знаниями, подходы к созданию информационных порталов и обоснована необходимость в структурировании знаний (материалов) для дальнейшей эффективной обработки и поиска. Дополнительное внимание уделяется обратной связи с конечными пользователями системы и наличию подсистемы мотивации. *Ключевые слова:* управление знаниями, веб-приложение, AJAX, веб-сервер, веб-фреймворк, система управления контентом, система управления базами данных, WordPress, WAMP, PHP, Apache, MySQL.

The research overlooks problems of presenting, storing and future using of knowledge within the organization, which were received by its employees according to actual experience in current tasks performing. General tips of knowledge management, knowledge sharing and information portals creation approaches are considered. It indicates the necessity of knowledge structuring for its next effective processing and search. The additional attention to the feedback to system users and existence of motivation modules is paid. *Keywords:* knowledge management, web-application, AJAX, web-server, web-framework, content management system, database management system, WordPress, WAMP, PHP, Apache, MySQL.

УДК 004.021:004.85 / Журавська І. М. / Генерація суб-оптимальних маршрутів безпілотного літального апарата з використанням нейронної мережі Хопфілда // Проблеми інформаційних технологій. – 2018. – №1 (023). С. 181-185. Бібл.: 11 назв., укр.

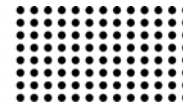
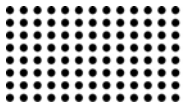
УДК 004.021:004.85 / Журавская И. Н. / Генерація суб-оптимальних маршрутів безпілотного летального апарата з використанням нейронної мережі Хопфілда // Проблеми інформаційних технологій. – 2018. – №1 (023). С. 181-185. Бібл.: 11 назв., укр.

UDC 004.021:004.85 / Zhuravska I. M. / Generation of sub-optimal routes for UAV via Hopfield neural network // The problems of information technology. – 2018. – No.1 (023). P. 181-185. Ref.: 11 titles, ukr.

У статті розглянуто проблему вирішення задачі комівояжера з допомогою нейронної мережі Хопфілда (НМХ) для генерації суб-оптимального маршруту безпілотного літального апарата (БПЛА). Розглянуто принцип задачі комівояжера для заданих умов: БПЛА має відвідати кожен вузол один раз, оптимальним шляхом вважається найкоротший, після завершення обльоту БПЛА має

В статье рассмотрена проблема решения задачи коммивояжера с помощью нейронной сети Хопфилда для генерации суб-оптимальных маршрутов беспилотного летального аппарата (БПЛА). Рассмотрена задача коммивояжера для заданных условий: БПЛА должен посетить каждый узел один раз, оптимальным путем считается самый короткий, после завершения облета БПЛА

This article is devoted to solving the traveling salesman problem using Hopfield neural network to generate the sub-optimal routes. Specifics of traveling salesman problem for an unmanned aerial vehicle (UAV) were studied and cited in the article. The traveling salesman problem for studying the subject has such characteristics: UAV must visit each node once; the best path is the shortest; after the flyby UAV must return to the point of



повернутись у точку вильоту. Досліджено загальну архітектуру НМХ та її особливості при реалізації задачі комівояжера. На основі проведених досліджень, розробленого алгоритму та програмного додатку окреслені переваги та недоліки використання НМХ для вирішення задачі генерації маршрутів БПЛА. *Ключові слова:* БПЛА, нейронна мережа Хопфілда, задача комівояжера, суб-оптимальний маршрут, вузол.

должен вернуться в точку вылета. Исследована общая архитектура нейронной сети Хопфилда (НМХ) и ее особенности при реализации задачи коммивояжера для полёта БПЛА. На основе проведенных исследований, разработанного алгоритма и программного приложения выделены преимущества и недостатки использования НМХ для решения задачи генерации маршрутов БПЛА. *Ключевые слова:* БПЛА, нейронная сеть Хопфилда, задача коммивояжера, суб-оптимальный маршрут, узел.

departure. A general structure of Hopfield neural network (HNN) for UAV's flight was learned and described. It was discovered that HNN has some distinctive features in case of solving of the traveling salesman problem by this way for generating UAV routes. The algorithm was tested on basis of the research and the application realizing the task. The routes generated by the program are approximately the same length, but they look completely different. The point is that the HNN can only solve the traveling salesman problem approximately (the result is usually a little longer than the ideal value). Therefore, the routes generated using HNN are always approximately the same in length, but different in the order of passing nodes. Basic advantages, deviations, and disadvantages of the proposed approach were determined. The risks that can be faced in applying the proposed approach are analyzed: multiple UAVs may be in the same place at the same time; height of the UAV movement can be too small for maneuvering. *Key-words:* UAV, Hopfield neural network, traveling salesman problem, sub-optimal route, node.

УДК 004.8, 004.94 / Б.А. Салтан, Д.А. Собко, І.В. Кулаковська / Математична модель алгоритму вибору зброї ботом в комп'ютерній грі жанру шутер // Проблеми інформаційних технологій. – 2018. – № 1 (023). – С.186-193; Бібл.: 10 назв., укр.

УДК 004.8, 004.94 / Б.А. Салтан, Д.А. Собко, И.В. Кулаковская / Математическая модель алгоритма выбора оружия ботом в компьютерной игре жанра шутера // Проблемы информационных технологий. – 2018. – № 1 (023). – С.186-193; Библ.: 10 назв., укр.

UDC 004.8, 004.94 / B.A. SALTAN, D.A. SOBKO, I.V. KULAKOVSKA / Mathematical model for algorithm of election of weapons with bot in computer games shooter // Problems of Information Technologies. – 2018. – № 1 (023). – С.186-193; Bible: 10 titles, ua.

Пропонується розроблена математична модель алгоритму вибору зброї ботом в комп'ютерній грі жанру шутер з використанням сценарного аналізу, що наближає поведінку ШІ до людської у ситуаціях пов'язаних із вибором засобу ураження та прогнозуванням переміщення противника, що дозволяє гравцю отримати від гри досвід високої якості. Розроблений інструментарій зорієнтований на подальшу розробку та оптимізацію ігрового процесу який покликаний покращити інтерактивність та степінь реалізму у комп'ютерних іграх. Наведені проектні рішення щодо розробленого моделі; проводиться узагальнення проведеної роботи та отриманих результатів, визначаються перспективи подальшого навчання ШІ з використанням нейронних мереж. Математична модель, що побудована в дані роботі, дає змогу визначати параметри керування, а разом з ним і стратегії, які регулюють рівень складності гри і дають змогу гравцеві отримати перемогу, при правильному виборі його власної стратегії. *Ключові слова:* математична модель, комп'ютерна гра, параметри керування, алгоритми вибору, сценарний аналіз, стратегії в шутері.

Предлагается разработанная математическая модель алгоритма выбора оружия ботом в компьютерной игре жанра шутер с использованием сценарного анализа, что приближает поведение ИИ к человеческому в ситуациях связанных с выбором средства поражения и прогнозированием перемещения противника, что позволяет игроку получить от игры опыт высокого качества. Разработанный инструментари ориентирован на дальнейшую разработку и оптимизации игрового процесса который призван улучшить интерактивность и степень реализма в компьютерных играх. Приведенные проектные решения по разработанного модели; проводится обобщение проведенной работы и полученных результатов, определяют перспективы дальнейшего обучения ИИ с использованием нейронных сетей. Математическая модель, построенная в данной работе, позволяет определять параметры управления, а вместе с ним и стратегии, которые регулируют уровень сложности игры и позволяют игроку одержать победу, при правильном выборе его собственной стратегии. *Ключевые слова:* математическая модель, компьютерная игра, параметры управления, алгоритмы выбора, сценарный анализ, стратегии в шутере.

Research and development of computer games using neural networks. The Paper is aimed at research and development of a computer game - a simulator, the result of which is an AI, trained with the help of neural networks, the behavior of which is as close as possible to human in situations associated with the choice of the means of defeat and prediction of the movement of the enemy. That allows a player to get high quality experience from the game. The proposed mathematical model of the algorithm for selecting a weapon bot in a computer genre shooter game using scenario analysis that approximates the behavior of AI to human in situations involving the choice of a means of defeat and prediction of the movement of the enemy, which allows the player to obtain from the game the experience of high quality. The developed toolkit is focused on the further development and optimization of the game process, which is designed to improve the interactivity and degree of realism in computer games. The design decisions concerning the developed model are given; the generalization of the work performed and the results obtained, the prospects for further study of AI using neural networks are determined. The mathematical model built into the given work allows you to determine the control parameters, and with it the strategies that regulate the level of complexity of the game and allow the player to win, with the right choice of his own strategy. *Key words:* mathematical model, computer game, control parameters, choice algorithms, scenario analysis, strategy in a shooter.

УДК 004.056.5 / О.О. Боскін, С.Р. Мазманян, А.М.Левіцька / Аналіз безпеки інтерфейсу 802.11 (Wi-Fi з'єднань), класифікація паролів // Проблеми інформаційних технологій. – 2018. – № 1 (023). – С.194-204: Бібл.: 11 назв, укр.

Проаналізовано надійність алгоритму шифрування WPA/WPA2, на основі запропонованих авторами стереотипів проведена класифікація та кластеризація паролів, які були застосовані користувачами при аутентифікації в бездротових мережах стандарту 802.11. Ключові слова: Безпека інтерфейсу, WPA, Wi-Fi, 802.11, пароль, стереотип паролю, olcHashcat.

УДК 004.056.5 / О.О. Боскин, С.Р. Мазманян, А.М.Левіцька / Анализ безопасности интерфейса 802.11 (Wi-Fi соединений), классификация паролей // Проблемы информационных технологий. – 2018. – № 1 (023). – С.194-204: Библ.: 11 названий, укр.

Проанализированы надежность алгоритма шифрования WPA / WPA2, на основе предложенных авторами стереотипов проведена классификация и кластеризация паролей, которые были применены пользователями при аутентификации в беспроводных сетях стандарта 802.11. Ключевые слова: Безопасность интерфейса, WPA, Wi-Fi, 802.11, пароль, стереотип пароля, olcHashcat.

UDC 004.056.5 / O.O. Boskin, S.R. Mazmanyanyan, A.M. Levitskaya / Analysis of interface security 802.11 (Wi-Fi connections), classification of passwords // Problems of information technologies. – 2018 - No. 1 (023). - P.194-204: Bibl.: 11 titles, Ukr.

The reliability of the WPA / WPA2 encryption algorithm has been analyzed. Based on the stereotypes proposed by the authors, classification and clustering of passwords that were used by users during authentication in wireless networks of the 802.11 standard were carried out. Keywords: Interface Security, WPA, Wi-Fi, 802.11, password, password stereotype, olcHashcat.

УДК 004.89:519.8 / Горбач Т.В., Славгородський В.Ю., Шубін І.Ю., Ковалевська А.В. / Моделі електронного навчання та вимоги до програмного забезпечення // Проблеми інформаційних технологій. – 2018. – №1 (23). – С. 205-218. Бібл.: 10 назв.

Розглядаються особливості моделей електронного навчання та вимоги до програмного забезпечення, що використовуються в системі дистанційного навчання. Особливу увагу приділено використанню мікросервісів і еталонної моделі об'єкта контенту для спільного використання – Sharable Content Object Reference Model.

УДК 004.89:519.8 / Горбач Т.В., Славгородский В.Ю., Шубин И.Ю., Ковалевская А.В. / Модели электронного обучения и требования к программному обеспечению // Проблемы информационных технологий. – 2018. – №1 (23). – С. 205-218. Библ.: 10 назв.

Рассматриваются особенности моделей электронного обучения и требования к программному обеспечению, используемых в системе дистанционного обучения. Особое внимание уделено использованию микросервисов и эталонной модели объекта контента для совместного использования – Sharable Content Object Reference Model.

УДК 004.89:519.8 / Horbach T.V., Slavhorodskiy V.Yu., Shubin I.Yu., Kovalevska A.V. / E-learning models and software requirements // Problems of Information Technology. – 2018. – №1 (23). – P. 205-218. Bibl.: 10 names.

Features of e-learning models and software requirements used in the distance learning system are considered. Particular attention is paid to the use of microservices and a reference model of the content object for sharing – Sharable Content Object

УДК 004.02:519.8 / Ткаченко В.В., Череди́нченко О.Ю., Вовк М.А., Ершова С.І. / Підхід до збору інформації щодо екологічної обстановки при виникненні надзвичайних ситуацій техногенного характеру // Проблеми інформаційних технологій. – 2018. – №1 (23). – С.219-226: Бібл.: 17 назв., укр.

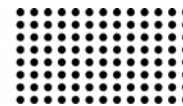
Сучасне управління наслідками промислових аварій, стихійних лих та усіх видів надзвичайних ситуацій базується на інтенсивному використанні інформаційних технологій. За допомогою інформаційних технологій визначаються управлінські дії та пов'язані з цим ресурси, які розгортаються там, де це необхідно. Проблема організації збору, обробки та аналізу даних є однією з найбільш актуальних і невирішених проблем сьогодні. Проведено аналіз існуючих підходів щодо підтримки рішень в системах екологічного моніторингу. Збір, узагальнення, обробка та видача інформації про екологічну обстановку організовується та здійснюється з метою прогнозування наслідків надзвичайних ситуацій техногенного характеру та своєчасного інформування відповідних сил реагування. Особливо важливою є задача оперативного визначення меж небезпечної зони та локалізація зони найвищої концентрації небезпечних речовин. Якість прийняття рішень, а також масштаби наслідків надзвичайної ситуації, суттєво залежать від часу збору екологічних даних. Сучасні прилади та обладнання дозволяють вимірювати та передавати моніторингові дані до центру обробки. Слабким місцем залишається розміщен-

УДК 004.02:519.8 / Ткаченко В.В., Череди́нченко О.Ю., Вовк М.А., Ершова С.И. / Подход к сбору информации по экологической обстановке при возникновении чрезвычайных ситуаций техногенного характера // Проблемы информационных технологий. – 2018. – №1 (23). – С. 219-226: Библ.: 17 назв., укр.

Современное управление последствиями промислових аварій, стихійних бедствий и всех видов чрезвычайных ситуаций базируется на интенсивном использовании информационных технологий. С помощью информационных технологий определяются управленческие действия и связанные с этим ресурсы, которые разворачиваются там, где это необходимо. Проблема организации сбора, обработки и анализа данных является одной из наиболее актуальных и нерешенных проблем сегодня. Проведен анализ существующих подходов по поддержке решений в системах экологического мониторинга. Сбор, обобщение, обработка и выдача информации об экологической обстановке организуется и осуществляется с целью прогнозирования последствий чрезвычайных ситуаций техногенного характера и своевременного информирования соответствующих сил реагирования. Особенно важной является задача оперативного определения границ опасной зоны и локализация зоны высокой концентрации опасных веществ. Качество принятия решений, а также масштабы последствий чрезвычайной ситуации, существенно зависят от времени сбора экологических данных. Современные приборы и оборудова-

UDC 004.02:519.8 / Tkachenko V.V., Cherednichenko O.Yu., Vovk M.A., Ershova S.I. / An approach to information collection about the environmental situation in the case of anthropogenic emergency. – 2018. – №1 (23). – P. 219-226: Bibl.: 17.

The task of the operative determination of the danger zone borders and the localization of the zone of the highest concentration of hazardous substances is especially important. The quality of decision-making, as well as the magnitude of the consequences of an emergency, is significantly dependent on the time of environmental data collection. Modern devices and equipment allow measuring and transmitting monitoring data to the processing center. The bottleneck is the placement and operative movement of measuring equipment. The paper proposes the use of deformed configuration method to determine the point sequence of environmental parameter measurement. The management of the search process is carried out by choosing the locally optimal direction of the center shift of the simplex and selecting the value of the step, which provides optimization of the function in the center of the simplex. This provides an opportunity to determine the direction of movement of mobile measuring devices and allows to reduce the time of the finding the places with the highest level of dangerous substance due to the convergence of the algorithm for a finite number of steps. The ob-



ня та оперативне пересування вимірювального обладнання. В роботі пропонується для визначення послідовності точок вимірювання екологічних параметрів застосування методів деформованих конфігурацій, які використовують правильні симплекси в якості основної конфігурації. Управління процесом пошуку в таких методах здійснюється за рахунок вибору локально оптимального напрямку зсуву центру симплекса і вибору величини кроку, що забезпечує оптимізацію функції в центрі симплекса. Це надає можливість визначити напрямок руху пересувних вимірювальних приладів, що дозволяє скоротити час локалізації місць з найвищим рівнем небезпечної речовини за рахунок збіжності алгоритму за кінцеве число кроків. Отримані оперативні дані використовуються для первинної оцінки рівня екологічної безпеки та класифікації надзвичайної ситуації.

УДК 004.9 / Яшина О.М. / Порівняльна характеристика актуальних засобів розробки під мобільні платформи // Проблеми інформаційних технологій. – 2018. – 1(018). – С.227-233: Бібл.:10 назв., укр.

Здійснено порівняння актуальних засобів розробки під мобільні платформи. Вказано переваги та недоліки різних засобів розробки для різних мобільних платформ. Також обґрунтовується доцільність використання нативних та гібридних додатків та засобів їх розробки. *Ключові слова:* програмний продукт, інформаційні технології, програмне забезпечення, мобільний додаток, мобільна платформа, Android, iOS.

УДК 004.8 /Ходаков В.Е., Яцюк С.В. /Концептуальні основи побудови продукційних систем експертного оцінювання соціально-економічних систем //Проблеми інформаційних технологій. – 2018. - № 1 (023). – С.234-240: Бібл.: 10 назв., рос.

У даній роботі розглянута важливість ефективного вкладу інвестицій і розподілу бюджету між регіонами. Виділені показники, які визначають інвестиційну привабливість регіонів. Продемонстровано створення продукційної бази знань на прикладі спрощеної моделі показників привабливості регіонів і застосування методів зниження потужності бази знань для спрощення пошуку відповіді. Наданий опис алгоритму роботи експертної системи, яка використовує представлену базу знань. *Ключові слова:* інвестиційна привабливість, база знань, експертна система, дерево рішень, потужність баз знань.

ние позволяют измерять и передавать мониторинговые данные в центр обработки. Слабым местом остается размещение и оперативное передвижение измерительного оборудования. В работе предлагается для определения последовательности точек измерения экологических параметров применение методов деформированных конфигураций, которые используют правильные симплексы в качестве основной конфигурации. Управление процессом поиска в таких методах осуществляется за счет выбора локально оптимального направления смещения центра симплекса и выбора величины шага, что обеспечивает оптимизацию функции в центре симплекса. Это дает возможность определить направление движения передвижных измерительных приборов, что позволяет сократить время локализации мест с высоким уровнем опасного вещества за счет сходимости алгоритма за конечное число шагов. Полученные оперативные данные используются для первичной оценки уровня экологической безопасности и классификации чрезвычайной ситуации.

УДК 004.9 / Яшина О.М. / Сравнительная характеристика актуальных средств разработки под мобильные платформы // Проблемы информационных технологий. – 2018. – 1(018). – С. 227-233: Библ.: 10 назв., укр.

Осуществлено сравнение актуальных средств разработки под мобильные платформы. Указано достоинства и недостатки средств разработки под мобильные платформы. *Ключевые слова:* программный продукт, информационные технологии, программное обеспечение, Android, iOS, мобильное приложение, мобильная платформа.

УДК 004.8 /Ходаков В.Е., Яцюк С.В. /Концептуальные основы построения продукционных систем экспертного оценивания социально-экономических систем //Проблемы информационных технологий. – 2018. № 1(023). – С.234-240: Библ.: 10 назв., русс.

В данной работе рассмотрена важность эффективного вложения инвестиций и распределения бюджета между регионами. Выделены показатели, определяющие инвестиционную привлекательность регионов. Продемонстрировано создание продукционной базы знаний на примере упрощенной модели показателей привлекательности регионов и применение методов снижения мощности базы знаний для упрощения нахождения ответа. Описан алгоритм работы экспертной системы, использующей представленную базу знаний. *Ключевые слова:* инвестиционная привлекательность, база знаний, экспертная система, дерево решений, мощность базы знаний.

tained operational data are used for the initial assessment of the level of environmental safety and the classification of the emergency situation.

UDC 004.9 /Oxana Yashyna /Comparative characteristic of actual development tools for mobile platforms // The problems of information technologies. – 2018. – № 1 (023). – P.227-233: Bible: 10 titles, ua.

Comparison of actual development tools for mobile platforms is carried out. The advantages and disadvantages of various development tools for different mobile platforms are stated. It also justifies the feasibility of using native and hybrid applications and their development tools. *Keywords:* software product, information technology, software, Android, programming language, mobile application, iOS, mobile platform.

UDC 004.8 /Khodakov V.Ye., Yatsiuk S.V. /Conceptual basis of construction of production systems of expert estimation of socio-economic systems //The problems of information technologies. – 2018. – № 1 (023). – P.234-240: Ref.: 10 titles, rus.

In this work, the importance of the effective investment and budget allocation between regions, were considered. Indicators defining the investment attractiveness of the regions were singled out. The creation of the knowledge base based on a simplified model of regional attractiveness indicators and the application of methods to reduce the capacity of the knowledge base to simplify the finding of the answer were demonstrated. The algorithm of how works the expert system, which uses the presented knowledge base, was described. *Keywords:* investment attractiveness, knowledge base, expert system, decision tree, knowledge base capacity.

УДК 004.82 004.622 519.766 / Катеринич Л.О., Петелько Ю.Ю. / Алгоритм пошуку зв'язків і залежностей у даних веб-сторінок // Проблеми інформаційних технологій. – 2018. – № 1 (023). – С.241-247 Бібл.: 10 назв., укр.

Значне збільшення числа користувачів Інтернет призвело до збільшення обсягу доступної інформації. Така інформація може бути корисна, однак отримання такої інформації і пошук взаємозв'язку між інформацією, що міститься у великій кількості «сирих» даних, є складним завданням. Дослідження спрямовано на аналіз методів отримання даних, алгоритму пошуку залежностей у даних веб-сторінок і досліджень інструментарію, який забезпечує можливість реалізації алгоритму. Основна ідея - описати способи отримання даних. **Ключові слова:** очистка даних, інтеграція даних, вибір даних, інтелектуальний аналіз даних, оцінка шаблонів.

УДК 004.94 / Лєпа Є.В. / Моделювання корпоративної комп'ютерної мережі // Проблеми інформаційних технологій. – 2018. – №1 (023). – С.248-252: Бібл.: 10 назв., рус.

У статті розглядається імітаційна модель локальної комп'ютерної мережі в системі моделювання Riverbed Modeler. Досліджена пропускна здатність мережі, виконаної по стандартах 10Baset і 100Baset. Визначена ширина каналу Internet для оптимального співвідношення затримка/вартість.

УДК 004.652.4 : 004.4'22 / Колесник Л.В., Кириченко Н.А., Костоглот І.В. / Розробка засобу проектування високонавантажених реляційних систем зберігання даних: оптимізація структури та запитів SQL // Проблеми інформаційних технологій. – 2018. – № 1 (023). – С.253-260: Бібл.: 12 назв., укр.

Дана стаття присвячена темі проектування та перепроєктування високонавантажених реляційних систем зберігання даних. У статті будуть розглянуті головні методи та засоби для пришвидшення роботи однієї з найпопулярніших СУБД – СУБД MySQL. Будуть розглянуті методи оптимізації структури бази даних та SQL запитів. На основі розглянутих засобів було розроблено та протестоване програмне забезпечення для оптимізації швидкості роботи бази даних. Програмне забезпечення дозволить проводити процес оптимізації як в напівавтоматичному, так і в автоматичному режимах. На основі проведених тестів буде дана оцінка автоматичному процесу оптимізації бази даних. **Ключові слова:** СУБД, MySQL, база даних, SQL, структура, оптимізація, автоматизація,

УДК 004.82 004.622 519.766 / Катеринич Л.А., Петелько Ю.Ю. / Методика прийняття рішення по месту расположения продукции в складских помещениях // Проблемы информационных технологий. – 2018. – № 1 (023). – С.241-247: Библ.: 10 назв., укр.

Значительное увеличение числа пользователей Интернет привело к увеличению объема доступной информации. Такая информация может быть полезна, однако извлечение такой информации и поиск взаимосвязи между информацией, содержащейся в большом количестве «сырых» данных, являются сложной задачей. Исследование направлено на анализ методов извлечения данных, алгоритма поиска зависимостей в данных веб-страниц и исследованной инструментария, который обеспечивают возможность реализации алгоритма. Основная идея - описать методы извлечения данных. **Ключевые слова:** очистка данных, интеграция данных, выбор данных, интеллектуальный анализ данных, оценка шаблонов.

УДК 004.94 / Лєпа Є.В. / Моделювання корпоративної комп'ютерної мережі // Проблеми інформаційних технологій. – 2018. – №1 (023). – С.248-252: Бібл.: 10 назв., рус.

В статье рассматривается имитационная модель локальной компьютерной сети в системе моделирования Riverbed Modeler. Исследованная пропускная способность сети, выполненной по стандартам 10Baset и 100Baset. Определена ширина канала Internet для оптимального соотношения задержка/стоимость.

УДК 004.652.4 : 004.4'22 / Колесник Л.В., Кириченко Н.А., Костоглот И.В. / Разработка способа проектирования высоконагруженных реляционных систем хранения информации: оптимизация структуры и запросов SQL // Проблемы информационных технологий. – 2018. – № 1 (023). – С.253-260: Библ.: 12 назв., укр.

Данная статья посвящена теме проектирования и перепроектирования высоконагруженных реляционных систем хранения данных. В статье будут рассмотрены основные методы и средства для ускорения работы одной из самых популярных СУБД - СУБД MySQL. Будут рассмотрены методы оптимизации структуры базы данных и SQL запросов. На основе рассмотренных средств будет разработано и протестировано программное обеспечение для оптимизации скорости работы базы данных. Программное обеспечение позволит проводить процесс оптимизации как в полуавтоматическом, так и в автоматическом режимах. На основе проведенных тестов будет дана оценка автоматическому процессу оптимизации базы данных. **Ключевые слова:** СУБД, MySQL, база данных, SQL, структура, оптимизация, автоматизация,

U.D.C. 004.82 004.622 519.766 / Katerynych L.O., Petelko Y.Y. / Algorithm of search of connections and dependences in the website data // Problems of Information Technologies. – 2018. – № 1 (023). – С.241-247: Bible: 10 titles, ua.

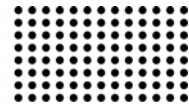
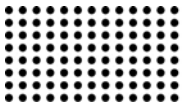
A significant increase in the number of users of global networks, including the Internet, have caused increase in the amount of information that is freely accessible. Big amount of such information can be practically useful, however the extraction of useful information and find the relationship between the information contained in a large number of "raw" data is a complex task. The research is aimed at analyzing methods for extracting significant data, algorithm for searching dependencies in web page data and research for software development tools which insure feasibility of algorithm and possibility of practical use. The main idea of initial steps is to describe methods of data extraction. Initials step of the work describes methods how data can be extracted from freely accessible storages. **Keywords:** data cleaning, data integration, data selection, data mining, pattern evaluation.

UDK 004.94 / Lєpa E.V. / Modeling of the corporate computer network // Problems of information technologies. – 2018. – No. 1 (023). – С.248-252: The bible: 10 names, rus

In this article, a simulation model of a local computer network in the Riverbed Modeler modeling system is considered. The investigated network capacity, performed by the standards 10Baset and 100Baset. The width of the Internet channel is determined for the optimal delay / cost ratio.

УДК 004.652.4 : 004.4'22 / L. Kolesnyk, N. Kyrychenko, I. Kostohlot / Web tool development for high-loaded data storage systems design: optimization of data base structure and SQL queries // Problems of Information Technologies. – 2018. – № 1 (023). – С.253-260 Bible: 12 titles, ua.

This article is devoted to the subject of designing and redesigning high-loaded relational data storage systems. The main methods and tools for speeding up one of the most popular DBMS MySQL database will be discussed in the article. Methods for optimizing database structure and SQL queries will be considered. The software based on these tools, will be developed and tested to optimize the speed of the database. The software will enable two modes for the optimization process - semiautomatic and automatic mode. The basis of the performed tests will give an assessment of the automatic process of database optimization. **Keywords:** DBMS, MySQL, database, SQL, structure, optimization, automation, relational systems, data storage, acceleration



реляційні системи, зберігання даних, пришвидшення.

реляционные системы, хранение данных, ускорение

УДК 629.544:531.7 / Накул Ю.О. / Розробка інформаційної моделі комп'ютерної системи контролю завантаження контейнеровоза // Проблеми інформаційних технологій. – 2018. – № 1 (023). – С.254-266: Бібл.: 13 назв., укр.

УДК 629.544:531.7 / Накул Ю.А. / Разработка информационной модели компьютерной системы контроля загрузки контейнеровоза // Проблеми информационных технологий. – 2018. – № 1 (023). – С.254-266: Библ.: 13 назв., укр.

U.D.C. 629.544:531.7 / Nakul Yu. O., Development of the information model of a computer system for controlling the loading of a container ship // Problems of Information Technologies. – 2018. – № 1 (023). – С.254-266: Bible: 13 titles, ua.

Завантаження контейнерних суден – складний і відповідальний процес, що вимагає безпосереднього контролю капітана. Існуючі системи контролю не дають можливості автоматичної реєстрації переміщень контейнерів, контролю їх маси та журналювання всіх дій кранового, що часто приводить до загублення контейнерів, нерівномірному розподілу їх маси, затримок в порту та змушує задіяти всю команду судна для нагляду за процесом. Відомі інструментальні засоби не дають можливості вести моніторинг параметрів завантаження в реальному часі, а лише дають можливість зробити попередній теоретичний розрахунок. Розроблено інформаційну модель гетерогенної комп'ютерної системи автоматичного контролю за завантаженням судна, що дозволяє здійснювати ефективно планування завантаження/розвантаження, контролювати його перебіг та здійснювати моніторинг параметрів контейнерів в реальному часі. Метод дає можливість створювати гнучкі системи автоматизації та може слугувати базою для подальших інструментів оптимізації процесів завантаження. Використання методу дозволяє відобразити результати всіх процесів в реальному часі та реалізувати інші методи визначення параметрів судна, таких як остійність та інше. *Ключові слова:* судно, інформаційна система, контроль, завантаження, UML-діаграма.

Загрузка контейнерных судов – сложный и ответственный процесс, требующий непосредственного контроля капитана. Существующие системы контроля не дают возможности автоматической регистрации перемещений контейнеров, контроля их массы и всех действий кранового, что часто приводит к утери контейнеров, неравномерному распределению их массы, задержек в порту и заставляет задействовать всю команду судна для наблюдения за процессом. Известны инструментальные средства не дают возможности вести мониторинг параметров загрузки в реальном времени, а лишь дают возможность сделать предварительный теоретический расчет. Разработана информационная модель гетерогенной компьютерной системы автоматического контроля за загрузкой судна, позволяет осуществлять эффективное планирование загрузки / разгрузки, контролировать его ход и осуществлять мониторинг параметров контейнеров в реальном времени. Метод дает возможность создавать гибкие системы автоматизации и может служить базой для дальнейших инструментов оптимизации процессов загрузки. Использование метода позволяет отображать результаты всех процессов в реальном времени и реализовать другие методы определения параметров судна, как остойчивость и прочее. *Ключевые слова:* судно, информационная система, контроль, загрузки, UML-диаграмма.

The loading of container vessels is a complex and responsible process requiring direct control by the captain. Existing control systems do not allow the automatic registration of container movements, control of their mass and logging of all crane operations, which often leads to the loss of containers, uneven distribution of their mass, delays in the port and forces the entire command of the vessel to control the process. Known tools do not give the ability to monitor the boat parameters in real time, but only give an opportunity to make a preliminary theoretical calculation. The information model of the heterogeneous computer system of automatic control of boat loading is developed, which allows to effectively plan loading / unloading, to control its flow and to monitor the parameters of containers in real time. The method makes it possible to create flexible automation systems and can serve as a basis for further tools for optimizing download processes. Using the method allows you to display the results of all processes in real time and implement other methods of determining the parameters of the vessel, such as stability, and so on. *Keywords:* ship, information system, control, loading, UML diagram.

Babkin O. V.	165	Катеринич Л. О.	240
Dos E.V.	165	Кириченко Н.А.	252
Gerasin O. S.	86	Кирюшатова Е. В.	97
Kondratenko G. V.	86	Кирюшатова Т. Г.	97
Kozlov O. V.	86	Ковалевська А. В.	204
Kropachev A. V.	165	Козуля Т. В.	139
Mingxin H.	86	Колесник Л.В.	252
Prykhodko N.	117	Кондратець В. О.	110
Prykhodko S.	117	Костоглот І.В.	252
Smykodub T.	117	Кулаковська І. В.	185
Spinov A.	117	Ларін В. О.	147
Varlamov A. A.	165	Левицька А. М.	193
Zuev D. O.	165	Лепа Е. В.	247
Бантиш О. В.	147	Лещинська І. О.	20
Бідюк П. І.	53	Лещинський В. О.	20
Боскін О. О.	80, 193	Лобода О. М.	103
Братасюк Д. І.	226	Мазманян С. Р.	193
Бродський Ю. Б.	35	Мацуй А. М.	110
Ведь Е. В.	139	Накул Ю. О.	260
Везумський О.К.	153	Петелько Ю. Ю.	240
Верес М. М.	147	Пудло І. В.	53
Веселовская Г. В.	5	Рибачок Н. А.	159
Веселовська Г. В.	12	Рябенський В.М.	125
Вовк М. А.	218	Салтан Б. А.	185
Галкін О. В.	147	Скарга-Бандурова І. С.	60
Горбач Т. В.	204	Славгородський В. Ю.	204
Гремечевський Р. В.	226	Собко Д. А.	185
Грищук Р. В.	35	Соколов А. Е.	5
Давиденко Є. О.	172	Соколов А. Є.	12
Дворецька С. В.	172	Таргунакова Ю. Д.	125
Дворецький М. Л.	172	Ткаченко В. В.	218
Демківська Т. І.	53	Ульяновська Ю. В.	133
Демківський Є. О.	53	Ходаков В. Е.	233
Деркач М. В.	60	Ходаков Виктор Е.	5
Димов В. С.	80, 103	Чалий С. Ф.	20
Дичка А. І.	159	Чередніченко О. Ю.	218
Дідук В. А.	66	Чудайкін І.І.	125
Доровская И.А.	72	Чупринка В. І.	42
Доровской Д.В.	72	Чупринка Н. В.	42
Дорошенко А. Ю.	26	Шаронова Н. В.	26
Єна Б. О.	26	Шевчук М. М.	159
Єршова С. І.	218	Шубін І. Ю.	204
Журавська І. М.	180	Юсин Я. О.	159
Заболотня Т. М.	159	Яковенко В. О.	133
Захарченко Л. Н.	97	Янголенко О. В.	26
Захарченко Р. Н.	97	Яцюк С. В.	233
Зелінський Г. Ю.	42	Яшина О. М.	226



К СВЕДЕНИЮ АВТОРОВ ПУБЛИКАЦИЙ В СБОРНИКЕ «ПРОБЛЕМЫ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ»

«ПИТ» является периодическим научным журналом, рекомендованным ВАК Украины для опубликования научных работ с целью освещения полученных **новых научных и практических результатов, не публиковавшихся нигде ранее в открытой печати.**

Журнал принимает к рассмотрению статьи в следующие рубрики:

- Математическое моделирование;
- Информационные технологии;
- Информационная безопасность, компьютерные сети и телекоммуникации;
- Общие вопросы информатики;
- Краткие сообщения.

Объем статей до 10 стр., включая пункты (см. ниже): сведения об авторах и научные интересы. Статьи принимаются на **украинском, английском и русском** языках и публикуются на языке оригинала.

Публикация в сборнике платная. Стоимость 1 полной/неполной страницы формата **A4 40 грн**

Сборник входит в **научометрические базы** Research Bible, Open Academic Journals Index (OAJI), Directory of Open Access Journals (DOAJ), AcademicKeys, Index Copernicus, National Library of Ukraine (Vernadsky), РИНЦ (Российский индекс научного цитирования).

ТРЕБОВАНИЯ К СОДЕРЖАНИЮ И ОФОРМЛЕНИЮ НАУЧНЫХ СТАТЕЙ

В целях облегчения представления научных и практических результатов Редакционная коллегия рекомендует авторам придерживаться следующей структуры **научной** работы, содержащей основные разделы, которые могут иметь отличные от приведенных ниже наименования:

Введение (Слово «Введение» необязательно). Введение должно содержать:

- краткое освещение достигнутых результатов на сегодняшний день по рассматриваемой проблеме; актуальность проблемы; краткое описание задачи, определяющее её **актуальность и место** в общей научной или технической проблеме;
- освещение цели получения автором(ами) в статье принципиально **новых научных** и (или) **практических** результатов. Ориентировочный объем: ≤ 10% от общего объема статьи.

Постановка задачи или проблемы. Включает:

- формулирование цели решения задачи;
- описание задачи, физической и (или) феноменологической модели объекта (процесса) исследования;
- общую постановку решения задачи, выбор метода решения и его обоснование.

Решение задачи (анализ проблемы – основной раздел статьи). Раздел должен содержать:

- определение области определения задачи, ограничений, граничных и (или) начальных условий задачи; оценку точности (погрешности) полученного решения;
- постановку задачи, включая разработку её математической модели;
- описание реализации метода решения;
- общее представление полученного решения.

Основные результаты и выводы. Приводится представление (в том или ином виде) **основных практических результатов.** Излагаются основные выводы по работе, дается представление и **оценка их научной новизны.**

При этом автор должен по содержанию статьи руководствоваться требованиями ВАК Украины (приведены в конце).

В **списке литературы** обязательно должны присутствовать как можно больше первоисточников по исследуемому вопросу (не более чем 3-4-летней давности). Не следует ограничиваться цитированием работ, которые принадлежат только одному коллективу авторов или исследовательской группе. Необходимой является ссылка на современные заграничные публикации. Статьи, которые не содержат ссылок на работы, опубликованные на протяжении последнего десятилетия, автоматически считаются не отвечающими редакционным требованиям.

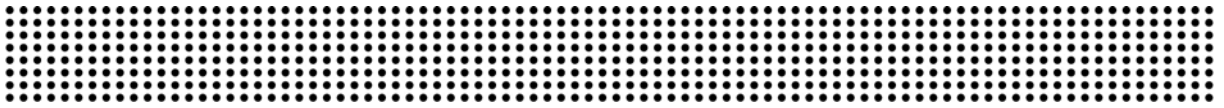
При этом приведенные конкретные наименования разделов статьи являются необязательными.

Стиль основной части должен быть: логически последовательным, доказательным.

ОБЩИЙ ВИД НАУЧНОЙ СТАТЬИ:

УДК находится в верхнем левом углу.

Строкой ниже находится список авторов, выровненный по правому краю: **И.О. Фамилия, И.О. Фамилия**



НАЗВАНИЕ НАУЧНОЙ СТАТЬИ ПО ЦЕНТРУ

Введение. Основной текст статьи набирается шрифтом Times New Roman, размер 10. ...

..... и т. д.

ЛИТЕРАТУРА: (обязательно латиницей!)

1. Smirnov I.N. Zdorov'e cheloveka kak filosofskaya problema //Voprosi filosofii. – 1985. – №7. – С.24-26.
2. Gladun V.P. Planirovanie reshenij. – К.: Naukova dumka, 1987. – 56 с.

Фамилия Имя Отчество, ученая степень, ученое звание, должность, место работы, телефон и E-mail для оперативной связи.

Научные интересы:

Структура **кратких сообщений** (к ним Редакция относит статьи объемом **до двух** страниц, включая литературу) – произвольная, но с обязательным последним разделом (см. выше): «Основные результаты и выводы».

Редакционная коллегия оставляет за собой право как не публиковать работы в случае их отклонения, так и принимать решение об их соответствии разделам журнала.

Для публикации автор должен предоставить в редакцию:

- электронный вариант статьи – файл, набранный в редакторе Microsoft Word for Windows, оформленный вместе с рисунками, таблицами и т.д., если таковые имеются;
- файл каждого рисунка – должен быть включен в статью, а также в виде отдельного файла.

Кроме основного текста, статья должна содержать:

- краткую аннотацию на украинском, английском и русском языках, включая УДК, фамилии авторов, название статьи, страницы, количество библиографических источников, язык и в конце ключевые слова, если они есть, – вся информация на трех языках (на отдельной странице);
- *Аннотация, соответствующая языку оригинала, определяет формат информационного файла для УРЖ «Джерело».* **ОБРАЗЕЦ:**
УДК 004.04 /Бараненко Р.В. /Аналіз моделей даних, що використовуються в ГІС //Проблеми інформаційних технологій. –2013. – №1 (013) – С.150-158. Бібл.: 6 назв., рус.

Рассматриваются особенности моделей данных, используемых в геоинформационных системах. Особое внимание уделено использованию послонной и объектно-ориентированной пространственных моделей данных.

Ключевые слова (если необходимы): геоинформационная система, модель, данные.

- рецензию стороннего (не входящего в коллектив авторов) доктора наук по специальности, по которой подается статья;
- сведения об авторах (ФИО полностью, ученая степень, должность, название кафедры и вуза);
- научные интересы авторов;
- в отдельном файле полные сведения об авторах, т.е. дополнительно – домашний и рабочий адреса и телефоны, e-mail (если есть) и др.

Требования к оформлению рукописи:

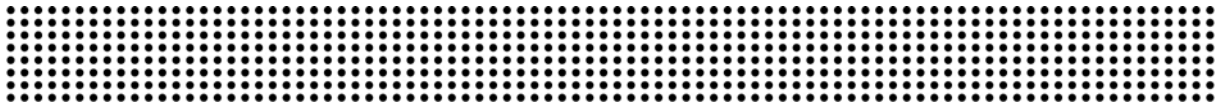
1. Машинописные рукописи не принимаются.
2. Рисунки, содержащие кривые линии, следует выполнить на отдельных листах тушью или другим способом (напр., с помощью графического редактора) – четко, черным цветом.

На обороте каждого рисунка следует указать номер рисунка, название статьи и фамилии авторов. Предполагается ограничение как на количество (≤ 5), так и на размеры рисунков.

3. Формулы должны быть напечатаны с учетом таких требований: начертание формул должно быть четким, принадлежность индексов ясна, дроби и скобки оформлены четко и ясно;
4. Список используемых источников помещается в конце статьи, нумеруется в соответствии с порядком появления текста.

Требования к оформлению файла:

1. Параметры страницы (опция меню **ФАЙЛ, Параметры страницы**):
 - размер бумаги – А4: 210×297 мм;
 - ориентация листа – книжная (альбомная не допускается);
 - поля: левое – 2,5 см, остальные – 2 см;
 - переплет – 0 см;
 - колонтитулы – 1 см;
 - страницы без нумерации.



Весь текст должен быть набран шрифтом Times New Roman, размер 10 пт, межстрочный интервал – одинарный.

Для выделения **отдельных слов** допускается:

- **полуужирное** начертание символов, а также использование *курсива* и подчеркивания;
- изменение интервала между символами;
- использование ^{верхнего} или _{нижнего} индексов.

Цветовое оформление не допускается.

2. Абзац должен иметь следующий формат (опция меню: **ФОРМАТ, Абзац**):

- отступ слева и справа – 0 см;
- первая строка – отступ на 1,0 см;
- интервалы перед и после абзаца – 0 см;
- интервал между строками – одинарный;
- выравнивание – по ширине.

3. Рисунки должны быть черно-белыми и вставлены в файл и распечатку статьи. Формат рисунков (tif, psx, bmp и др.) должен быть совместим с редактором текста **Microsoft Word 2010**. Разрешение рисунков – не менее 300 dpi. Рисунки могут быть вставлены в текст при помощи следующих команд:

- **Добавить рисунок – Рисунок Microsoft Word;**
- **Добавить рисунок – Bitmap Image;**
- **Добавить рисунок – Рисунок Corel Draw.**

4. Формулы должны быть набраны при помощи встроенного редактора формул Equation Editor 2.0, 3.0. Все формулы вставляются в таблицу с очерченным контуром, состоящую из двух колонок: в первой находится формула без абзаца и выровненная **по центру**, во второй – номер формулы (если такой имеется) тоже без абзаца и с выравниванием **по правому краю**. Граница между колонками таблицы устанавливается на отметке 14 см. В тексте формулы должны быть выделены сверху и снизу пустой строкой. Ниже приведен пример вставки формулы:

$\lim_{\tau \rightarrow 0} \int_0^{\tau} P d\tau = \pi \neq 0 .$	(1)
--	-----

Параметры в редакторе формул должны в точности соответствовать приведенным ниже.

Размеры (опция меню редактора Equation Editor: **РАЗМЕР, Определить . . .**):

- Обычный 10 пт.
- Крупный индекс 7 пт.
- Мелкий индекс 5 пт.
- Крупный символ 16 пт.
- Мелкий символ 10 пт.

Стили (опция меню **СТИЛЬ, Определить . . .**)

- Текст Times New Roman
- Функция Times New Roman
- Переменная Times New Roman
- Стр. греческие Symbol
- Пр. греческие Symbol
- Символ Symbol
- Матрица-вектор Times New Roman полуужирный
- Числа Times New Roman

5. Таблицы создаются только при помощи Microsoft Word. Предполагается ограничение как на количество (≤ 5), так и на размеры таблиц (см. п. 8).

6. Суммарный объем рисунков и таблиц должен быть менее 50% объема основного раздела.

7. На основании данных «**Для публикации . . .**» формируется информационный файл о статье для реферативного журнала ДЖЕРЕЛО.

Просьба к авторам строго придерживаться требования Редколлегии. В противном случае Ваша статья не будет соответствовать требованиям информативного обеспечения национального реферативного журнала «Джерело», что автоматически влечет за собой отклонение статьи от публикации в текущем номере журнала.

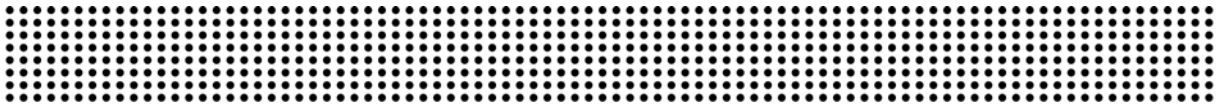
1. В случае необходимости уточнения по оформлению статей – связь с ответственным редактором сборника

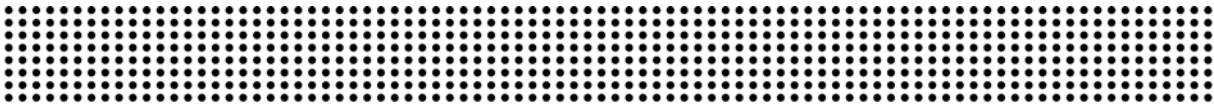
E-mail: Magazinepit@ukr.net

2. Рукописи авторам не возвращаются.

Рукописи для рассмотрения Редколлекцией сборника принимаются на кафедре Информационных технологий Херсонского национального технического университета по адресу:

73008, Украина, г. Херсон, Бериславское шоссе, 24, ХНТУ, корп. 3, ауд. 318.





„ПРОБЛЕМИ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ” №23 /2018/

Свідоцтво про реєстрацію KB №11321-20 IP

ВИДАВЕЦЬ:

Херсонський національний технічний університет

Адреса редакції: 73008 м. Херсон,
вул. Бериславське шосе, 24, корп.3, а. 318

Телефони: 8(0552) 51-57-31, 32-69-66

e-mail: Magazinepit@ukr.net

Підписано до друку 20.04. 2018.

Формат видання 60x84/8. Папір офсетний.

Ум. друк. арк. 18,14. Тираж 300 прим.

**Друк ПП «ОЛДІ-Плюс»: 73034 м. Херсон, Паровозна, 46-а,
e-mail: office@oldiplus.com, Свід. сер. ХС № 2 від 16.08.2000 р.**