

УДК 004.986

[https://doi.org/ 10.35546/kntu2078-4481.2019.3.17](https://doi.org/10.35546/kntu2078-4481.2019.3.17)

В.Г. ШЕРСТЮК

Херсонський національний технічний університет
ORCID: 0000-0002-9096-2582

І.В. СОКОЛ

Херсонський національний технічний університет
ORCID: 0000-0002-7324-1441

Р.Н. ЛЕВКІВСЬКИЙ

Херсонська державна морська академія
ORCID: 0000-0003-3114-179X

ДИНАМИЧЕСКИЕ ПРЕЦЕДЕНТНЫЕ СИСТЕМЫ: КЛАССИФИКАЦИЯ И ТРЕБОВАНИЯ ДЛЯ РЕШЕНИЯ ЗАДАЧ УПРАВЛЕНИЯ

В данной работе рассмотрены динамические прецедентные системы, относящиеся к классу интеллектуальных систем поддержки принятия решений, основанных на ситуационном подходе. Исследованы динамические предметные области, проведен анализ их особенностей. Предложен теоретический подход для адекватного описания динамики предметной области на уровне прецедентов и проблемных ситуаций. Выявлена необходимость учета динамики описания ситуации и прецедентов, неполноты, неточности и неопределенности описания прецедентов, а также требование ослабления вовлеченности лиц, принимающих решения, в процесс поиска и принятия решений. Показано, что прецедент необходимо рассматривать как развивающийся во времени процесс, причем анализировать необходимо полное множество одновременно протекающих в динамической системе процессов, а наблюдения могут поступать из различных источников, возможно, с некоторым запаздыванием относительно момента свершения реального события. Представлена классификация динамических прецедентных систем по формату исходной информации и по формату решения. Сформулированы требования к динамическим прецедентным системам при решении задач управления в динамических предметных областях. Предложены схемы реализации динамических прецедентных систем, показано, что для решения задач управления в сложных динамических системах целесообразно использовать гибридные динамические прецедентные системы, реализованные по схеме «ситуация + поток событий»-«план»-«сценарий»-«управляющее воздействие» либо «ситуация + граф»-«план»-«сценарий»-«действие», и обеспечивающие механизмы адаптации и верификации на уровне сценариев. Результаты работы позволяют использовать прецедентный подход при решении трудноформализуемых задач в динамических предметных областях.

Ключевые слова: динамическая предметная область, сложная динамическая система, динамический объект, параметр, прецедентная система, схема, решение, проблемная ситуация.

В.Г. ШЕРСТЮК

Херсонський національний технічний університет
ORCID: 0000-0002-9096-2582

І.В. СОКОЛ

Херсонський національний технічний університет
ORCID: 0000-0002-7324-1441

Р.М. ЛЕВКІВСЬКИЙ

Херсонська державна морська академія
ORCID: 0000-0003-3114-179X

ДИНАМІЧНІ ПРЕЦЕДЕНТНІ СИСТЕМИ: КЛАСИФІКАЦІЯ І ВИМОГИ ДЛЯ ВИРІШЕННЯ ЗАВДАНЬ УПРАВЛІННЯ

У даній роботі розглянуто динамічні прецедентні системи, що відносяться до класу інтелектуальних систем підтримки прийняття рішень, заснованих на ситуаційному підході. Досліджено динамічні предметні області, проведено аналіз їх особливостей. Запропоновано теоретичний підхід для адекватного опису динаміки предметної області на рівні прецедентів і проблемних ситуацій. Виявлено необхідність врахування динаміки опису ситуації і прецедентів, неповноти, неточності і невизначеності опису прецедентів, а також вимога ослаблення залученості осіб, які приймають рішення, до процесу пошуку і прийняття рішень. Показано, що прецедент необхідно розглядати як процес, що розвивається в часі, причому аналізувати необхідно повну множину процесів, які одночасно виникають в динамічній системі, а спостереження можуть надходити з різних джерел, можливо, з певним запізненням щодо моменту звершення реальної події. Представлено класифікацію

динамічних прецедентних систем за форматом вихідної інформації і за форматом рішення. Сформульовано вимоги до динамічних прецедентних систем при вирішенні завдань управління в динамічних предметних областях. Запропоновано схеми реалізації динамічних прецедентних систем, показано, що для вирішення задач управління в складних динамічних системах доцільно використовувати гібридні динамічні прецедентні системи, реалізовані за схемою «ситуація + потік подій» - «план» - «сценарій» - «керуючий вплив» або «ситуація + граф»-«план»-«сценарій»-«дія», такі, що забезпечують механізм адаптації та верифікації на рівні сценаріїв. Результати роботи дозволяють використовувати прецедентний підхід при вирішенні важкоформалізованих задач в динамічних предметних областях.

Ключові слова: динамічна предметна область, складна динамічна система, динамічний об'єкт, параметр, прецедентна система, схема, рішення, проблемна ситуація.

V.G. SHERSTJUK

Kherson National Technical University
ORCID: 0000-0002-9096-2582

I.V. SOKOL

Kherson National Technical University
ORCID: 0000-0002-7324-1441

R.N. LEVKIVSKYI

Kherson State Maritime Academy
ORCID: 0000-0003-3114-179X

DYNAMIC CASE-BASED SYSTEMS: CLASSIFICATION AND REQUIREMENTS FOR SOLVING CONTROL PROBLEMS

This work presents dynamic case-based systems considered as a class of intelligent decision support systems based on a situational approach. Dynamic domains are investigated, their features are analyzed. A theoretical approach is proposed for an adequate description of the domain dynamics at the case and problem situation levels. The necessity of considering dynamics of problem situations and case descriptions, the incompleteness, inaccuracy and uncertainty of case descriptions, as well as the requirement of weakening the involvement of decision-makers in the case search and decision-making process, is revealed. It is shown that the case must be considered as a process developing in time, and it is necessary to analyze the full set of processes simultaneously occurring in the dynamic system, so that observations can come from various sources and possibly with some delay with respect to the moment of a real event occurrence. The classification of dynamic case-based systems according to the format of the initial information and the format of the solution is presented. The requirements to dynamic case-based systems are formulated for solving various control problems in dynamic domains. The schemes for implementing dynamic case-based systems are proposed; it is shown that for solving control problems in complex dynamic systems it is advisable to use hybrid dynamic case-based systems implemented according to the schemes "situation + flow of events" → "plan" → "scenario" → "control actions" or "situation + graph" → "plan" → "scenario" → "action" providing a mechanism for adaptation and verification at the scenario level. The results of the work make it possible to use a case-based approach to solve weak problems in dynamic domains.

Keywords: dynamic domain, complex dynamic system, dynamic object, parameter, case-based system, scheme, solution, problem situation.

Постановка проблеми

В настоящее время прецедентные системы (ПС), представляющие особый класс интеллектуальных систем поддержки принятия решений (ИСППР), основанных на ситуационном подходе, достаточно изучены и активно развиваются. На сегодняшний день разработаны десятки моделей и механизмов вывода по прецедентам [1], а наиболее удачные ПС используются для решения широкого круга практических задач в условиях неполноты и неопределенности исходных данных.

В то же время, как известно, в подавляющем большинстве ПС являются по сути ИСППР статического типа, основанными на детально и прозрачно описанных прецедентах с четко обозначенными границами, заданными множеством достоверных, но статичных параметров проблемной ситуации. Таким образом, если в ПО преобладают статичные объекты, а процессы относятся к категории достаточно медленно изменяющихся, прецедентный подход к решению большинства практических задач управления, планирования, диагностики и т.д. является вполне адекватным. Однако, как только проистекающие процессы начинают демонстрировать некоторую динамику, исходные данные изменяются в более быстром темпе, чем позволяет время, необходимое для поиска решений в ПС, что приводит к запаздыванию получаемых решений относительно проблемной ситуации [2].

Кроме того, подавляющее большинство ПС предназначены для решения различных задач в интерактивном режиме, при непосредственном участии лиц, принимающих решения (ЛПР), которые

«помогают» ПС выполнять отбор уместных прецедентов, производить адаптацию и верификацию решений, оценивать полезность найденных решений и необходимость их сохранения на будущее. Естественно, включение человека в контур поиска и принятия решений вносит в работу ПС дополнительные задержки и зависимость от психофизиологического состояния ЛПР.

Таким образом, при необходимости решения трудноформализуемых задач в динамических ПО трудности, связанные применением ПС, становятся непреодолимыми из-за особенностей классического прецедентного подхода. Известные попытки адаптировать ПС к условиям неполноты, неточности и неопределенности исходной информации и высокой динамики ситуаций в открытых слабоструктурированных ПО были противоречивыми – иногда действительно удавалось построить успешно работающие динамические ПС, но участие ЛПР в контуре поиска и принятия решений существенно ограничивало их практическое применение. Это, в свою очередь, сужало сферы применения ПС для сравнительно несложных задач в весьма узких ПО. В то же время, прецедентный подход хорош в тех случаях, когда задачи не могут быть решены с использованием других подходов, например в случае отсутствия адекватных и полных моделей, невозможности построить исчерпывающее множество правил и т.д. Но проблема состоит в том, что чем проще задача и уже ПО, тем больше возможности создать необходимые модели и требуемые правила, что нивелирует практическую полезность ПС.

Таким образом, существует очевидная проблема использования прецедентного подхода при решении трудноформализуемых задач в динамических ПО. Решение этой проблемы возможно с помощью разработки динамических прецедентных систем (ДПС), допускающих, во-первых, динамику описания ситуации и прецедентов, во-вторых, неполноту, неточность и неопределенность описания прецедентов, и, в-третьих, ослабление вовлеченности ЛПР в процесс поиска и принятия решений [3]. Для этого необходимо разработать новые модели и механизмы поиска и адаптации решений в ПС.

Статья посвящена вопросам исследования динамических ПО, классификации динамических ПС и выработки требований к их разработке.

Анализ последних исследований и публикаций

В классических ПС независимые друг от друга проблемные ситуации никогда не пересекаются во времени – в каждый момент времени ПС рассматривает только один прецедент, а сам прецедент не допускает возможности развития проблемной ситуации, являясь всего лишь статическим «снимком» значений определенных параметров ситуации в некоторый заданный момент времени [4]. Если описание проблемной ситуации изменяется быстрее, чем ПС успевает сформировать решение, то полученное решение может являться запоздалым. Во многих динамичных ПО запоздалое решение – это хуже, чем вообще отсутствие какого-либо решения [5].

С целью преодоления указанного недостатка в [6] была предложена концепция «непрерывных» ПС (Continuous CBR) для работы с ситуациями, развивающимися во времени. Их особенностью является циклическая, безостановочная организация процесса принятия решений, учитывающая динамику изменений внешней среды. Основанные на данном подходе ПС ACBARRS и SINS применяются для управления перемещением роботов. Их особенностью состоит в том, что входная и выходная информация представлена векторами аналоговых величин, а динамика ситуации отражается упрощенно посредством изменения значений параметров; сами же прецеденты имеют слишком упрощенное представление и содержат прямые зависимости между значениями входных сигналов и соответствующими значениями выходных управляющих сигналов. К положительным свойствам «непрерывных» ПС относится их способность работать почти в реальном времени.

В [7] предложена концепция «беспрерывной» ПС (Ceaseless CBR), работающей в квазиреальном времени, и предназначенной для обработки распределенного во времени непрерывного потока сигналов, формируемого внешней средой. ИС Alba и SOID, разработанные на основе данной концепции, предназначены для решения достаточно узкой задачи обнаружения вторжений в компьютерные сети, в них используются статистические методы сравнения последовательностей сигналов. Это препятствует применению данного подхода в других ПО, где невозможно наработать соответствующую статистику. В то же время, к положительным сторонам концепции следует отнести адаптивность к условиям неточной, неполной и противоречивой исходной информации.

В [8] предложена концепция «прецедентов допустимого времени» для обработки протяженных во времени ситуаций. Для моделирования динамики в них используются совокупности временных рядов, включающих неточные и зашумленные данные. Данный подход позволяет представить эволюцию состояний динамической системы с помощью последовательности статических снимков числовых параметров во времени, на которые накладываются определенные ограничения. Однако, для зависимых переменных требуется наличие числовых оценок, которые далеко не всегда могут быть получены. Использование данного подхода в ИС Cseek и MNAOMIA показало его низкую эффективность в системах с изменяющимся составом и структурой, имеющих высокую динамику и отличающихся неточностью и неполнотой исходной информации [9].

В [10] для представления динамики ситуаций в прецедентах предложено использовать ситуационное исчисление SitCalc, однако, это исчисление является неразрешимым, поэтому получение решений за конечное время не гарантировано. Более того, сравнительно низкое быстродействие интерпретатора ограничивает использование данного подхода в системах с высокой динамикой процессов.

В [11] представлена идея «проворной» ПС (Agile CBR), преобразующей поступающие сигналы от различных коинцидентных источников в самоорганизующуюся структуру, в которой и производится поиск подобных фрагментов. Использование нелинейных методов поиска совпадений ограничивает быстродействие и приводит к сильной зависимости от особенностей ПО. В развитие данной идеи в [12] предложена концепция «отслеживающей» ПС (Trace CBR), основанная на графовом представлении «следа», оставляемого пользователем в процессе взаимодействия с компьютерной системой. Однако, чрезмерно высокий уровень интерактивности данного подхода ограничивает сферу его применения и препятствует использованию в быстродействующих ДПС.

В [13] для управления мобильными роботами представлен метод динамических окон с неявным представлением динамики, в [14] – метод связанных прецедентов, выражающих состояние динамической системы таким образом, что каждый очередной прецедент определен через предшествующий прецедент и действие, изменившее состояние системы. В сложных динамических системах это создает проблему неуправляемого роста размеров хранилища прецедентов.

В [15] представлены темпоральные расширения правдоподобного вывода на прецедентах с использованием метрической временной логики, для которой существует алгоритм вывода с полиномиальной оценкой сложности. Проблема вывода решается как задача согласования временных ограничений, а поиск подобных прецедентов производится на основе метода ближайшего соседа.

Общий анализ состояния работ в области исследования ДПС показывает, что вопросы анализа динамики процессов, протекающих в сложных динамических системах (СДС), и связанные с ними задачи диагностики, планирования, управления и предсказания нежелательных ситуаций исследованы фрагментарно, а полученные модели узкоспециализированы и малоэффективны.

Одной из основных проблем является не вполне адекватный способ моделирования динамики ПО – логический, в виде вектора параметров, временных рядов, потоков сигналов, графов, самоорганизующихся карт Кохонена, уравнений динамики, связанных списков и т.д., и ограниченность способов представления решений ПС векторами выходных параметров, выводами диагностики, логическими заключениями, элементарными действиями.

Формулирование цели исследования

На основе анализа существующих подходов можно сделать вывод, что в ДПС прецедент необходимо рассматривать как развивающийся во времени процесс, причем анализировать необходимо полное множество одновременно протекающих в СДС процессов. Это означает, что прецеденты могут быть зависимыми, «растянутыми», пересекаться во времени, а наблюдения могут поступать из различных источников, возможно, с некоторым запаздыванием относительно момента свершения реального события. Существенно, что рассмотрение множества наблюдаемых в СДС процессов в виде событий должно производиться параллельно, при этом необходим учет возможного изменения наблюдаемых параметров проблемной ситуации непосредственно в ходе поиска прецедента. Необходимо также ослабление временных зависимостей между связанными событиями, например, вместо четкой информации о временных интервалах использование приближенных либо нечетких оценок, а также установление порядка появления событий на качественном уровне.

Исходя из результатов анализа публикаций, в настоящее время отсутствует адекватный с точки зрения построения ДПС подход к отражению динамики ПО, и для построения ДПС необходимо разработать и обосновать соответствующий теоретический подход. Целью данной статьи является анализ особенностей динамических ПО и классификация ДПС по способу моделирования динамики ПО и способу представления решений, что позволит сформулировать требования к разработке ДПС и определить эффективные механизмы их функционирования.

Изложение основного материала исследования

Основой структуры всякой ПО являются объекты. Объекты имеют свойства и находятся в определенных отношениях, – это справедливо как для статичных, так и для динамичных ПО. Различие между ними, однако, проявляется в том, что структура статичной ПО со временем либо не изменяется, либо изменяется очень медленно, а в динамичной ПО и свойства объектов, и отношения между ними, и состав и структура самой ПО могут быть подвержены быстротечным изменениям. Поскольку объекты динамичной ПО эволюционируют во времени, их называют динамическими объектами (ДО).

В динамичных ПО может одновременно существовать множество СДС. В данной статье далее будем рассматривать СДС не в традиционном математическом, а в семантическом смысле, как некоторую абстракцию. ДПО будем определять с использованием двух существенных признаков – состава ее элементов и их взаимоотношений (рис. 1).

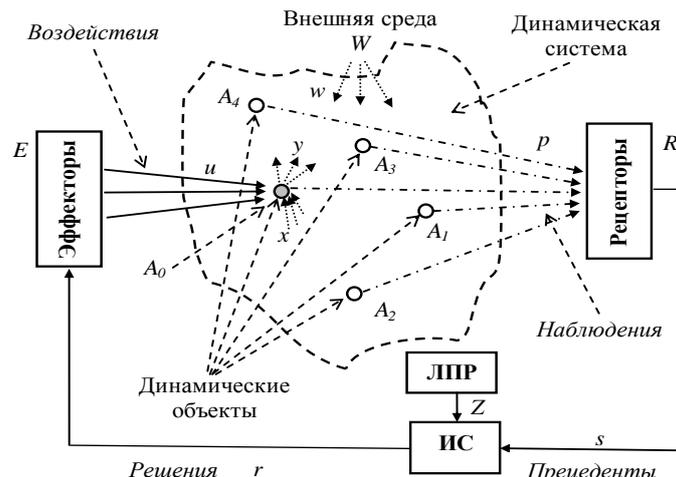


Рис. 1. Описание динамичной предметной области

В динамичной ПО все объекты или их часть являются динамическими. ДО будем рассматривать как некую сущность, имеющую:

- а) состояние;
- б) входы (точки приложения внешних воздействий, воспринимающие эти воздействия);
- в) выходы (точки, значения величин в которых характеризуют состояние объекта).

ДО реагирует на внешние воздействия изменением своего внутреннего состояния и выходных величин, характеризующих его состояние. И воздействие на ДО, и его реакция могут изменяться с течением времени. Они могут быть наблюдаемы, т.е. могут быть измерены соответствующими устройствами. Как правило, ДО имеет некоторую внутреннюю структуру, состоящую из взаимодействующих динамических элементов.

Динамические объекты отличаются от статических проявлением некоторой активности – они изменяют с течением времени свои свойства и отношения с другими объектами, оказывая тем самым влияние как на структуру СДС, так и на свойства и отношения других объектов. Существенно, что такие изменения могут быть не только реакцией на внешние по отношению к ДО воздействия (воздействий других объектов, внешней среды и т.п.), но и следствием самостоятельного выполнения объектом определенных действий.

Внешняя среда W воздействует на СДС и входящие в нее объекты. Ее воздействия w , как правило, носят стохастический характер. Каждый из ДО A_i имеет свои входы x и выходы y , а его активность обычно представляют как:

$$\bar{Y} = f(\bar{X}, \bar{W}),$$

где \bar{Y} – вектор выходов ДО, $\bar{Y} = \overline{y_1, \dots, y_n}$;
 \bar{X} – вектор входов ДО, $\bar{X} = \overline{x_1, \dots, x_m}$;
 \bar{W} – вектор внешних воздействий, $\bar{W} = \overline{w_1, \dots, w_k}$.

СДС может быть наблюдаемой полностью или частично. Обычно наблюдение осуществляется рецепторами R – датчиками или измерителями, фиксирующими состояние входов x всех или части объектов, а также самой СДС. Таким образом, в каждый момент времени с помощью рецепторов можно оценить значения множества параметров, прямо или косвенно отражающих свойства объектов, свойства отношений объектов, а также свойства самой СДС.

Активность ДО в СДС, как правило, управляема. Это означает, что существуют также эффекторы E – объекты (программы, устройства), которые осуществляют различного рода управляющие воздействия u на ДО, изменяя его свойства или отношения с другими объектами и СДС в целом. В результате приложения управляющих воздействий ДО изменяет свою активность, соответственно изменяется и его влияние на СДС. Управление активностью ДО осуществляется как:

$$\bar{U} = \Phi(\bar{X}, \bar{Y}, \bar{W}),$$

где \bar{U} – вектор управления ДО, $\bar{U} = \overline{u_1, \dots, u_l}$.

Отображение Φ обычно реализуется некоторой системой управления, в которую входит ДПС и/или ЛПР. В присутствии ЛПР СДС становится целенаправленной, причем ее цель предполагается установленной извне. Чтобы двигаться к установленной цели, СДС должна иметь соответствующие элементы – ДПС, рецепторы и эффекторы, и выполнять целевые действия, результаты которых являются ожидаемыми и прогнозируемыми только в структурированных ПО. Что касается слабоструктурированных и неструктурированных ПО, в них выполнение действия не всегда приводит к искомому результату из-за выявленных особенностей СДС: стохастичности внешних воздействий, неполной наблюдаемости, нелинейности ДО, нестационарности процессов и т.д. [16]. Именно в таких случаях функции управления частично или полностью берет на себя ЛПР.

Компенсация внешних воздействий обычно выполняется путем формирования управляющих воздействий на эффекторы ДО, что изменяет параметры активности самого ДО. Вследствие наличия внешних воздействий и необходимости их компенсации решение задачи управления в СДС представляет собой достаточно сложную проблему, в том числе из-за высокой размерности пространства принятия решений и взаимокорреляции поведения отдельных ДО в СДС.

Если известно отображение f , описывающее функционирование ДО (а известно оно только в структурированных ПО), на его основе можно построить математическую модель и получить решение с помощью методов теории управления. Если знание f неполно и неточно, но известно отображение Φ , представляющее собой модель управления объектом, можно использовать методы теории ситуационного управления. Большинство динамичных ПО относится к классам слабоструктурированных и неструктурированных, и в них корректно построить отображение f не представляется возможным. Более того, при решении трудноформализуемых задач в слабоструктурированных и неструктурированных ПО зачастую невозможно построить и адекватное отображение Φ . В то же время, ДПС могут решать различные задачи управления в динамичной ПО, рассматривая заданное подмножество значений параметров СДС как прецедент s (см. рис. 1). В результате поиска ДПС может найти решения r , которые, поступая на вход эффектора E , окажут необходимые воздействия на ДО и на СДС в целом. Эффективность и качество полученного решения r связаны не с адекватностью моделей f или Φ , а с адекватностью моделирования динамики, присущей СДС, в механизме поиска решений ДПС.

Классификация прецедентных систем

В зависимости от того, какую исходную информацию ДПС получает на входе и в какой форме генерирует решение на выходе, можно выделить различные схемы ДПС.

Классификация по формату исходной информации

По формату получаемой исходной информации могут быть выделены простые и составные разновидности ДПС.

К простым разновидностям относятся:

- «состояние»;
- «ситуация»;
- «диагноз»;
- «событие»;
- «формула».

В первом случае (тип входа «состояние») исходное описание проблемной ситуации может представлять собой некоторое подмножество $\{\dot{x}_1, \dots, \dot{x}_i\}$ (либо полное множество $\{\dot{x}_1, \dots, \dot{x}_k\}$) параметров СДС и внешней среды (рис. 2) из наблюдаемого вектора \bar{X} .

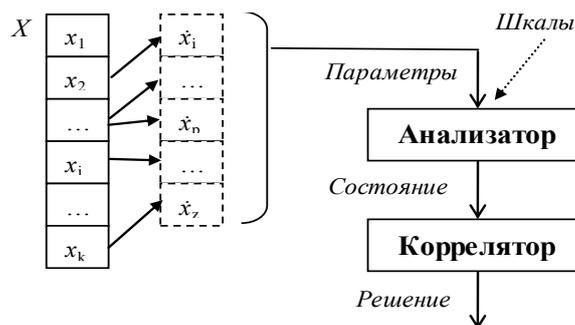


Рис. 2. Прецедентная система со входом типа «состояние»

Что касается структуры ДПС, будем исходить из традиционных определений ситуационного подхода [17], а также из того факта, что описания проблемных ситуаций могут быть неполными, неточными и даже противоречивыми.

В структуре ДПС минимально необходимы Анализатор и Коррелятор. В задачи Анализатора может входить выделение значимого подмножества наблюдаемых параметров и преобразование значений параметров с использованием заданных метрических и/или топологических шкал (например, логико-лингвистические преобразования, сопоставляющие числовым значениям параметров (20 м, 30 сек) качественные оценки («близко», «быстро»)). Таким образом, параметры могут быть представлены как количественными, так и качественными оценками значений. Задача коррелятора заключается в выборе управляющего воздействия, основываясь на описании проблемной ситуации. Коррелятор содержит конечное множество управляющих воздействий и некоторый механизм выбора и адаптации управляющего воздействия для каждого из классов проблемных ситуаций. Логика работы коррелятора зависит как от конкретной модели, лежащей в основе ДПС, так и от прикладной задачи.

Во втором случае (тип входа «ситуация») исходное описание проблемной ситуации может, кроме непосредственно подмножества значений параметров, включать результаты обобщения по одному или нескольким классифицирующим признакам $\{\ddot{x}_c, \dots, \ddot{x}_m\}$ (рис. 3). В структуру ДПС дополнительно входят Классификатор и набор эталонов. При формировании множества признаков могут использоваться как преобразования значений параметров по заданным шкалам, так и вычисления значений по известным функциональным зависимостям φ между значениями наблюдаемых параметров.

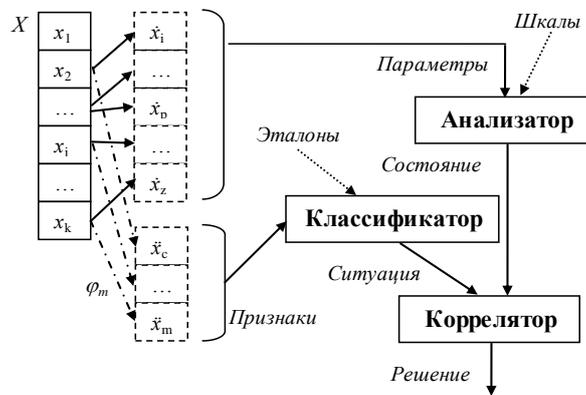


Рис. 3. Прецедентная система со входом типа «ситуация»

В третьем случае (тип входа «диагноз») в исходной информации присутствуют результаты выявленных отклонений δ от некоторых установленных (нормируемых) показателей, называемых индикаторами (рис. 4), которые формируются с помощью дополнительного элемента структуры ПС – Дифференциатора. Для выполнения диагностики вводят прямые репрезентативные показатели \hat{x}_d , характеризующие обобщенное состояние объектов или класс проблемной ситуации, сложившейся в СДС, а также косвенные показатели \hat{x}_e , связанные с параметрами состояния СДС известными функциональными зависимостями φ_e .

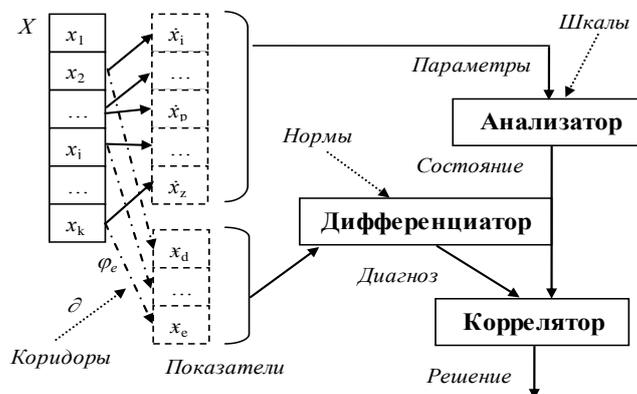


Рис. 4. Прецедентная система со входом типа «диагноз»

Следует заметить, что нормы могут быть заданы как четкие или нечеткие значения показателей, а также в виде числовых интервалов (т. наз. коридоров).

Четвертый вариант (тип входа «событие») предполагает формирование специально вводимыми с структуру ДПС Триггерами различных событий при изменении значений определенных параметров на основе вектора наблюдаемых параметров СДС (рис. 5), при этом различным фиксируемым изменениям параметров по значению или по времени могут сопоставляться события различных классов.

Связи задают подмножества параметров и пределы изменения их значений для каждого конкретного класса событий. Регистр предназначен для выполнения установок пороговых значений параметров Триггерам и выполняет функцию элемента обратной связи.

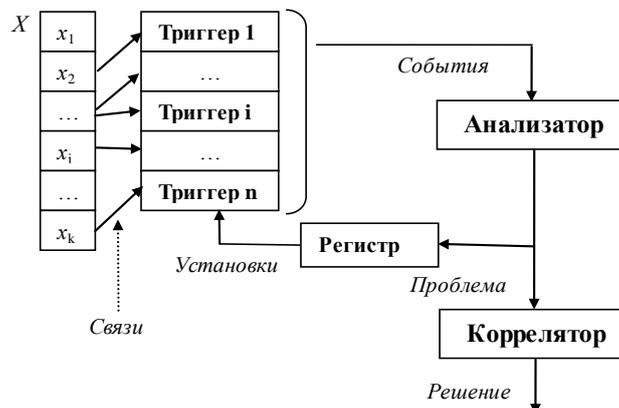


Рис. 5. Прецедентная система со входом типа «событие»

В пятом случае (тип входа «формула») описание ситуации задается множеством формул некоторого языка представления знаний – т.е. в виде логической модели (рис. 6). Этот вариант является достаточно редким. ДПС подобного типа содержит Интерпретатор (логических формул) и Коррелятор. В зависимости от используемой логической модели этот вариант является разновидностью одного из предыдущих. Например, при использовании Situation Calculus и большинства псевдофизических логик – разновидность варианта «ситуация», для Event Calculus и темпоральных моделей, логики действий и т.д. – разновидность варианта «событие», для Fluent Calculus – разновидность варианта «состояние».

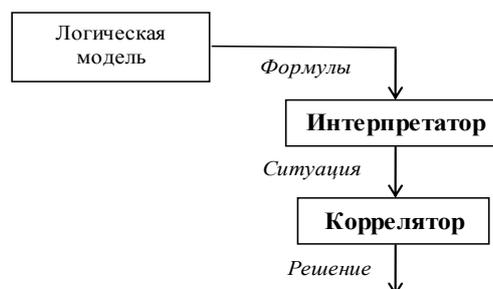


Рис. 6. Прецедентная система со входом типа «формула»

Представленные выше простые варианты форматов исходной информации не вполне «дружат» с динамичными ПО, поэтому в ДПС они используются достаточно редко. При решении практических задач в условиях динамики чаще используется некоторый составной вариант, который может быть определен на основе последовательностей, которые бывают двух разновидностей: неупорядоченные и упорядоченные. Упорядоченная последовательность отличается тем, что в ее основе лежит заданный частичный или полный порядок, причем чаще всего для этого используется временная шкала. Элементами такой последовательности могут быть «состояния» или «события», а также комбинации типа «состояние + событие». Так, при отображении последовательности «состояний», выраженных множеством числовых параметров, на временную шкалу получаем известную структуру, – временные ряды (рис. 7).

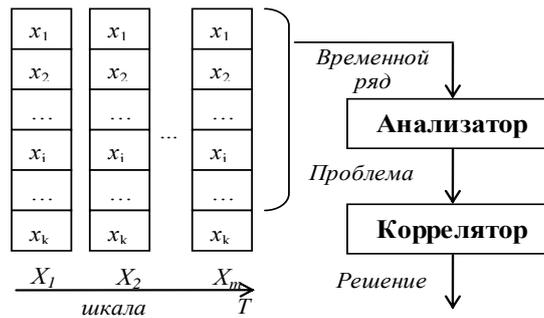


Рис. 7. Прецедентная система со входом типа «временной ряд»

Неупорядоченная последовательность событий эквивалентна графовым структурам, которые довольно часто используются для представления информации о различных процессах, в том числе о бизнес-процессах (рис. 8).

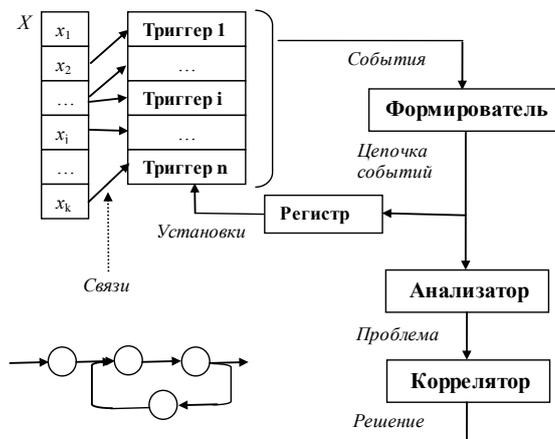


Рис. 8. Прецедентная система со входом типа «граф»

Упорядоченная последовательность событий также может быть представлена в виде потоков, называемых также диаграммами Хассе (рис. 9).

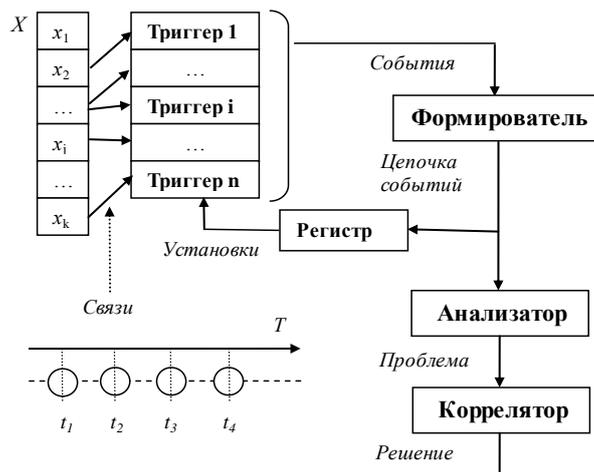


Рис. 9. Прецедентная система со входом типа «поток событий»

Использование комбинации «состояние + событие» имеет свою особенность: «событие» переводит СДС из «состояния» данного элемента в «состояние» последующего элемента последовательности (рис. 10). Известно также использование последовательностей, состоящих из элементов, представляющих собой комбинацию описания «ситуации» в виде логической модели и «обновления» (update) в виде формул, которое, будучи добавленным к описанию «ситуации», после процедуры логического вывода даст описание «ситуации» последующего элемента последовательности. Однако, в ДПС такой подход нецелесообразен ввиду чрезвычайно низкого быстродействия процедур

вывода, порождающего проблему значительных задержек поиска уместных решений.

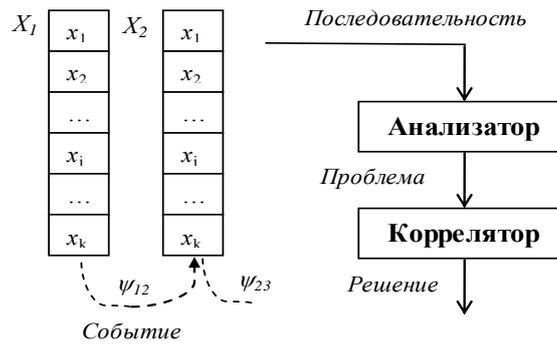


Рис. 10. Прецедентная система со входом типа «состояние + событие»

Классификация по формату решения

Выделяют следующие разновидности форматов генерируемых решений в ДПС:

- вектор выходных параметров;
- действие;
- диагноз;
- правило;
- логическое заключение;
- план;
- сценарий.

Простейшим вариантом решения является вектор выходных параметров:

$$\bar{U} = \langle u_1, u_2, \dots, u_m \rangle,$$

Где каждый из выходных параметров u_i сопоставляется с конкретным эффектором E_i и представляет собой некоторое управляющее воздействие. Выходные параметры могут быть аналоговыми или дискретными.

Другим распространенным вариантом является действие $a \in A$, где A – конечное множество возможных действий. Предполагается, что если $U = a$, выполнение a в проблемной ситуации s приведет к желаемой (целевой) ситуации s_1 : $s \xrightarrow{a} s_1$. Оба представленных варианта являются одношаговыми и активно используются в ПС, предназначенных для решения задач управления различными объектами. В ДПС оба рассмотренных варианта используются достаточно редко, хотя и позволяют достигнуть высокого быстродействия, достаточного даже для работы в реальном времени.

В ПС, предназначенных для решения задач диагностики объектов и систем, используются решения в виде диагноза $U = d$, при этом подразумевается, что $d \in C$, C – конечное множество возможных классов ситуаций. Обычно производят разбиение множества C на подмножества штатных, нештатных, опасных, критических ситуаций и т.д. В ДПС такой вид решений обычно не используется.

Решения в виде правила $U = p$ (продукционного, трансформационного и т.п.) применяются, в основном, для интерактивного представления решений ЛППР в ИСППР советующего типа. В ДПС может применяться для внесения корректных изменений в модель динамической ПО при наступлении определенных событий или достижении ситуаций заданного класса.

Логические заключения являются естественным решением для тех ПС, в которых исходная информация представлена логическими моделями, тогда для поиска уместного решения требуется выполнение логического вывода. В ДПС такой формат решений практически не используется.

План является собой более сложное, многошаговое решение, часто представляемое как частично упорядоченная совокупность действий в форме $U = \Pi = \langle a_1, a_2, \dots, a_l \rangle$, $a_1, \dots, a_l \in A$. Подразумевается, что выполнение плана Π в проблемной ситуации s влечет $s \xrightarrow{a_1} s_1 \xrightarrow{a_2} s_2 \xrightarrow{a_3} \dots \xrightarrow{a_l} s_g$, где s_g – целевая ситуация.

План может быть представлен в экстенциональной форме как графовая структура (например, И-ИЛИ-граф), сеть (TN, FRN и др.) или список, а также в интенциональной форме как совокупность формул логики действий (Action Logic). Решения в виде плана могут быть одноуровневыми и многоуровневыми, причем в последних, как правило, элементами планов более высокого уровня являются планы нижних уровней. Решения такого рода представляются в виде иерархических сетей.

Сценарий является другой разновидностью многошаговых решений, и представляет собой последовательность разнородных элементов, связанных между собой операциональными и условными отношениями. Элементами сценария чаще всего являются правила (логико-трансформационные, продукционные, метаправила и т.п.), действия и условия. Реализация сценария характеризуется упорядоченностью во времени. Сценарии часто требуют своей конкретизации, т.е. присвоения значений переменным и проверки условий, использованных при их построении. Наиболее распространенной экстенциональной формой представления сценария является сеть, вершинами которой являются элементы сценария, а дугами – отношения между ними. Такая форма представления дает возможность расширения как новыми типами элементов, так и новыми видами отношений, например, причинно-следственными. Интенциональное представление сценариев возможно с использованием различных логических исчислений. Следует заметить, что планы по своей сути представляют собой методы решений, а сценарии – процедуры решений.

Схемы реализации динамических ПС

Схема реализации ДПС определяется форматом исходной информации и форматом решения. Наиболее распространенными являются схемы «состояние»-«действие», «состояние»-«диагноз», «ситуация»-«действие», «ситуация»-«план», реже встречается схема «состояние»-«вектор параметров». Достаточно часто в ДПС используются комбинированные схемы, например «ситуация + граф»-«диагноз».

В некоторых ДПС используются более сложные схемы, например «ситуация»-«стратегия»-«действие». Применение сложных схем ставит вопрос разбиения процесса принятия решений на этапы, каждый из которых реализуется одним из уровней ДПС: например, для исходной ситуации на первом этапе на основе прецедентов может быть найдена стратегия (первый этап будет функционировать по схеме «ситуация»-«правило»), а на втором этапе для выбранной стратегии с помощью нечеткой ситуационно-событийной сети выбирается требуемое конкретное действие.

Планирующие ДПС обычно работают по схемам «состояние»-«план», «ситуация»-«сценарий», «ситуация»-«план». Выбор конкретной схемы принятия решений определяет возможность применения различных моделей и методов получения решений в ДПС.

Требования к динамическим прецедентным ИС

Проведенный анализ динамических ПО и особенностей СДС позволяют в рамках ситуационного подхода представить следующие требования, реализация которых необходима при разработке ДПС при решении различных задач управления динамическими объектами:

1. Учет неполной наблюдаемости СДС. Поскольку исходная информация неполна и неточна, а зачастую недостоверна и противоречива, от ДПС требуется обработка неполной и неточной информации от нескольких независимых источников в условиях воздействия шумов и искажений.
2. Учет целенаправленной деятельности ЛПР. Это означает, что решения чаще всего представляют собой планы, направленные на достижение поставленных перед ЛПР целей, а процесс принятия решений сводится к решению задачи адаптивного планирования. Соответственно, от ДПС будет требоваться решать задачи адаптивного планирования и обратную ей задачу распознавания планов вместо ЛПР, что способствовало бы вытеснению последнего из контура поиска и принятия решений [341].
3. Учет высокой размерности пространства принятия решений, что требует принятия мер по его существенному сужению. Наиболее эффективной мерой в данном случае является построение индексированных иерархических структур исходной информации [342].
4. Учет высокой динамики процессов в ДПО. От ДПС требуется обеспечение возможности обработки исходной информации непрерывно и параллельно, причем структура и качество исходной информации могут подвергаться изменению в процессе поиска и принятия решений. Поскольку динамика в СДС описывается событиями, основу структуры прецедентов и описания проблемной ситуации должны составлять последовательности событий.
5. Учет динамики изменения структуры СДС и целей управления, что дополнительно характеризует высокую динамику СДС. Соответственно, ДПС должна быть адаптивной к указанным изменениям и обеспечивать требуемую робастность решений.
6. Дефицит времени на обработку информации определяет высокие требования по быстродействию. ДПС должна успевать оценить ситуацию, найти уместный прецедент и предложить решение за время, которое не превышает времени оценки ситуации ЛПР, поэтому вычислительная сложность используемых в ДПС методов и алгоритмов должна быть минимальной, возможность «экспоненциального взрыва» при поиске должна предотвращаться, а в случае использования логических моделей представления знаний они должны обладать свойством разрешимости.
7. Учет критериев и ограничений при поиске решений означает, что любое допустимое решение должно проверяться (верифицироваться) на согласование с заданным множеством ограничений [343], а из множества допустимых решений должны выделяться оптимальные в смысле заданных

критериев. Может требоваться проведение анализа возможных последствий на несколько шагов вперед.

8. Учет возможного отсутствия прототипов принятия решений в случае нестандартных ситуаций. Необходимо предусмотреть при отсутствии уместных прецедентов возможность формирования решений с использованием других подходов, например на основе правил, моделей, нейронных сетей и др. В этом случае ДПС должна быть гибридной [344] и обладать развитыми функциями обучения.

Следует заметить, что целенаправленная деятельность ЛПР подразумевает логическую природу процесса управления. Поскольку использование логических моделей представления знаний исключается по требованиям быстродействия, следует использовать другие способы реализации логических процедур управления, например сценарии. При этом возможно ограничить контекст рассмотрения путем согласования сценариев управляющих воздействий с определенными параметрами, составляющими контекст ситуации (в т.ч. состояние внешней среды, оценки внешних воздействий), что возможно при решении задач адаптации и верификации решений.

Таким образом, для решения задач управления в СДС целесообразно использовать гибридные ДПС, реализованные по схеме «ситуация + поток событий»-«план»-«сценарий»-«управляющее воздействие» либо «ситуация + граф»-«план»-«сценарий»-«действие», и обеспечивающие механизм адаптации и верификации на уровне сценариев.

Выводы

1. По результатам анализа динамических ПО и особенностей СДС получен вывод, что прецедент в ДПС необходимо рассматривать как развивающийся во времени процесс, анализируя одновременно полное множество проистекающих в СДС процессов. Наблюдения могут поступать из различных источников, возможно, с некоторым запаздыванием относительно момента свершения реального события. Рассмотрение множества наблюдаемых в СДС процессов в виде событий должно производиться параллельно, при этом необходим учет возможного изменения наблюдаемых параметров проблемной ситуации непосредственно в ходе поиска прецедента. Необходимо также ослабление временных зависимостей между связанными событиями.

2. Предложен теоретический подход для адекватного описания динамики предметной области на уровне прецедентов и проблемных ситуаций с учетом неполноты, неточности и неопределенности описания прецедентов. Представленный подход направлен на ослабление вовлеченности лиц, принимающих решения, в процесс поиска и принятия решений по прецедентам.

3. Представлена классификация динамических прецедентных систем по формату исходной информации и по формату решения, позволяющая описывать механизмы поиска и принятия решений по прецедентам с помощью схем реализации.

4. Сформулированы требования к динамическим прецедентным системам при решении задач управления в динамических предметных областях. Предложены адекватные схемы реализации динамических прецедентных систем, показано, что для решения задач управления в сложных динамических системах целесообразно использовать гибридные динамические прецедентные системы, реализованные по схеме «ситуация + поток событий»-«план»-«сценарий»-«управляющее воздействие» либо «ситуация + граф»-«план»-«сценарий»-«действие», и обеспечивающие механизм адаптации и верификации на уровне сценариев.

Результаты работы позволяют использовать прецедентный подход при решении трудноформализуемых задач в динамических предметных областях.

Список использованной литературы

1. Case-Based Reasoning Research and Development / eds.: Ram A., Wiratunga N. Lecture Notes in Artificial Intelligence. 2011. Vol. 6880. – 498 p.
2. Черняховская Л. Разработка динамической модели процесса управления в проблемных ситуациях на основе базы знаний прецедентов. Управление в сложных системах. 1999. – С. 207–212.
3. Шерстюк В.Г. Динамическая сценарно-прецедентная интеллектуальная система для управления подвижными объектами. Искусственный интеллект. 2011. №4. – С. 362–373.
4. Pal S.K., Shiu S.C. Foundations of Soft Case-Based Reasoning. N.Y.: J. Wiley & Sons, 2004. – 274 p.
5. Xia Q., Rao M. Incorporating system dynamics in case-based reasoning for process operation support. Computational Cybernetics and Simulation: Proc. of 1997 IEEE Int. Conf. on Systems, Man, and Cybernetics. Orlando, 1997. Vol. 3. Pp. 2075–2080. doi: 10.1109/ICSMC.1997.635170
6. Ram A., Santamaria J.C. Continuous Case-Based Reasoning. Case-Based Reasoning: Proc. of AAAI-93 Workshop. Washington, 1993. – Pp. 86–93.
7. Martin F.J. Case-Based Sequence Analysis in Dynamic, Imprecise, and Adversarial Domains: tesi doctoral. Barcelona: Universitat Politècnica De Catalunya, 2004. – 285 p.

8. Jaere M., Aamodt A., Skaalle P. Representing temporal knowledge for case-based prediction. *Advances in case-based reasoning*. 2002. Vol. 2416. – Pp. 174–188.
9. Marling C., Shubrook J., Schwartz F. Towards case-based reasoning for diabetes management. *Computational Intelligence*. 2009. Vol. 25, №3. – Pp. 165–179.
10. Funk P., Robertson D. Capturing and Matching Dynamic Behavior in Case-Based Reasoning. *Progress in Case-Based Reasoning: Proc. of First UK Workshop*. London, 1995. – Pp.85–90.
11. Craw S. Agile case-based reasoning: A grand challenge towards opportunistic reasoning from experiences. *Grand Challenges in Reasoning from Experiences: Proc. of IJCAI-09 Workshop*. Pasadena, 2009. – Pp. 33–39.
12. Cordier A., Mascaret B., Mille A. Dynamic Case-Based Reasoning for Contextual reuse of Experience. *Case-Based Reasoning: Proc. of ICCBR'2010 Workshop*. Alessandria, 2010. Pp. 69–78.
13. Urdiales C., Perez E.J., Vrazquez-Salceda J., Sanchez-Marre M., Sandoval F. A purely reactive navigation scheme for dynamic environments using Case-Based Reasoning. *Automatic Robotics*. 2006. Vol. 21. – Pp. 65–78.
14. Floyd M., Esfandiari B. A Case-Based Reasoning Framework for Developing Agents Using Learning by Observation. *Tools with Artificial Intelligence: Proc. on 23rd IEEE Int. Conf. ICTAI*. Boca Raton, 2011. – Pp. 531–538.
15. Еремеев А.П., Куриленко И.Е., Смирнова А.Е. Разработка темпорального расширения методов рассуждений на основе прецедентов. *Интеллектуальные системы и информационные технологии: Труды конгресса IS&IT'11*. Москва, 2011. Том 1. – С. 50–59.
16. Sherstjuk V. Scenario-Case Coordinated Control of Heterogeneous Ensembles of Unmanned Aerial Vehicles. *Actual Problems of Unmanned Aerial Vehicles Developments: Proc. of 2015 IEEE 3rd Int. Conf*. Kyiv, 2015. – Pp.275–279.
17. Поспелов Д.А. Ситуационное управление: теория и практика. М.: Наука, 1986. 288 с.
18. Шерстюк В.Г. Основы теории динамических сценарно-прецедентных интеллектуальных систем. Херсон: Феникс, 2012. – 476 с.
19. Шерстюк В.Г. Сценарно-прецедентное управление эргатическими динамическими объектами. Saarbrucken: Lambert Academic Publishing, 2013. – 407 p.

References

1. Case-Based Reasoning Research and Development / eds.: Ram A., Wiratunga N. *Lecture Notes in Artificial Intelligence*. 2011. Vol. 6880. 498 p. doi: 10.1007/3-540
2. Chernjahovskaja L. Razrabotka dinamicheskoy modeli prozessa upravleniya v problemnykh situacziyakh na osnove bazy znaniy preczedentov [Development of a dynamic model of the control process in problem situations based on the case base]. *Control in complex systems*. 1999. Pp. 207–212.
3. Sherstjuk V.G. Dinamicheskaya scenarno-preczedentnaya intelektual'naya sistema dlya upravleniya podvizhnymi ob'ektami [Dynamic scenario-case intelligent system for controlling moving objects]. *Artificial Intelligence*. 2011. Vol. 4. Pp. 362–373.
4. Pal S.K., Shiu S.C. *Foundations of Soft Case-Based Reasoning*. N.Y.: J. Wiley & Sons, 2004. 274 p. doi: 10.1002/0471644676
5. Xia Q., Rao M. Incorporating system dynamics in case-based reasoning for process operation support. *Computational Cybernetics and Simulation: Proc. of 1997 IEEE Int. Conf. on Systems, Man, and Cybernetics*. Orlando, 1997. Vol. 3. Pp. 2075–2080. doi: 10.1109/ICSMC.1997.635170
6. Ram A., Santamaria J.C. Continuous Case-Based Reasoning. *Case-Based Reasoning: Proc. of AAAI-93 Workshop*. Washington, 1993. Pp. 86–93. doi: 10.1016/s0004-3702(96)00037-9
7. Martin F.J. *Case-Based Sequence Analysis in Dynamic, Imprecise, and Adversarial Domains: tesi doctoral*. Barcelona: Universitat Politecnica De Catalunya, 2004. 285 p.
8. Jaere M., Aamodt A., Skaalle P. Representing temporal knowledge for case-based prediction. *Advances in case-based reasoning*. 2002. Vol. 2416. Pp. 174–188. doi: 10.1007/3-540-46119-1_14
9. Marling C., Shubrook J., Schwartz F. Towards case-based reasoning for diabetes management. *Computational Intelligence*. 2009. Vol. 25, №3. Pp. 165–179. doi: 10.1111/j.1467-8640.2009.00336.x
10. Funk P., Robertson D. Capturing and Matching Dynamic Behavior in Case-Based Reasoning. *Progress in Case-Based Reasoning: Proc. of First UK Workshop*. London, 1995. Pp.85–90. doi: 10.1007/3-540-60654-8_24
11. Craw S. Agile case-based reasoning: A grand challenge towards opportunistic reasoning from experiences. *Grand Challenges in Reasoning from Experiences: Proc. of IJCAI-09 Workshop*. Pasadena, 2009. Pp. 33–39. doi: 10.1007/978-1-4899-7687-1_34
12. Cordier A., Mascaret B., Mille A. Dynamic Case-Based Reasoning for Contextual reuse of Experience. *Case-Based Reasoning: Proc. of ICCBR'2010 Workshop*. Alessandria, 2010. Pp. 69–78. doi: 10.1007/11891451_27

13. Urdiales C., Perez E.J., Vrazquez-Salceda J., Sanchez-Marre M., Sandoval F. A purely reactive navigation scheme for dynamic environments using Case-Based Reasoning. *Automatic Robotics*. 2006. Vol. 21. Pp. 65–78. doi: 10.1007/s10514-006-7231-8
14. Floyd M., Esfandiari B. A Case-Based Reasoning Framework for Developing Agents Using Learning by Observation. *Tools with Artificial Intelligence: Proc. on 23rd IEEE Int. Conf. ICTAI*. Boca Raton, 2011. – Pp. 531–538. doi: 10.1109/ictai.2011.86
15. Eremeev A.P., Kurilenko I.E., Smirnova A.E. Razrabotka temporal'nogo rasshireniya metodov rassuzhdenij na osnove precedentov [Development of the temporal expansion of case-based reasoning methods]. *Intellektual'nye sistemy i informacionnye tekhnologii: Trudy kongressa IS&IT'11 [Intelligent Systems and Information Technology: Proceedings of the IS & IT'11 Congress]*. Moscow, 2011. Vol. 1. Pp. 50–59.
16. Sherstjuk V. Scenario-Case Coordinated Control of Heterogeneous Ensembles of Unmanned Aerial Vehicles. *Actual Problems of Unmanned Aerial Vehicles Developments: Proc. of 2015 IEEE 3rd Int. Conf. Kyiv, 2015*. Pp.275–279. doi: 10.1109/apuavd.2015.7346620
17. Pospelov D.A. *Situacionnoe upravlenie: teoriya i praktika [Situational Control: Theory and Practice]*. Moscow: Nauka, 1986. 288 p.
18. Sherstjuk V.G. *Osnovy teorii dinamicheskikh scenarno-precedentnykh intellektual'nykh system [Fundamentals of the dynamic scenario-case intelligent systems theory]*. Kherson: Phenix, 2012. 476 p.
19. Sherstjuk V.G. *Sczenarno-precedentnoe upravlenie ergaticheskimi dinamicheskimi ob`ektami [Scenario-case control of ergatic dynamic objects]*. Saarbrucken: Lambert Academic Publishing, 2013. 407 p.