

УДК 004.89

Ю. М. ТКАЧЕНКО

*Национальный аэрокосмический университет им. Н. Е. Жуковского «ХАИ», Украина*

## РАЗРАБОТКА ИНСТРУМЕНТАЛЬНЫХ СРЕДСТВ ДЛЯ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ ПО АНАЛИЗУ ДЕЯТЕЛЬНОСТИ ОБУЧАЕМЫХ В ТЕХНИЧЕСКОМ ВУЗЕ

*Описана постановка задачи синтеза системы поддержки принятия решений по анализу деятельности обучаемых в технических вузах. Предложен подход к созданию такой системы на основе мультиагентной технологии. Применение данного подхода обеспечит повышение эффективности учебного процесса за счет снижения доли рутинного труда при формировании трасс обучения как отдельных студентов, так и студенческих групп, а также снижения рисков принятия неверных решений в процессе анализа деятельности обучаемых. Приведен метод оценки эффективности разработанной системы с использованием скользящего контроля.*

**Ключевые слова:** *технический вуз, преподаватели, учебный процесс, обучаемые, трасса обучения, система поддержки принятия решений, мультиагентная технология.*

### Введение

В области систем поддержки принятия решений существует ряд задач, которые можно решать, синтезировав средство автоматизации как систему, состоящую из агентов [1 - 3]. В эти задачи входят: эффективная организация параллельной работы, извлечение информации из инженерных баз знаний, интеллектуальный индивидуальный помощник лица принимающего решения (ЛПР) и т.д. Существуют различные варианты применений агентов к решению задач автоматизации управления учебным процессом [4, 5].

В учебном процессе высшей школы требуются постоянное сотрудничество преподавателей-специалистов, работающих в различных областях, с обучаемыми. Эти участники процесса, по сути, работают над одной задачей параллельно и независимо в течение значительных промежутков времени, причем хотя и существует возможность общения между собой, но задачи, которые они решают, часто не имеют общих точек соприкосновения и не требуют такого общения, хотя в последствии при аттестации выпускников необходимо выполнение определенных условий для удачного достижения поставленной цели.

Для эффективной координации деятельности как преподавателей, так и обучаемых и для обеспечения необходимой синхронизации получаемых результатов в рамках единой образовательной стратегии, необходимо создать некоторую (возможно, интеллектуальную) среду. Такая среда должна не только автоматизировать индивидуальную работу, но и помогать пользователям обмениваться инфор-

мацией и координировать свою работу по поиску глобально оптимального или близкого к таковому решения (зачастую локально оптимальное решение не выполняет общие для системы ограничения).

*Цель статьи* состоит в изложении подхода к синтезу системы поддержки принятия решений при управлении учебным процессом, с использованием мультиагентной технологии.

### 1. Постановка задачи исследования

В качестве исходных данных рассматривается структура технического вуза, а также технология организации учебного процесса при подготовке обучаемых по квалификационным уровням бакалавр, специалист и магистр.

Необходимо на основе анализа существующих подходов, стандартов и инструментальных средств определить рациональный путь построения диалоговой системы для поддержки участников учебного процесса (студентов, преподавателей, работников служб вуза).

### 2. Агентная парадигма как основа синтеза диалоговой системы с распределенным искусственным интеллектом

Работу по организации и поддержке учебного процесса выполняет множество различных подразделений вуза, и каждое из этих подразделений, в свою очередь, делится на множество структурных единиц, выполняющих свои локальные задачи. Поэтому методы поддержки принятия координирую-

щих решений в рамках одной службы (например, кафедры) можно распространить и на весь вуз в целом.

Попытки решения задачи эффективной организации командного взаимодействия предпринимаются различными учеными на протяжении последних 15 лет, при этом исследователи используют следующие методы и средства: технологии распределенного решения задач параллельного проектирования [6, 7], общие архитектуры или платформы инженерных и промышленных приложений [8, 9], мультиагентные системы [4, 5], мультиэкспертные системы [10] и т.д.

Системы поддержки коллективной работы делятся на два класса: мультиэкспертные системы (использующие концепцию «классной доски») и мультиагентные системы. В мультиэкспертных системах каждый специалист (иногда называемый «источником знаний») имеет доступ к «классной доске», на которой отражается информация, которую специалист хочет сообщить другим. В мультиагентной системе каждый агент независим и имеет собственное представление о текущей ситуации в системе. Это различие не столько в реализации, сколько в концепции. В случае «классной доски», мы имеем возможность видеть систему в целом, что положительно сказывается на ее целостности, но имеет ряд недостатков. Во-первых, «классная доска» может оказаться узким местом коммуникаций в системе. Во-вторых, возникают проблемы общего представления, поскольку разные инструменты используют различные формы представления своих данных, и нет особого резона доставлять данные в общее место, где далеко не каждый будет способен их понять. В-третьих, на протяжении жизненного цикла процесса обучения могут появляться новые инструментальные средства, внедрение которых будет порождать изменение структуры среды проектирования. В мультиагентной системе каждый агент строит собственную модель текущего решения, основываясь на своих данных и данных других агентов. В таких системах имеются коммуникационный протокол и формат сообщений (язык коммуникаций), в соответствии с которым должны оформляться запросы и ответы [3, 6]. Обобщенно можно сказать, что агенты автономны и гетерогенны, т.е. отсутствует единое управление. Коммуникации между агентами могут быть синхронными и асинхронными, направленными (peer-to-peer), общими (broadcast) или групповыми (multicast). Важно, что семантика сообщений между агентами должна быть высокого уровня. Это означает не тривиальную пересылку команд на запуск/останов, а полную реализацию информационных потоков между агентами в системе путем обмена сообщениями на языке, ана-

логичном языке высокого уровня в программировании [11 - 13].

Структура исследований в области многоагентных систем в настоящее время очень широка и сравнима с широтой исследований в области искусственного интеллекта. Это не случайно, поскольку обусловлено интегрированностью самого понятия такой системы, сложностью архитектуры и многообразием компонент каждого отдельного агента, многообразием математических и программных средств, используемых при его описании и разработке, сложностью структуры и разнообразием вариантов взаимодействия агентов между собой, сложностью и разнообразием компонент внешней среды, в которой функционируют агенты и т.д.

В нашем случае под агентом будем понимать механизм инкапсуляции и обмена распределенными знаниями и функциями. Каждый агент - это процесс, обладающий определенной частью знаний об организации и реализации учебного процесса и возможностью обмениваться этими знаниями с остальными агентами. В зависимости от типа, агент может поддерживать и интерфейс с пользователем. Под многоагентной системой в дальнейшем будем понимать многокомпонентную систему, состоящую из агентов со специфицированным интерфейсом.

### **3. Анализ возможностей применения существующих методов и средств построения систем интеллектуальных агентов для обеспечения принятия решений по организации учебного процесса в вузе**

Учитывая структуру технического вуза, а также изложенное выше описание подходов к организации коллективного взаимодействия компонентов диалоговой системы, можно сделать вывод о принципиальной возможности применения именно мультиагентной технологии при синтезе систем поддержки принятия решений по организации учебного процесса в техническом вузе. В пользу этого также говорит и возможность параллельной работы подразделений вуза, преподавателей, обучаемых и независимость на определенном этапе локальных решений между собой.

Использование технологии распределенных объектов на сегодняшний день может рассматриваться как необходимое, но, по-видимому, не достаточное условие для обеспечения единого информационного пространства (в том числе пространства знаний). С нашей точки зрения, наиболее приемлемые решения лежат в использовании агентно-ориентированной архитектуры в качестве каркаса системы, в которой агенты обеспечивают динамиче-

ское связывание и взаимодействие, используя стандартные интерфейсы и форматы данных (или знаний). При этом вся бизнес-логика работы системы может обеспечиваться различными средствами – от простейших приложений до интеллектуальных систем поддержки принятия решений. Важен тот факт, что использование агентного подхода в таком случае сделает изначально закрытые системы открытыми и обеспечит возможность интеграции их в единое информационное пространство [14]. При этом также не будут потеряны ценные данные (знания), хранящиеся в этих системах.

Применение агентного подхода к интеллектуальным системам поддержки принятия решений даст в результате возможность реализации СППР как единой системы с распределенным интеллектом, которая имеет множество положительных свойств: обработка информации, приближенная к местам ее получения, повышение надежности и отказоустойчивости системы в целом, эффективное распараллеливание работы различных структурных единиц вуза, обеспечение полной наблюдаемости системы (что абсолютно невозможно в случае использования монолитной системы).

Поддержка моделей взаимодействия, основанных на концепции языка взаимодействия агентов (Knowledge Query and Manipulation Language - KQML), требует соответствующих соглашений между всеми подсистемами [11]. Поэтому первый вопрос, который возникает в процессе сопряжения различных систем или компонент, связан с введением в архитектуру дополнительного уровня или класса агентов, ответственного за преобразование форматов информационного обмена.

Для поддержки обработки описанного выше контекста сообщений между подсистемами могут быть использованы специальные коммуникационные агенты (фасилитаторы, медиаторы или информационные агенты [12]), которые располагаются между отправителем и получателем сообщения и выполняют действия по стандартизации интерфейса, интеграции информации из различных источников, преобразованию запросов и ответов.

Использование фасилитаторов становится особенно важным в условиях интеграции новых информационных систем с существующими, которые не имеют необходимой степени гибкости в преобразовании входных/выходных форматов. Агенты (и другие программные компоненты системы) взаимодействуют с использованием фасилитаторов, которые могут выполнять преобразования специфических внутренних форматов представления данных (знаний) из/в стандартный формат обмена знаниями (Knowledge Interchange Format - KIF) [13]. Таким образом, каждый агент выполняет логический вывод

в своих внутренних терминах, а получает и передает информацию другим агентам в необходимой им для понимания форме посредством фасилитаторов.

Каждый фасилитатор отвечает за обеспечение интерфейса между локальным объединением агентов (в рамках подсистемы) и удаленными агентами и преследует тем самым 4 основных цели:

- 1) обеспечивает надежный механизм передачи сообщений,
- 2) маршрутизирует сообщения по назначению,
- 3) преобразует приходящие сообщения в формат представления адресата,
- 4) осуществляет мониторинг агентов.

Таким образом, использование промежуточных агентов (фасилитаторов), особенно при условии интеграции существующих систем, позволяет получить гомогенное информационное пространство для работы всей системы, а также возможность отслеживания несоответствий при формировании решений в отдельных подсистемах и обеспечения их корректировки для поддержки адекватного формирования согласованных комплексных решений масштаба структурной единицы (или вуза в целом). Результат будет зависеть от сложности исполнения фасилитаторов и от трактовки выполняемых ими задач (преобразование сообщений, корректировка результатов вывода подсистем и т.д.).

#### **4. Архитектуры и спецификации агентных систем для поддержки единого информационного пространства в рамках технического вуза**

Существующие варианты архитектур многоагентных систем и рациональный выбор архитектуры отдельного агента и многоагентной системы в целом существенно зависят от концептуальной модели агента, принятого для ее описания формализма и языка спецификаций, математической модели кооперации агентов при совместном функционировании в системе, на какое приложение или класс приложений ориентирована многоагентная система, а также от ряда других факторов. Тем не менее, можно выбрать некоторые характерные варианты, воплощающие в себе основные принципы конструирования архитектур и которые рассматриваются в сообществе специалистов как перспективные.

Основное назначение архитектуры взаимодействия агентов состоит в том, чтобы обеспечить скоординированное поведение агентов при решении общей или своих частных задач. Здесь можно выделить два основных варианта архитектур. В одном из них агенты не образуют иерархии и решают общую задачу полностью в распределенном варианте. В другом варианте координация распределенного

функционирования агентов в той или иной мере поддерживается специально выделенным агентом, который при этом относится к мета-уровню по отношению к остальным агентам. Существуют и, по-видимому, будут возникать более сложные, иерархически организованные схемы взаимодействия агентов.

Обобщенная классификация архитектур агентов основывается на парадигме, лежащей в основе принятой архитектуры. По этому признаку различают два основных класса архитектур [4]:

- архитектура, которая базируется на принципах и методах искусственного интеллекта, т.е. систем, основанных на знаниях («deliberative agent architecture», «архитектура разумного агента»);

- архитектура, основанная на поведении («reactive architecture») или «реактивная архитектура» (основанная на реакции системы на события внешнего мира).

На самом деле к настоящему времени среди разработанных архитектур не существует таких, о которых можно было бы определенно сказать, что она является чисто поведенческой или основана только на знаниях. Любая из разработанных архитектур является по сути гибридной, имея те или иные черты от архитектур обоих типов.

С другой стороны, независимо от лежащей в основе формализации парадигмы, архитектуры агентов классифицируются в зависимости от вида структуры, наложенной на функциональные компоненты агента и принятых методов организации взаимодействия его компонент в процессе работы. Как правило, архитектура агента организуется в виде нескольких уровней. Среди многоуровневых архитектур различают горизонтальную организацию взаимодействия уровней и вертикальную организацию.

Перечислим базовые характеристики, которыми наделяют агентов практически все исследователи [4], и которые используются в дальнейшем изложении:

- активность, способность к организации и реализации действий;

- реактивность, способность воспринимать состояние среды;

- автономность, относительная независимость от окружающей среды;

- общительность (или «социальное поведение», «social ability»), вытекающая из необходимости решать свои задачи совместно с другими агентами;

- целенаправленность, предполагающая наличие собственных источников мотивации, а в более широком плане, особых интенциональных характеристик.

Подход, который родился довольно давно в

рамках исследований в области искусственного интеллекта (ИИ), уже в конце 90-х на Западе начал активно переходить в русло практического применения в областях программирования, несвязанных напрямую с ИИ. Разработка агентных систем в Европе приобрела такие масштабы, что встал вопрос о стандартизации, для решения которого ещё в 90-х годах были основаны две организации MASIF (Mobile Agent System Interoperability Facility) и FIPA (Foundation of Physical Intelligent Agents), занявшиеся разработкой стандартов. Их работа вылилась в появление стандарта MASIF и стандарта FIPA, дающих рекомендации к созданию систем мобильных агентов и систем интеллектуальных агентов соответственно.

Логично предположить, что область применения мобильных агентов в основном ограничивается веб-технологиями, а также использованием «персональных помощников» PDA (Personal Digital Assistant). Стандарты же FIPA напротив, описывают архитектуру мультиагентной системы и структуру самого агента, делая в основном упор на реализуемые агентами функции, и таким образом, могут являться основой для синтеза промышленной системы поддержки принятия координирующих решений на машиностроительном предприятии. Кроме того, стандартах FIPA описывается возможность интеграции агентной системы с существующим программным обеспечением, что является важным свойством создаваемой системы.

Приведем наиболее значимые сравнительные характеристики агентных платформ по двум стандартам.

OMG MASIF (Object Management Group):

- Управление агентами,
- Перенос агентов,
- Именованное,
- Типы агентных систем,
- Учет агентов.

FIPA (Foundation for Intelligent Physical Agents):

- Управление агентами (White pages, Yellow pages, Directory facilitator),
- Агентная платформа (Name system, Directory, agent management),
- Взаимодействие агентов (сообщения, формальная семантика, протоколы),
- Интеграция с существующим ПО (ARB, Wrapper, интеграция систем без агентов),
- Сервис онтологий (термины, аксиомы, связи, запросы).

Модель агентной платформы согласно стандарту FIPA изображена на рисунке 1. Сравнительные характеристики наиболее распространенных агентных платформ приведены ниже в таблице 1.

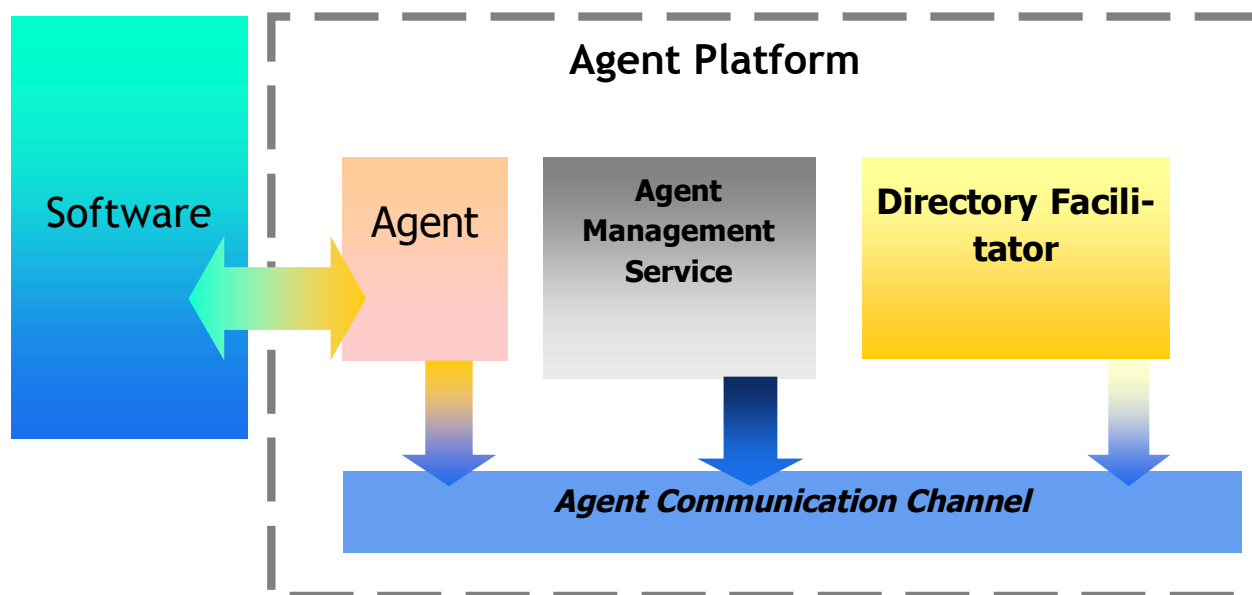


Рис. 1. Модель агентной платформы согласно стандарту FIPA

Таблица 1

Наиболее распространенные агентные платформы

Платформа	Особенности
FIPA OS	Оболочка, поддерживающая основные стандарты FIPA, для создания интеллектуальных агентов. Содержит среду, планировщик задач. Канал сообщений, оболочку для агентов формата JESS. Поддержка RMI, HTTP.
JADE	Java Agent Development Framework, позволяющая распределять агентов по вычислительным платформам, удаленно конфигурировать. Допускает размещение агентов на мобильных устройствах.
Grasshopper	Попытка реализовать платформу, поддерживающую оба стандарта (OMG MASIF и FIPA). Используется в основном для исследовательских проектов.
Zeus	Платформа для создания распределенных мультиагентных систем, реализующих коллаборативные принципы работы (совместно работающее над одной задачей сообщество агентов).
JINI	Обеспечивает предоставление базовых сервисов для мобильных агентов. Не является полноценной агентной платформой.
AgentBuilder	Коммерческая платформа и средства для разработки интеллектуальных агентов и агентных приложений. Поддерживает стандарт обмена знаниями KQML, а также кросс-платформенное размещение на основе CORBA.
BeeGent	Основана на понятиях «состояние» и «переход». Поддерживает агент-обложку для интеграции существующего ПО. Совместима со стандартом FIPA. Платная.
AgentFarm	Автономные агенты фреймворк-структуры. Платформа использует Java.
Aglets	Открытая разработка фирмы IBM. Основная идея – создание мобильных перемещаемых в рамках Java-окружения интернет-агентов. Использует собственный сервер Tahiti.

Исходя из возможного использования системы в качестве дополнительной (надстройки) к существующей можно сделать вывод о необходимости выбора стандарта FIPA как наиболее подходящего для такого типа систем ввиду возможности интеграции их с уже существующими.

При разработке системы также была проведена реализация прототипа подобной системы на основе существующего стандарта ДСТУ на использование

технологии CORBA для построения распределенных систем.

### 5. Оценка качества формируемых решений по анализу учебного процесса в вузе

Качество координирующих решений, формируемых прототипом системы поддержки принятия

решений (СППР), оценивалось на примере реальных данных (деятельность отдельных студентов и студенческой группы в целом в течение учебного года), из которых формировались сценарные примеры для обучения и проверки системы (правила для обучающей и предикаты для контрольной выборки), а также было проведено сравнение с результатами решения этих ситуаций несколькими экспертами.

Качество определялось общей ошибкой принятия неверного решения, получаемой путем взвешенного усреднения ошибок первого рода (нераспознанных ситуаций) и второго рода (ситуаций, классифицированных как коллизии, но не являющихся таковыми).

Само координирующее решение по внесению изменений в учебный процесс принималось в системе практически без ошибок (при достаточном размере обучающей выборки), т.к. координирующее решение почти однозначно определяется типом обнаруженной коллизии и редко допускает различные варианты. Поэтому основная задача системы состоит именно в распознавании признаков потенциальных коллизий и прогнозировании возможных рассогласований, что и было проверено с помощью разработанного прототипа СППР.

Для оценки качества работы системы применялся метод скользящего контроля [14, 15], а именно разбиение с одним отделяемым объектом (LOOCV).

Суть метода скользящего контроля заключается в следующем. Выборка  $X^k = \{x_1, \dots, x_k\}$  с известными ответами делится на два непересекающихся множества  $X^1 \cup X^c = X^k : X^1 \cap X^c = \emptyset$ , одно из которых используется для обучения, второе для контроля. Скользящий контроль с одним отделяемым объектом подразумевает перебор всех вариантов разбиения с одним объектом (примером или прецедентом) в качестве тестового и использование остальных прецедентов для обучения, после чего усредняются значения ошибок по всем таким разбиениям.

Выделяется один объект из выборки –  $\{X^i\}_1^f : X^i \cap X^j = \emptyset, i \neq j, \bigcup_{i=1}^f X^i = X^k$ . Тогда общая ошибка будет определяться как средняя ошибка по всем вариантам разбиений, то есть:

$$P(a(X^k) \neq y^*) \approx \frac{1}{f} \sum_{i=1}^f P(a(X^i) \neq y^* | \bigcup_{j \neq i} X^j),$$

где  $P$  – средняя ошибка (как несовпадение результата решения, выданного системой, с правильным решением  $y^*$ ).

Для случая распознавания ситуаций или клас-

сификации (который имеет место с диссертационной работе), вместо усреднения ошибки по разбиениям используется простой подсчет неправильно классифицированных объектов (прецедентов).

Метод скользящего контроля такого вида (в противоположность методу «5-2» или  $k$ -кратному разбиению) был выбран для избегания попадания нескольких схожих обучающих примеров в одну часть разбиения, тем самым возможно получение смещенной оценки ошибки.

Однако следует заметить, что формируемые решения в основном представляют собой оценивание уровня знаний обучаемого. При этом необходимо отдельно оценивать ошибки первого и второго рода, то есть ошибку распознавания при существующей коллизии и ошибочное определение нормальной ситуации как коллизии.

Анализ статистики затрат при возникновении ошибок первого и второго рода (по обучающей выборке) показывает, что затраты на исправление неправильного решения для ошибок первого и второго рода приблизительно равны  $P(y = +1) = 73\%$  и  $P(y = -1) = 27\%$  соответственно (если взять за 100% сумму этих затрат по одной и той же ситуации), т.е. убытки от нераспознанной коллизии намного больше, чем убытки от «ложной» коллизии (при условии что решение по ней будет принято, т.е. будут внесены изменения в учебный процесс).

Так как ошибки неравнозначны, то для оценки общей ошибки системы необходимо брать взвешенные значения ошибок первого и второго рода с весами соответственно 0,73 и 0,27 (несмотря на то, что ошибочно распознанная коллизия может быть легко проигнорирована пользователем при составлении результирующих документов).

Для тестирования была составлена выборка из 30 наиболее типовых прецедентов, имевших место в статистике проведения учебного процесса в вузе за определенный период. С привлечением экспертов для каждого случая были выделены обучающие правила, которые использовались при моделировании, а также предикаты, которые использовались в качестве исходных при проверке на каждом шаге разбиения.

Для сравнения результатов моделирования указанные ситуации были предложены двум экспертам (ведущим преподавателям) для решения. Сводные результаты представлены в таблице 2 (взвешенная ошибка вычислялась с использованием ранее указанных весов ошибки I и II рода).

Можно видеть, что имеющиеся в прототипе СППР правила позволяют формировать решения, вероятность ошибки в которых находится на уровне 16%, что приблизительно соответствует значению

Таблица 2

Результаты моделирования процесса формирования координирующих решений

	эксперт 1	эксперт 2	СППР	эксперт 1, %	эксперт 2, %	СППР, %
Пропущенные коллизии (ошибка I рода)	7	3	4	23,33%	10,00%	13,33%
Излишние коллизии (ошибка II рода)	4	3	7	13,33%	10,00%	23,33%
Взвешенная ошибка				20,63%	10,00%	16,03%

ошибки эксперта. Можно также сделать вывод, что качество работы системы определяется качеством обучения базы знаний, которую она использует для распознавания и решений потенциальных коллизий. Таким образом, увеличивая размер выборки, или проведя извлечения знаний экспертов, можно добиться улучшения качества работы системы. Главное же преимущество СППР состоит в том, что эффективность ее работы постоянна, в то время как человеку свойственно увеличение ошибок из-за большого количества ситуаций, что имеет место при реализации учебного процесса в техническом вузе.

В качестве основы для оценки важности ошибок первого и второго рода была взята статистика проведения учебного процесса в Национальном аэрокосмическом университете им. Н. Е. Жуковского в 2012 году. При составлении обучающей и контрольной выборок были взяты данные по обучению студентов на специальности «Программное обеспечение систем» и соответствующая статистика по корректировке учебного процесса.

### Выводы

1. Проанализированы возможности применения существующих методов и средств построения информационных систем для обеспечения принятия координирующих решений при реализации учебного процесса в техническом вузе. В качестве основы для построения подобной системы была выбрана многоагентная парадигма как программная реализация типового фрагмента СППР в виде интеллектуального агента.

2. Рассмотрены архитектуры и спецификации агентных систем для поддержки единого информационного пространства в рамках технического вуза, и был сделан вывод об их пригодности для реализации СППР по формированию координирующих решений.

3. Приведена оценка эффективности разработанного метода формирования координирующих решений. Оценка проводилась с помощью метода скользящего контроля с одним отделяемым объектом. (LOOCV). В результате моделирования был получен уровень общей ошибки риска принятия

неверных решений в ходе реализации учебного процесса (около 16%).

### Литература

1. Vittikh, V. A. *Developing of Multi-Agent System for the Decision Making Process for Companies with Networking Organisation [Electronic resource] / V. A. Vittikh, P. O. Skobelev // XVII World Congress on Simulations : thes. rep. – Switzerland, Lousanna, 2000. – Access mode: <http://eup.ru/Documents/2002-08-05/F182.asp>. – 3.05.2015.*
2. Рассел, С. *Искусственный интеллект: современный подход [Текст] / С. Рассел, П. Норвиг. – М. : Изд. дом «Вильямс», 2006. – 1408 с.*
3. Jennings, N. R. *Developing Industrial Multi-Agent systems [Text] / N. R. Jennings, J. M. Corera, I. I. Laresgoiti // In Proceedings of First International Conference on Multi-Agent systems : thes. rep. – San-Francisco, 1995. – P. 17 – 24.*
4. Городецкий, В. И. *Многоагентные системы (обзор) [Текст] / В. И. Городецкий, М. С. Грушинский, А. В. Хабалов // Новости искусственного интеллекта. – 1998. – № 2. – С. 64 - 116.*
5. Тарасов, В. Б. *От многоагентных систем к интеллектуальным организациям: философия, психология, информатика [Текст] / В. Б. Тарасов. – М. : Эдиториал УРСС, 2002. – 352 с.*
6. Гаврилова, Т. А. *Базы знаний интеллектуальных систем [Текст] / Т. А. Гаврилова, В. Ф. Хорошевский. – СПб. : Пупер, 2000. – 384 с.*
7. Sriram, D. *An object-oriented frame-work for Collaborative Engineering Design [Electronic resource] / D. Sriram, R. Logcher, A. Wong, S. Ahmed // Lecture Notes in Computer Science. – Springer-Verlag, 1991. – P. 51-92. – Access mode: <http://www.mel.nist.gov/msidstaff/>. – 3.05.2015.*
8. SHADE: *Technology for Knowledge-Based Collaborative Engineering [Electronic resource] / J. M. McGuire, D. R. Kuokka, J. C. Weber, T. R. Gruber, G. R. Olsen // Journal of Concurrent Engineering: Applications and Research. – 1993. – Vol. 1(3). – Access mode: <http://citeseer.ist.psu.edu/mcguire93shade.html>. – 3.05.2015.*
9. Cutkovsky, M. R. *An Experiment in Integrating*

*Concurrent Engineering Systems [Electronic resource] / M. R. Cutkovsky, R. S. Engelmores, R. E. Fikes et al. // IEEE Computer. – 1993. – P. 28-37. – Access mode: <http://doi.ieeecomputersociety.org/10.1109/2.179153>. – 3.05.2015.*

10. Gero, J. S. *Expert systems in computer-aided design [Text] / J. S. Gero* – Elsevier Science Publishers B. V. – 1987. – P. 435 - 465.

11. Finin, T. *KQML as an Agent Communication Language [Text] / T. Finin, Richard Don McKay Fritzson, R. McEntire // Third International Conference on Information and Knowledge Management : thes. rep. - ACM Press, 1994. – P. 456 - 463.*

12. Виттих, В. А. *Управление открытыми системами на основе интеграции знаний [Текст] / В. А. Виттих // Автоматрия. – 1999. – № 3. – С. 38 -49*

13. Wiederhold, G. *Mediators in the Architecture of Future Information Systems [Text] / G. Wiederhold // IEEE Computer. – 1992. – V. 5, № 3. – P. 38 - 49.*

14. Джексон, П. *Введение в экспертные системы. [Текст] / П. Джексон. – М. : Изд. дом «Вильямс», 2001. – 624 с.*

15. Воронцов, К. В. *Комбинаторные оценки качества обучения по прецедентам [Текст] / К. В. Воронцов // Докл. РАН. – 2004. – Т. 394. – С. 175–178.*

*Поступила в редакцию 3.06.2015, рассмотрена на редколлегии 23.06.2015*

**Рецензент:** д-р техн. наук, проф., проф. каф. программной инженерии С. Ю. Шабанов-Кушнаренко, Харьковский национальный университет радиоэлектроники, Харьков.

## **РОЗРОБКА ІНСТРУМЕНТАЛЬНИХ ЗАСОБІВ ДЛЯ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ ПО АНАЛІЗУ ДІЯЛЬНОСТІ УЧНІВ В ТЕХНІЧНОМУ ВНЗ**

*Ю. М. Ткаченко*

Описана постановка задачі синтезу системи підтримки прийняття рішень по аналізу діяльності учнів в технічних вузах. Запропоновано підхід до створення такої системи на основі мультиагентної технології. Застосування даного підходу забезпечить підвищення ефективності навчального процесу за рахунок зниження частки рутинної праці при формуванні трас навчання як окремих студентів, так і студентських груп, а також зниження ризиків прийняття невірних рішень в процесі аналізу діяльності учнів. Наведено метод оцінки ефективності розробленої системи з використанням ковзного контролю.

**Ключові слова:** технічний вуз, викладачі, навчальний процес, учні, траса навчання, система підтримки прийняття рішень, мультиагентна технологія.

## **DEVELOPMENT OF TOOLS FOR SUPPORT OF DECISION-MAKING ACCORDING TO THE ANALYSIS OF ACTIVITY OF TRAINEES IN TECHNICAL COLLEGE**

*I. M. Tkachenko*

It describes the formulation of the problem of synthesis of decision support systems for the analysis of the activities of students in technical colleges. An approach to the creation of such system based on multi-agent technology is described. Applying this approach will improve the efficiency of educational process by reducing the share of routine work in the formation of runs of training as individual students and student groups, as well as reduce risks making the wrong decisions in the process of analysis of the trainees. The method of an assessment of efficiency of the developed system with use of the sliding control is given.

**Key words:** technical college, teachers, educational process, trainees, route of training, decision support system, multi-agent technology.

**Ткаченко Юлія Михайлівна** - аспірант каф. інформатики, Національний аерокосмічний університет ім. Н. Е. Жуковського «Харьковский авиационный институт», Харьков, Украина, e-mail: juli.tkachenko@mail.ru.