

УДК 004.896

А.П. Пинков, А.Н. Афанасьев, Н.Н. Войт, Д.С. Канев

РАЗРАБОТКА МЕТОДОВ И СРЕДСТВ КОМПЬЮТЕРНЫХ СИСТЕМ ОБУЧЕНИЯ АВТОМАТИЗИРОВАННОМУ ПРОЕКТИРОВАНИЮ МАШИНОСТРОИТЕЛЬНЫХ ОБЪЕКТОВ¹

Пинков Александр Петрович, кандидат экономических наук, окончил ульяновский филиал Куйбышевского планового института. Исполняющий обязанности ректора Ульяновского государственного технического университета. Имеет более 50 научных работ, в том числе монографию и статьи, по экономике, планированию, маркетингу, организации производства, организации высшего образования и корпоративного обучения. [e-mail: rector@ulstu.ru].

Афанасьев Александр Николаевич, доктор технических наук, профессор, окончил радиотехнический факультет Ульяновского политехнического института. Первый проректор, проректор по дистанционному и дополнительному образованию УлГТУ. Имеет более 200 статей в области САПР. Область научных интересов: автоматизированные системы обучения, организация вычислительных процессов и структур ЭВМ, проектирование интеллектуальных систем, САПР, управление сложными потоками работ, диаграмматика графических языков. [e-mail: a.afanasev@ulstu.ru].

Войт Николай Николаевич, кандидат технических наук, окончил факультет информационных систем и технологий УлГТУ, доцент кафедры «Вычислительная техника» УлГТУ. Имеет более 114 научных статей в области интеллектуальных САПР, Case-, Cals-технологий. Область научных интересов: интеллектуальные системы разработки сложных автоматизированных систем, автоматизированные среды обучения, графические языки и грамматики. [e-mail: n.voit@ulstu.ru].

Канев Дмитрий Сергеевич, окончил факультет информационных систем и технологий УлГТУ, старший научный сотрудник кафедры «Вычислительная техника» УлГТУ. Имеет более 20 статей в области САПР. Область научных интересов: разработка и внедрение программно-аппаратных платформ, способствующих поддержке, интенсификации и повышению вовлеченности обучающихся в образовательный процесс с помощью информационных технологий. [e-mail: dima.kanev@gmail.com].

Аннотация

В настоящее время наиболее перспективными направлениями в области компьютеризированного обучения являются адаптивные и мультимедиа технологии.

Основной целью адаптивных обучающих систем (АОС) является реализация управления процессом обучения с учётом индивидуальных особенностей пользователей. Адаптивные методы позволяют сократить время и повысить эффективность процесса обучения за счёт удержания пользователей в оптимальной зоне обучения, изменяя последовательность предъявления материала и заданий, содержание, темп обучения и нагрузку.

В статье проведён анализ существующих методов к построению адаптивных автоматизированных обучающих систем, определены их преимущества и недостатки. Предложены АОС с авторским методом синтеза траектории обучения, метод корректировки профиля обучаемого посредством автоматического анализа выполненных им операций в программных пакетах автоматизированного проектирования (на примере системы автоматизированного проектирования КОМПАС).

Ключевые слова: автоматизированная обучающая система, модель предметной области, модель обучаемого, методы адаптации, система рекомендаций.

¹ Исследование выполнено при финансовой поддержке РФФИ и Правительства Ульяновской области в рамках научного проекта № 16-47-732152. Исследование поддержано грантом Министерства образования и науки Российской Федерации, проект № 2.1615.2017/ПЧ.

DEVELOPMENT OF METHODS AND MEANS OF COMPUTER SYSTEMS FOR MACHINE OBJECTS CAD TRAINING

Aleksandr Petrovich Pinkov, Candidate of Economics, graduated from Ulyanovsk branch of Kuibyshev Planning Institute; Acting Rector of Ulyanovsk State Technical University, an author of more than 50 papers, a monograph, and articles in the field of economics, planning, marketing, production engineering, higher education organization, and corporate training. e-mail: rector@ulstu.ru.

Aleksandr Nikolaevich Afanasev, Doctor of Engineering, Professor; graduated from the Radioengineering Faculty of Ulyanovsk Polytechnic Institute; First Vice-Rector for Distance and Extended Education at Ulyanovsk State Technical University; an author of more than 200 articles in the field of CAD, interested in automated training systems, computational process and computer structure organization, intelligent system design, CAD, composite workflow control, graphic language diagrammatics. e-mail: a.afanasev@ulstu.ru.

Nikolai Nikolaevich Voit, Candidate of Engineering, Associate Professor at the Department of Computer Science of Ulyanovsk State Technical University; graduated from the Faculty of Information Systems and Technologies of Ulyanovsk State Technical University; an author of more than 114 articles in the field of intelligent CAD systems, CASE and CALS technologies; interested in intelligent systems for development of complex computer-aided systems, automated training systems, graphical languages and grammatics. e-mail: n.voit@ulstu.ru.

Dmitrii Sergeevich Kanev, graduated from the Faculty of Information Systems and Technologies of Ulyanovsk State Technical University; Chief Staff Scientist at the Department of Computer Science of Ulyanovsk State Technical University; an author of more than 20 articles in the field of CAD; interested in development and distribution of firmware systems intended for maintenance, intensification, and increase of trainee involvement in educational process with the use of information technologies. e-mail: dima.kanev@gmail.com.

Abstract

Currently, the most promising areas in the field of computer-based training are adaptive and multimedia technologies.

The main objective of adaptive learning systems (ALS) is the implementation of learning management taking into account the individual characteristics of users. Adaptive methods can reduce the time and improve the efficiency of the learning process at the expense of the users held in an optimal training zone, changing the sequence of presentation material and tasks, content, learning pace and load.

The article provides the analysis of the existing methods for construction of adaptive automated training systems, their advantages and disadvantages are determined. ALS with the author's method of synthesis of a learning path, the method for correcting the student's profile through automatic analysis of operations carried out in CAD software packages (by the example of COMPASS CAD system).

Key words: automated training systems, domain model, student model, adaptation methods, recommendations system.

ВВЕДЕНИЕ

В настоящее время наиболее перспективными направлениями в области компьютеризированного обучения являются адаптивные и мультимедиа технологии.

Основной целью адаптивных систем является реализация управления процессом обучения с учётом индивидуальных особенностей пользователей. Адаптивные методы позволяют сократить время и повысить эффективность процесса обучения за счёт удержания пользователей в оптимальной зоне обучения, изменяя последовательность предъявления материала и заданий, содержание, темп обучения и нагрузку.

В состав математического обеспечения адаптивной системы входят модель профиля обучаемого и модель предметной области, на основе которых формируется персонализированная траектория обучения. Формирование

профиля обучаемого производится путем тестирования, которое позволяет проверить текущий уровень знания. При обучении пакетам систем автоматизированного проектирования (САПР), кроме теоретической подготовки, обучающая система должна обеспечить освоение практических навыков работы с САПР, т. е. выполнение проектных операций, а также оценить степень текущих умений и навыков.

Несмотря на значительные успехи в области разработки адаптивных обучающих систем (АОС), задачи интеграции обучающих систем с САПР в полном объёме не решены.

В работе рассматривается АОС с авторским методом синтеза траектории обучения, предлагается метод корректировки профиля обучаемого посредством автоматического анализа выполненных им операций в САПР (на примере САПР КОМПАС).

1 Анализ существующих подходов

Ниже приведен анализ основных методов и систем, используемых в адаптивном обучении.

В работе [1] предметная область представляет собой множество элементов курса, находящихся в отношении порядка. Элементами являются теоретические блоки, практические задания, тесты, упражнения и т. д. Модель обучаемого имеет оверлейный вид. Пространство обучения делится на три области: область освоенного материала, неосвоенного и доступного для освоения. При этом формулируется цель обучения. Управление процессом обучения заключается в достижении состояния освоенности пространства знаний, удовлетворяющее цели обучения. Основным недостатком такого подхода и системы в целом является отсутствие формирования индивидуальной траектории обучения.

В системе IDEA [2] модель предметной области представлена в виде деревьев, которые содержат учебный материал, тренировочные упражнения, справочную информацию. Модель обучаемого представлена в виде базы данных истории обучения, в которой содержится информация обо всех событиях, произошедших в процессе обучения. В основу модели процесса управления обучением положена система принятия решения в виде продукций, оперирующих с текущей стратегией (задачей обучения), базой данных истории обучения, типом обучаемого, локальными событиями и деревом учебных целей. В результате формируется учебный план. Недостатками подхода являются: трудоёмкость создания курсов (в среднем на подготовку 1 часа урока требуется 10–15 часов работы автора), отсутствует интеграция обучающей системы с накопленным корпоративным опытом.

В работе Касьяновой Е.В (система WAPE) [3] модель предметной области представлена как множество элементов знаний, связанных отношениями порядка и часть-целое. Элементы знаний имеют ссылки на ресурсы курса (элементарные информационные ресурсы учебников и задачников, а также примеры, тесты и проекты), содержание которых относится к данной единице знаний. Модель обучаемого представляет собой вероятностную оверлейную модель, содержащую условную вероятность, что обучаемый обладает определённым знанием. Для прохождения определённой темы система позволяет генерировать траекторию обучения на основе сравнения текущего уровня знания обучаемого с необходимыми для понимания рассматриваемой темы. Алгоритм проверяет, все ли предварительные знания достаточно усвоены студентом, и находит единицы знаний, нуждающиеся в изучении. После этого генерируются последовательность подцелей и последовательность информационных элементов для изучения. Недостатком является отсутствие интеграции обучающей системы с накопленным корпоративным опытом.

В работе Доррера А.Г. [4] модель предметной области представляется как последовательность модулей с различной сложностью. Каждый модуль содержит теоретический и справочный материалы, задания для самостоя-

тельной работы, а также набор контрольных материалов для самопроверки и получения оценки. Модель процесса обучения построена на основе цветных сетей Петри.

В зависимости от уровня ответов обучаемого возможны несколько сценариев обучения:

- ответы верные, в этом случае изучение данного модуля завершается и возможен переход к следующему модулю, а также повышение уровня сложности;
- ответы неточные, в этом случае обучаемый должен изучить дополнительный материал и затем пройти повторное тестирование;
- ответы абсурдные, в этом случае обучаемый должен изучить материал модуля с самого начала;
- возможно понижение уровня сложности.

Основным недостатком является слабая адаптивная составляющая.

В работе [5] модель предметной области рассматривается как множество учебных курсов. Для построения оптимальной траектории обучения используется генетический алгоритм, где генотипом является последовательность отобранных учебных курсов. В качестве оптимального размера начальной популяции выбрано 50, большая численность увеличит вероятность найти более качественное решение, однако это приводит к существенным задержкам по времени. Функции приспособленности учитывают результаты предварительного тестирования, степень связанности учебных материалов, сложность и изученность материалов. В качестве метода отбора используется метод рулетки, операция мутации выполняется через изменение последовательности учебных планов. Этот подход не предусматривает динамическое изменение учебно-проектного материала.

В работе [6] описывается система автоматической генерации персональных обучающих рекомендаций. Модель предметной области представляет множество неструктурированных учебных материалов, которые автоматически формируются из сети Интернет с помощью технологий индексации и интеллектуального анализа текста на основе программного обеспечения Nutch. Модель обучаемого содержит взвешенную последовательность обучающих элементов, которые ему интересны. Обучаемые группируются на основе сходства и различий своих предпочтений, для этого могут использоваться различные методы кластеризации. Рекомендации строятся на основе проявленных интересов обучаемого через анализ посещённых веб-страниц, при этом результаты ранжируются в соответствии с мерой (TF-IDF) сходства содержания. Данный подход может использоваться для неформального обучения, использование его механизмов для систем формального обучения (а именно такие системы являются наиболее эффективными с точки зрения качества и времени обучения) является проблематичным. К тому же в нем отсутствует оценка качества содержания интернет-источников.

В работе [7] модель предметной области содержит три составляющие: структуру, организованную в дерево (курс, главы, разделы и подразделы), учебные материалы и тестовые задания. Модель обучаемого представля-

ет собой оверлейную модель, которая инициализируется после предтеста. Для анализа результатов тестирования используется теория Демпстера–Шафера. Уровень знания описывается тремя вариантами: низкий, средний и мастер. Адаптация происходит после теста, система вычисляет уровень знаний и обновляет модель обучаемого. Недостатком является слабая адаптация.

В работе [8] модель предметной области представлена в виде графа, в котором вершинами являются понятиями предметной области. Модель обучаемого состоит из трёх компонентов: профиль обучаемого; модель знаний – оверлейная модель над понятиями предметной области; предпочитаемый стиль обучения. Модель адаптации строится на основе продукций. Система использует сочетание техник адаптивной навигации и адаптивного представления.

В работе [9] учитываются стиль мышления обучаемого. Модель предметной области содержит 12 типов материалов: 8 теоретических и 4 практических. Модель обучаемого описывается 3 субмоделями: цели и предпочтения (какие курсы хочет посетить, каковы предпочтения, такие как шрифт, тип, размер, цвет и другие параметры, связанные с интерфейсом); мышление и стиль обучения (содержит информацию о конкретном пути обучении и подходе к обучению); знание и исполнение (результаты тестов, проектов, задач). Система поддерживает следующие стили мышления: теоретик, организатор, новатор, гуманитарный. Стили обучения: активный, рефлексивный, дедуктивный, индуктивный, визуальный, словесный, последовательный, глобальный. Модель процесса обучения строит наиболее подходящую траекторию обучения с использованием адаптивной навигации, выбором содержания и аннотированных ссылок, которые согласуются с профилем ученика. Траектория обучения выбирается на основе формулы, которая учитывает стиль мышления, стиль обучения и уровень знаний.

В системе GRAPPLE [10, 11] модель предметной области описывается через понятия и отношения, с понятиями могут быть связаны информационные ресурсы и факты. Модель обучаемого – оверлейная. Модель адаптации строится на основе множества правил «если–то», используя модели предметной области и обучаемого, принимает решение, какую информацию и как отобразить (используются механизмы адаптивного представления, адаптации на уровне содержания, адаптации на уровне ссылок). В рамках проекта разработан язык GAL (Generic Adaptation Language).

Общими недостатками рассмотренных методов и средств являются: отсутствие механизмов интеграции с пакетами САПР; отсутствие динамического наполнения модели предметной области, что не позволяет автоматически включать гибридные знания и единицы опыта предприятия в процесс обучения; большая трудоемкость наполнения учебно-практическим материалом; отсутствие виртуальной составляющей (в первую очередь тренажерных систем), что затрудняет формирование практических компетенций обучающихся.

2 СТРУКТУРА АОС

Обобщенная структура АОС представлена на рисунке 1. Рассмотрим компоненты АОС.

1. Блок обучения отвечает за обучение (подбор материалов, тестирование, построение сценария обучения). Предметная область хранит материал для обучения, множество учебных материалов и связей между ними. База тестов и заданий содержит тестовый материал для проверки уровня знаний обучаемого и практические задания для проверки умений и навыков. Профиль обучаемого хранит информацию об уровне текущих компетенций обучаемого, результаты тестирования и т. д. Построение траектории обучения – построение траектории обучения и подбор тестов для обучаемого на основе анализа данных его профиля и информации из предметной области. Сценарий обучения – последовательность материалов для изучения обучаемым и подборка тестовых заданий. Контроль – множество тестовых заданий для проверки усвоения материала.

2. Блок рекомендаций отвечает за построение рекомендаций и диагностику уровня навыков обучаемого. Генератор операций отслеживает действия обучаемого и кодирует их для дальнейшего анализа. Генератор состояния формирует состояние проекта на основе операций. Состояние проекта хранит историю состояний проекта при работе в САПР. Факты – база фактов, полученная из множества операция и состояний проекта. Генератор правил заполняет правила для экспертной системы (ЭС) по определенным алгоритмам. Эксперт – эксперт по САПР, который заполняет базу правил ЭС. ЭС – механизм подбора рекомендаций на основе фактов и правил. Рекомендации – сформированные рекомендации для обучаемого. Анализ рекомендаций – корректировка профиля пользователя на основе предоставленных рекомендаций.

Процесс обучения строится в соответствии со следующими шагами.

1. Обучаемый проходит тестирование, на основе результатов которого происходит формирование текущего сценария обучения с использованием данных из баз предметной области, тестов и заданий и профиля обучаемого.
2. В процессе работы с САПР генератор операций обрабатывает действия пользователя, на основе данной информации формируются факты для ЭС.
3. На основе заложенных правил и полученных фактов ЭС формирует рекомендации и предоставляет обучаемому.
4. На основе выполненных операций, которые привели к выдаче рекомендаций, корректируется профиль обучаемого.
5. После изменения профиля обучаемого формируется новый сценарий обучения.

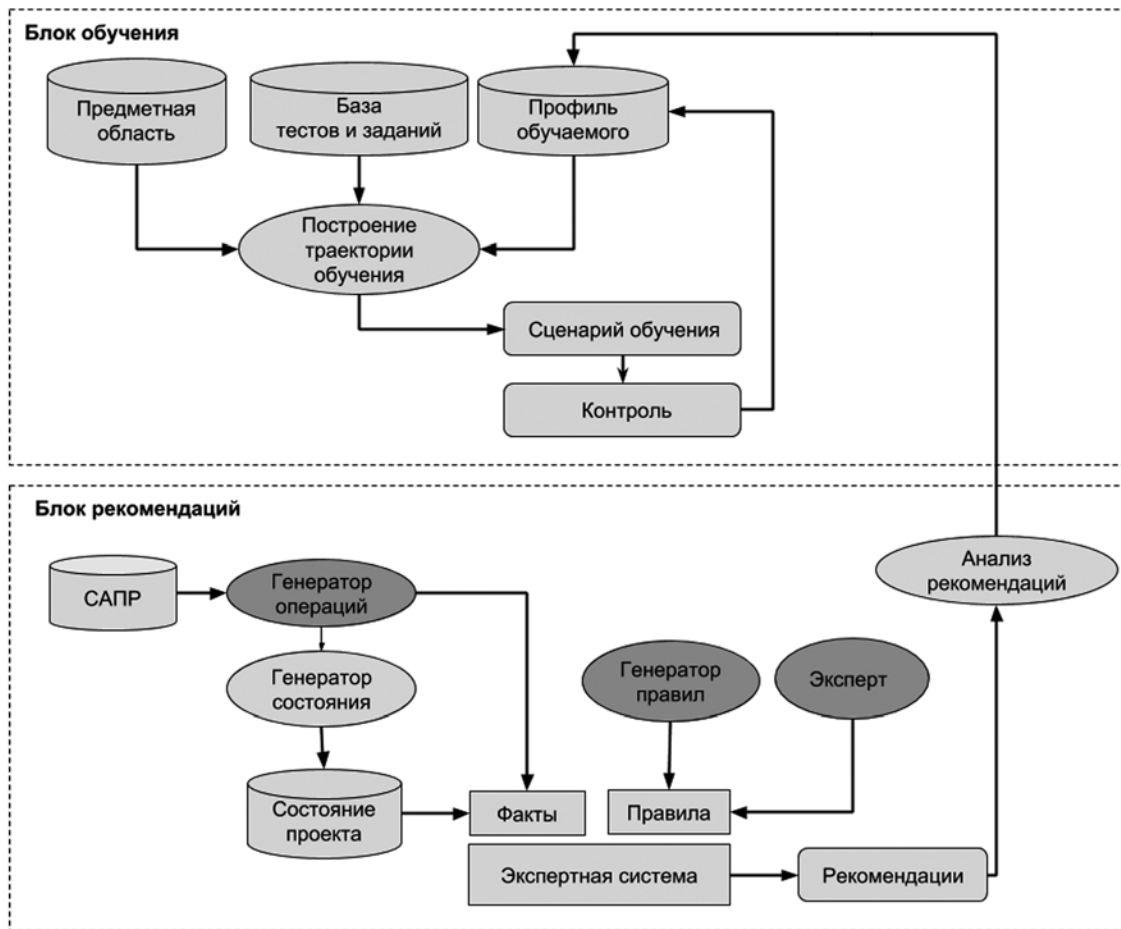


Рис. 1. Схема обучающей системы

3 Модели обучающей системы

Модель предметной области имеет вид:

$$O = (PSL, T, R, F, Ax),$$

где $PSL = \{psl_i | i = 1..x\}$ – множество проектных решений,

T – термины прикладной области, которую описывает онтология.

Множество терминов представлено в виде:

$$T = \{C, In\},$$

где $C = \{A, P, D, GOAL, COMP\}$ – множество классов онтологии (A – атомы знаний, P – понятия, D – учебный материал, $GOAL$ – цель обучения, $COMP$ – компетенции), у класса «Учебный материал» определён слот «являетсяСправочником» с диапазоном значений *истина* или *ложь*,

In – множество объектов классов онтологии.

R – множество отношений между объектами онтологии:

$$R = \{R_{learn}, R_{part}, R_{next}\},$$

где R_{learn} – бинарное отношение «изучается_в», имеющее семантику «connected_to» и связывающее объекты классов онтологии («Атом», «Понятие») с объектами класса «УчебныйМатериал»,

R_{part} – бинарное отношение «состоит_в», имеющее семантику «part_of» и связывающее объекты классов онтологии («Атом», «Понятие») с объектами классов «Понятие», «ЦельОбучения»,

R_{next} – бинарное отношение «изучается_после», имеющее семантику «after_of» и связывающее объекты классов онтологии («Атом», «Понятие») с объектами класса «Понятие» и «Атом».

Множество интерпретирующих функций представлено в виде:

$$F = \{Fatom_op, Fpsl_a, Fedu, Fdefine, Fsimilar, T\},$$

где $Fatom_op: A \rightarrow \{Operation\}$ – функция отображения объекта класса «Атом» на множество операций проектного решения,

$Fpsl_a: PSL \rightarrow \{A\}$ – функция отображения проектного решения на множество объектов класса «Атом»,

$Fedu: \{A\} \rightarrow \{D\}$ – функция построения упорядоченного множества учебных материалов для изучения определённых атомов знаний,

$Fdefine: P \rightarrow \{D\}$ – функция поиска учебных материалов, описывающих определённое понятие,

$Fsimilar: D \rightarrow \{D\}$ – функция поиска наиболее похожих учебных материалов,

$T: D \rightarrow \mathbb{Q}^+$ – дидактическая сложность материала.

Множество аксиом представлено в виде:

$Ax = \{AxAHP, AxAHD, AxPAfP, AxPAfA, AxAAfP\}$,
где $AxAHP$ – «атомы состоят в понятиях», если атом Y состоит в понятии X , которое состоит в понятии Z , то атом Y состоит в понятии Z в виде *SWRL*:

Понятие $(?x) \wedge$ Атом $(?y) \wedge$ Понятие $(?z) \wedge$ состоит_в $(?y, ?x) \wedge$ состоит_в $(?y, ?z) \rightarrow$ состоит_в $(?y, ?z)$.

$AxAHD$ – «атомы состоят в учебных материалах», если атом Y состоит в понятии X , которое изучается в учебном материале Z , то атом Y изучается в учебном материале Z в виде *SWRL*:

Понятие $(?x) \wedge$ Атом $(?y) \wedge$ УчебныйМатериал $(?z) \wedge$ состоит_в $(?y, ?x) \wedge$ изучается_в $(?y, ?z) \rightarrow$ изучается_в $(?y, ?z)$.

$AxPAfP$ – «атомы изучаются после атомов», если атом Y состоит в понятии X , которое изучается после понятия Z , а атом C состоит в Z , то атом Y изучается после C в виде *SWRL*:

Атом $(?y) \wedge$ Понятие $(?x) \wedge$ Понятие $(?z) \wedge$ Атом $(?c) \wedge$ состоит_в $(?y, ?x) \wedge$ состоит_в $(?c, ?z) \wedge$ изучается_после $(?x, ?z) \rightarrow$ изучается_после $(?y, ?c)$.

$AxPAfA$ – «понятия изучаются после атомов», если атом Y состоит в понятии X , которое изучается после атома C , то атом Y изучается после C в виде *SWRL*:

Атом $(?y) \wedge$ Понятие $(?x) \wedge$ Атом $(?c) \wedge$ состоит_в $(?y, ?x) \wedge$ изучается_после $(?x, ?c) \rightarrow$ изучается_после $(?y, ?c)$.

$AxAAfP$ – «атомы изучаются после понятий», если атом Y изучается после понятия X и атом C состоит в X , то атом Y изучается после C в виде *SWRL*:

Атом $(?y) \wedge$ Понятие $(?x) \wedge$ Атом $(?c) \wedge$ состоит_в $(?c, ?x) \wedge$ изучается_после $(?y, ?x) \rightarrow$ изучается_после $(?y, ?c)$.

Модель обучаемого инженера имеет вид:

$$U = (UA, P, C, P_{AB}, C_{AB}, P_S, C_S, ACT_G, ACT_C),$$

где $UA = \{a_i \in A | i = 1..nua\}$ – подмножество атомов знаний, которые необходимо изучить,

$P = A \rightarrow [0..1]$ – степень владения знаниями атома A ,

$C = A \rightarrow N$ – количество контрольных измерений знаний атома A ,

$P_{AB} = A \rightarrow [0..1]$ – степень владения умениями атома A ,

$C_{AB} = A \rightarrow N$ – количество контрольных измерений умений атома A ,

$P_S = A \rightarrow [0..∞]$ – степень владения навыками атома A ,

$C_S = A \rightarrow N$ – среднее время выполнения операций для атома A ,

$ACT_G = A \rightarrow N$ – суммарное количество действий при эталонном использовании навыка атома A ,

$ACT_C = A \rightarrow N$ – суммарное количество действий при текущем использовании навыка атома A .

Модель теста имеет вид:

$$TZ = (q, ANS, TA, TAQ),$$

где q – вопрос,

$ANS = \{ans_i | i = 1..n\}$ – множество ответов,

$TA = \{a_i \in A | i = 1..k\}$ – подмножество атомов знаний, которые оценивают тестовые задания,

$TAQ = \{q_i \in ANS | i = 1..m\}$ – множество правильных ответов.

Модель выполнения теста имеет вид:

$$UTZ = (U, TZ, QQ),$$

где U – профиль проектировщика,

TZ – тестовое задание,

$QQ = \{qq_i \in ANS | i = 1..k\}$ – подмножество ответов, выбранных обучаемым проектировщиком.

Модель сценария имеет вид:

$ST = (A, P, D, STZ, SPTZ, PERIOD, Fd_d, Fphase_tz, Fphase_ptz, Fphase_d, Fphase_first, Fnext_phase),$

где $A = \{a_i | i = 1..n\}$ – множество атомов знаний,

$P = \{p_i | i = 1..j\}$ – множество понятий,

$D = \{d_i | i = 1..k\}$ – множество учебных материалов,

$STZ = \{tz_i | i = 1..m\}$ – множество тестовых заданий,

$SPTZ = \{ptz_i | i = 1..l\}$ – множество практических заданий,

$PERIOD = \{pz_i | i = 1..h\}$ – множество этапов сценария,

$Fd_d = D \rightarrow D$ – функция следования учебных материалов,

$Fphase_d = PHASE \rightarrow D^*$ – функция принадлежности подмножеству учебных материалов определённому этапу сценария,

$Fphase_tz = PHASE \rightarrow TZ^*$ – функция принадлежности подмножества тестовых заданий определённому этапу сценария,

$Fphase_ptz = PHASE \rightarrow PTZ$ – функция принадлежности практического задания определённому этапу сценария,

$Fnext_phase = PHASE \rightarrow PHASE$ – функция следования этапов сценария,

$Fphase_first = PHASE \rightarrow D$ – функция первого учебного материала в определённом этапе сценария.

Модель операции имеет вид:

$$Operation = (id, type, number, pvo),$$

где id – уникальный идентификатор операции,

$type \in TypeOperation$ – тип операции,

$number$ – номер операции,

pvo – множество параметров операции со значением.

Модель параметра операции со значением имеет вид:

$$PVO = (key, value),$$

где $key \in ParamKey$ – название параметра,

$value \in ParamValue$ – значение параметра.

Модель исходных данных для формирования рекомендаций имеет вид:

$$S = (Operations, Rules, A, F_atom),$$

где $Operations = \{o \in Operation\}$ – множество проектных операций,

$Rules = \{r_i \mid i = 1..k\}$ – множество правил для поиска и замены неоптимальных проектных операций,

$A = \{a_i \mid i = 1..n\}$ – множество атомов знаний,

$F_atom = Operation \rightarrow A$ – функция отображения операции на атомы знаний.

Модель правила имеет вид:

$$Rule = (tpl, result),$$

где $tpl = \{t_i \mid i = 1..k\}$ – формула логики первого порядка для поиска в протоколе проектных операций;

$result = \{res_i \mid i = 1..n\}$, $res = (C, key, value)$ – множество оптимальных проектных операций, заданных как упорядоченные тройки (код операции, параметр операции или множество параметров операции, значение параметра операции или множество значений), где код – константа, а параметр операции и значение параметра операции – формулы логики первого порядка.

Для формулы логики первого порядка зададим алфавит символов:

- предметные переменные:

- $X = \{xp \in TypeOperation\}$ – множество операций,

- $P = \{p \in ParamKey\}$ – множество ключей параметров операций,

- $T = \{t \in ParamValue\}$ – множество значений параметров операций;

- символы логических операций: $\neg, \wedge, \vee, \rightarrow, \leftrightarrow$;

- кванторы: \forall, \exists ;

- вспомогательные символы: «()», «{}» – скобки;

«.,» – запятая;

- термы:

- $type = Operations \rightarrow TypeOperation$ – определение типа операции,

- $code = Operations \rightarrow \mathbb{N}$ – определение кода операции,

- $number = Operations \rightarrow \mathbb{N}$ – определение номера операции,

- $param = Operations \times ParamKey \rightarrow$

$ParamValue$ – определение значения параметра операции,

- $param_start = Operations \times ParamKey \rightarrow ParamValue$ – значение по умолчанию параметра операции,

- арифметические действия: $\mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$ – умножение, сложение, вычитание, деление;

- предикаты:

2-арные функции: больше ($>$), меньше ($<$), равно ($=$), не равно (\neq), \in , eq_op – равенство результатов операций.

Модель рекомендации имеет вид:

$$R = (op_before, op_after, F_steps, F_message),$$

где $op_before \subset Operations$ – множество неоптимальных проектных операций, задействованных при формировании рекомендации,

$op_after \subset Operations$ – множество рекомендуемых проектных операций,

$F_steps = Operations \rightarrow \mathbb{N}$ – функция вычисления количества действий для построения проектных операций,

$F_message = Operations \times Operations \rightarrow Text$ – функция формирования текста рекомендации.

4 МЕТОД АДАПТАЦИИ ОБУЧАЕМОГО ИНЖЕНЕРА

Алгоритм генерации индивидуального сценария обучаемого подробно приведён в [12]. Ниже на содержательном уровне описан указанный алгоритм.

1. Выбор сценария обучения.

2. Загрузка следующего этапа выбранного сценария. При отсутствии предыдущего этапа активируется первый этап сценария.

3. Обучаемый последовательно изучает учебный материал в рамках этапа, в том числе дополнительные информационные активы промышленного предприятия.

4. Выполнение тестовых заданий, которые корректирует уровень знаний обучаемого инженера.

5. Проверка уровня умений на основе проектных заданий.

6. Оценка уровня знаний, умений, если они считаются удовлетворительными, обучаемый инженер переходит на следующий этап сценария, на второй шаг. При его отсутствии процесс обучения считается завершённым. В случае неудовлетворительного уровня знаний, умений на седьмом шаге синтезируется индивидуальная траектория обучения.

7. Генерация адаптивной траектории обучения. Выбираются элементы знаний, которые были изучены на данном этапе, но их уровень знаний, умений меньше 0,5.

8. Наполнение множества учебных материалов, которые имеют связь «часть-целое» с выбранными элементами знаний. Учебные материалы выбираются с целью минимизации суммарного времени обучения и количества повторений уже представленного материала. Также множество учебных материалов упорядочивается в соответствии с отношением порядка элементов знаний.

9. Формирование минимального множества тестовых и проектных заданий, проверяющих выбранное множество элементов знаний.

10. После прохождения индивидуального этапа обучения проводится контрольное тестирование, если и в этом случае характеристики обучаемого инженера не соответствуют ожидаемым, то строится повторная траектория, отличием которой является тот факт, что множество учебных материалов подбирается с наименьшим повторением относительного предыдущей траектории.

Рассмотрим алгоритм формирования рекомендаций.

1. Начало работы обучаемого с проектом.
2. Если проект новый, то переход к шагу 5.
3. Генерация операций на основе имеющегося проекта.
4. Добавление операции в последовательность операций.
5. Считывание управляющего воздействия от обучаемого.
6. Генерация операций на основе действия обучаемого.
7. Добавление операции в последовательность операций.
8. Формирование состояния проекта на основе последовательности операций.
9. Добавление состояния проекта в последовательность состояний проекта.
10. Поиск правила, которое соответствует последовательности операций.
11. Если правило не найдено, то переход к шагу 13.
12. Добавление рекомендации в индивидуальный список обучаемого и вывод на экран.
13. Поиск правила, которое соответствует последовательности состояний.
14. Если правило не найдено, то переход к шагу 16.
15. Добавление рекомендации в индивидуальный список обучаемого и вывод на экран.
16. Если работа с проектом не закончена, то переход к шагу 5.
17. Выход.

Приведем алгоритм корректировки профиля обучаемого.

1. Получить сформированную рекомендацию на основе профиля обучаемого.
2. Составить список проектных операций, задействованных в правилах при формировании рекомендаций.
3. Получить список элементов знаний, ассоциированных с проектными операциями.
4. Уменьшить уровень владения навыками для данных элементов знаний.

5 РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТОВ

Проведён эксперимент по обучению среди студентов Ульяновского государственного технического университета численностью 16 человек. Для оценки начального уровня навыков был проведён предтест, по результатам которого сформированы две группы из 8 и 10 человек, со средним уровнем навыков 0,5. Группы обучались по материалам курса «Азбука КОМПАС». Первая группа про-

ходила линейный сценарий обучения, вторая – адаптивный. После обучения стояла задача построить сборку «Крышка насоса» состоящую из 471 элемента. Результаты эксперимента приведены в таблице. При использовании адаптивного сценария обучения средний уровень навыков студентов выше на 20% по сравнению с линейным за счёт более детального обучения, что повлекло увеличение времени подготовки на 17%.

Таблица
Сравнение линейного и адаптивного сценариев обучения

	Средний уровень навыков до обучения	Среднее время обучения, ч	Средний уровень навыков после обучения	Среднее время построения сборки, ч	Общее затраченное время, ч
Линейный сценарий	0,5	1,65	0,745	2,8	4,5
Адаптивный сценарий	0,5	2	0,9	1,9	3,9
Выигрыш		-17%	20%	50%	20%

Система рекомендаций положительно сказывается на скорости выполнения проектных решений за счёт уменьшения дублирующих операций, ошибочных и лишних действий. При следовании рекомендациям, направленным на исключение ошибочных и лишних действий, уменьшается количество действий при выполнении проектной операции на 1, в среднем это даёт прирост 25% на одну рекомендацию. Исключение дублирующих проектных операций уменьшает количество действий от 30% в зависимости от количества продублированных операций.

РЕАЛИЗАЦИЯ

Блок обучения построен по классической трехзвенной архитектуре: клиент, сервер приложений и сервер баз данных.

В качестве платформы разработки использована технология Java Platform, Standard Edition, поддерживающая встроенные средства клиент-серверных приложений (технологии RMI и SOAP).

В качестве сервера баз данных используется MySQL. Система является масштабируемой, позволяет расширять функциональность с помощью добавления компонентов (плагинов). Плагины позволяют расширять как клиентскую, так и серверную части с помощью предоставления клиентского и серверного API.

Для реализации клиент-серверной технологии на Java Platform, Standard Edition выбрана технология веб-сервисов. Протокол SOAP используется для обмена сообщениями между веб-службами.

Блок рекомендаций построен на платформе .NET

Framework. В качестве примера выбрана система автоматизированного проектирования КОМПАС-3D V16. Для обработки событий используется технология Automation, реализованная в виде библиотеки на языке программирования C#.

ЗАКЛЮЧЕНИЕ

Разработана АОС, использующая новые онтологическую модель предметной области и метод формирования индивидуального сценария обучения. Онтологическая модель предметной области машиностроительного проектирования отличается использованием иерархических, зависимых, определяющих, последовательных и альтернативных связей. Метод формирования индивидуального сценария обучения отличается использованием динамических механизмов взаимодействия моделей пространства обучения (обучаемого, теста, выполнения теста, практического задания, решения практического задания, сценария) с онтологической моделью предметной области и обеспечивает формирование динамической траектории обучения и сокращение времени обучения.

С учётом предложенных моделей разработан метод формирования рекомендаций и корректировки профиля обучаемого на основе деятельности проектировщика в процессе выполнения рабочих задач, позволяющий повысить эффективность практической подготовки.

Основным количественным позитивом исследования является увеличение производительности труда проектировщиков в среднем на 20–25%.

СПИСОК ЛИТЕРАТУРЫ

1. Шабалина О.А. Модель процесса обучения и ее интерпретация в обучающей компьютерной игре // Вестник СГТУ. – 2013. – № 2 (70), вып. 1. – С. 158–167.
2. Кудрявцев В.Б., Алисейчик П.А., Вашик К. Моделирование процесса обучения // Фундаментальная и прикладная математика. – 2009. – Т. 15, № 05. – С. 111–169.
3. Касьянова Е.В. WAPE – адаптивная система поддержки дистанционного обучения программированию // Тр. Междунар. конф. «Вычислительные и информационные технологии в науке и образовании». – Павлодар, 2006. – Т. 1. – С. 606–615.
4. Доррер А.Г., Иванилова Т.Н. Моделирование интерактивного адаптивного обучающего курса // Современные проблемы науки и образования. – 2007. – № 5. – С. 52–59.
5. Huang M.J., Huang H.S., Chen M.Y. Constructing a personalized e-learning system based on genetic algorithm and case-based reasoning approach. *Expert Systems with Applications* 33, 551–564 (2007).
6. Khribi, M. K., Jemni, M., & Nasraoui, O. Automatic Recommendations for E-Learning Personalization Based on Web Usage Mining Techniques and Information Retrieval. *Educational Technology & Society*, 12 (4), 30–42 (2009).
7. Esichaikul V., Lamnoi S. and Bechter C. "Student Modelling in Adaptive E-Learning Systems," *Knowledge Management & E-Learning: An International Journal*, vol. 3, pp. 342–355, 2011.
8. Yasir Eltigani Ali Mustafa and Sami Mohamed Sharif. "An approach to Adaptive E-learning Hypermedia System based on Learning Styles (AEHS-LS): Implementation and evaluation," *International Journal of Library and Information Science*, Vol. 3, No. 1, 2011, pp. 15–28.
9. Mahnane L., Tayeb Laskri M. and Trigano P. "A Model of Adaptive e-learning Hypermedia System based on Thinking and Learning Styles", *IJMUE*, vol. 8, no. 3 (2013), pp. 339–350.
10. Ploum E.: *Authoring of adaptation in the GRAPPLE project*. Master Thesis, Eindhoven University of Technology (2009).
11. De Bra P., Smits D., van der Sluijs K., Cristea A.I. and Hendrix M. "GRAPPLE: Personalization and adaptation in learning management systems," in *Proc. World Conf. edmedia*, 2010, pp. 3029–3038.
12. Афанасьев А.Н., Войт Н.Н., Канев Д.С. Математическое моделирование процесса инженерного обучения в сложных инфокоммуникационных образовательных системах // *Радиотехника*. – 2014. – № 7. – С. 133–136.

REFERENCES

1. Shabalina O.A. Model protsessa obuchenii i ee interpretatsiia v obuchaiushchei kompiuternoii igre [Educational Games Development: Learning Process Model and Its Integration into the Game Context]. *Vestnik SGTU* [Bull. of Saratov State Technical University], 2013, no. 2 (70), iss. 1, pp. 158–167.
2. Kudriavtsev V.B., Aliseichik P.A., Vashik K. Modelirovanie protsessa obuchenii [Modelling Learning Process]. *Fundamentalnaia i prikladnaia matematika* [Fundamental and Applied Mathematics], 2009, vol. 15, no. 05, pp. 111–169.
3. Kasianova E.B. WAPE – adaptivnaia sistema podderzhki distantsionnogo obuchenii programirovaniu [WAPE is an Adaptive System for Distance Learning of Programming]. *Tr. Mezhdunar. konf. "Vychislitelnye i informatsionnye tekhnologii v nauke i obrazovanii"* [Proc. of Int. Sci. Conf. "Computer and Information Technologies in Science and Education"]. Pavlodar, 2006, vol. 1, pp. 606–615.
4. Dorrer A.G., Ivanilova T.N. Modelirovanie interaktivnogo adaptivnogo obuchaiushchego kursa [The Modelling of Interactive Adaptive Learning Course]. *Sovremennye problemy nauki i obrazovaniia* [Modern Problems of Science and Education], 2007, no. 5, pp. 52–59.
5. Huang M.J., Huang H.S., Chen M.Y. Constructing a Personalized E-Learning System Based on Genetic Algorithm and Case-Based Reasoning Approach. *Expert Systems with Applications*, 2007, no. 33, pp. 551–564.
6. Khribi, M. K., Jemni, M., & Nasraoui, O. Automatic Recommendations for E-Learning Personalization Based on Web Usage Mining Techniques and Information Retrieval. *Educational Technology and Society*, 2009, no.12 (4), pp. 30–42.

7. Esichaikul V., Lamnoi S. and Bechter C. Student Modelling in Adaptive E-Learning Systems. *Knowledge Management & E-Learning: An International Journal*, 2011, vol. 3, pp. 342–355.

8. Yasir Eltigani Ali Mustafa and Sami Mohamed Sharif. An Approach to Adaptive E-Learning Hypermedia System based on Learning Styles (AEHS-LS): Implementation and Evaluation. *International Journal of Library and Information Science*, 2011, vol. 3, no. 1, pp. 15–28.

9. Mahnane L., Tayeb Laskri M., and Trigano P. A Model of Adaptive E-Learning Hypermedia System based on Thinking and Learning Styles. *IJMUE*, 2013, vol. 8, no. 3, pp. 339–350.

10. Ploum E. Authoring of Adaptation in the GRAPPLE Project. *Master Thesis of Eindhoven University of Technology*, 2009.

11. De Bra P., Smits D., van der Sluijs K., Cristea A.I. and Hendrix M. GRAPPLE: Personalization and Adaptation in Learning Management Systems. *Proc. World Conf. Edmedia*. 2010, pp. 3029–3038.

12. Afanasiev A.H., Voit N.N., Kanev D.S. Matematicheskoe modelirovanie protsessa inzhenerenogo obucheniiia v slozhnykh infokommunikatsionnykh obrazovatelnykh sistemakh [Mathematical Modelling of Engineering Education Process in Complex Infocommunication Learning Systems]. *Radiotekhnika* [Radioengineering], 2014, no. 7, pp. 133–136.