

DOI: 10.32517/0234-0453-2024-39-3-53-65

ФОРМИРОВАНИЕ ИНДИВИДУАЛЬНОЙ ОБРАЗОВАТЕЛЬНОЙ ТРАЕКТОРИИ СТУДЕНТА ВУЗА НА ОСНОВЕ МОДИФИЦИРОВАННОГО МУРАВЬИНОГО АЛГОРИТМА

К. В. Захарченков¹, Т. В. Мрочек¹ ✉, В. В. Кутузов¹¹ Белорусско-Российский университет, г. Могилев, Беларусь

✉ mrovlad@mail.ru

Аннотация

Индивидуальная образовательная траектория позволяет предоставить каждому обучающемуся последовательность учебных материалов в соответствии с его темпом, стилем обучения и имеющимися пробелами в познаниях. В условиях цифровизации образования появляется возможность использовать современные способы повышения качества обучения для создания индивидуальной образовательной траектории. В статье представлена модель адаптации учебного контента, которая содержит алгоритм построения индивидуальной образовательной траектории, основанный на модифицированном муравьином алгоритме (алгоритме муравьиной колонии) и обеспечивающий персонализированный подбор последовательности учебных материалов. Модель адаптации учебного контента рассматривается на примере изучения языка SQL в рамках курса «Базы данных».

Предложенная модификация алгоритма муравьиной колонии позволяет повысить эффективность формирования компетенций при прохождении учебных дисциплин, имеющих значительную продолжительность и состоящих из слабо связанных между собой модулей, за счет отнесения каждого студента к наиболее подходящей подгруппе в соответствии с выбранной профессиональной ролью и организации полноценного повторения на протяжении изучения всей дисциплины. В качестве спецификаторов групп обучающихся предложено использовать желаемый квалификационный уровень освоения языка SQL (базовый или профессиональный) или направление подготовки (специальность). Повторение основ решения задач в процессе изучения дисциплины предложено выполнять на основе определяемых алгоритмом муравьиной колонии значений «феромона», формула для расчета которых обеспечивает высокую вероятность перенаправления студента на решение задач из списка на повторение при превышении допустимого количества взятых подсказок или попыток решения.

Ключевые слова: алгоритмы роевого интеллекта, муравьиный алгоритм, алгоритм муравьиной колонии, профессиональная подготовка по SQL, компетенции, индивидуальная образовательная траектория, учет забывания, модель обучающегося, модель адаптации учебного контента.

Для цитирования:

Захарченков К. В., Мрочек Т. В., Кутузов В. В. Формирование индивидуальной образовательной траектории студента вуза на основе модифицированного муравьиного алгоритма. *Информатика и образование*. 2024;39(3):53–65. DOI: 10.32517/0234-0453-2024-39-3-53-65.

FORMATION OF INDIVIDUAL EDUCATIONAL TRAJECTORY OF UNIVERSITY STUDENT BASED ON MODIFIED ANT COLONY OPTIMIZATION ALGORITHM

K. V. Zakharchenkov¹, T. V. Mrochek¹ ✉, V. V. Kutuzov¹¹ Belarusian-Russian University, Mogilev, Belarus

✉ mrovlad@mail.ru

Abstract

Individual educational trajectory allows to provide each learner with a sequence of learning materials in accordance with their pace, learning style and existing gaps in knowledge. Under the conditions of education digitalization, there is an opportunity to use modern ways to improve learning quality to create an individual educational trajectory. The article presents a model of learning content adaptation which contains an algorithm for building an individual educational trajectory based on a modified ant colony optimization algorithm that provides a personalized selection of a sequence of educational materials. The model of learning content adaptation is illustrated on the example of learning SQL within the framework of the course “Databases”.

The proposed modification of the ant colony optimization algorithm allows to increase in the efficiency of competence formation in the course of academic disciplines which have a significant duration and consist of loosely related modules by assigning each student to the most appropriate subgroup in accordance with the chosen professional role and organizing full-fledged repetition throughout the

discipline. It is proposed to use the desired qualification level of mastering the SQL (basic or professional) or the direction of training (specialty) as specifiers of student groups. The repetition of the basics of problem solving in the process of studying the discipline is proposed to be performed based on pheromone values determined by the ant colony optimization algorithm. The calculating formula for these values provides a high probability of redirecting the student to the solution of problems from the list for repetition when exceeding the permissible number of taken hints or attempts to solve.

Keywords: swarm intelligence algorithms, ant colony optimization algorithm, ACO algorithm, professional SQL training, competencies, individual educational trajectory, accounting for forgetting, learner model, training content adaptation model.

For citation:

Zakharchenkov K. V., Mrochek T. V., Kutuzov V. V. Formation of individual educational trajectory of university student based on modified ant colony optimization algorithm. *Informatics and Education*. 2024;39(3):53–65. (In Russian.) DOI: 10.32517/0234-0453-2024-39-3-53-65.

1. Введение

Построение индивидуальных образовательных траекторий (ИОТ) обучающихся является одной из целей цифровой трансформации отрасли науки и высшего образования в Российской Федерации к 2030 году¹. Создание условий для выстраивания педагогами ИОТ является одним из направлений развития системы образования Беларуси до 2025 года². Именно ИОТ позволяют сформировать для каждого обучающегося последовательность учебных материалов, подбираемую индивидуально и адаптивно [1, 2], т. е. с постоянной автоматизированной корректировкой с учетом пробелов в знаниях, темпа и стиля обучения, особенностей забывания ранее приобретенных знаний.

Однако в традиционном учебном процессе при изучении различных разделов информатики (как в средних специальных и высших учебных заведениях, так и в подавляющем большинстве массовых открытых онлайн-курсов (МООК)) чаще всего практикуется последовательное, линейное прохождение тем дисциплины. Материал может преподаваться и по спиралевидной схеме, т. е. в каждой последующей теме элементы изученного ранее материала излагаются на более высоком уровне [3, 4]. Как правило, преподаватели используют знакомые, привычные формы обучения (например, традиционные лекции, занятия с использованием мультимедиа и ЭВМ) и полагаются на свой опыт обнаружения пробелов в познаниях студента. При таком подходе отсутствует адаптивный учет индивидуальных особенностей восприятия и забывания материала, а значит, современные возможности повышения качества обучения не используются в полной мере.

Например, при самоподготовке студенты, изучающие программирование, могут использовать российские интернет-тренажеры:

- для отработки навыков программирования на различных процедурных языках — ряд

курсов на Каталог-курсов.ру, Codeforces, Codechick, Stepik и др.³;

- для обучения написанию SQL-запросов — «Упражнения по SQL», LearnDB, SQLAcademy⁴.

Однако данные интернет-ресурсы трудно применять в традиционном учебном процессе, т. к. они не обеспечивают преподавателю возможность в полной мере контролировать процесс обучения и приспособить задания к учебному плану дисциплины. Обратная связь у подобных ресурсов чаще всего ограничивается выдачей сообщения о правильном/неправильном ответе, а персонализированный подход к организации обучения по ИОТ отсутствует.

Задача формирования ИОТ является NP-трудной⁵ [1, 5], поэтому для ее решения применяются различные алгоритмы искусственного интеллекта [1, 6], например, биоинспирированные алгоритмы роевого интеллекта. Действия обучающихся при построении ИОТ, обеспечивающей достижение наилучших результатов обучения за минимальное время, напоминают действия членов (агентов) роевых популяций при поиске источников пищи. Поэтому муравьиный алгоритм (или алгоритм муравьиной колонии (англ. Ant Colony Optimization (ACO) Algorithm) является одним из наиболее часто используемых [1, 7]. Муравьиный алгоритм имеет небольшое количество параметров по сравнению с иными алгоритмами искусственного интеллекта и, как отмечается в работе А. Р. Dharshini, S. Chandrakumarmangalam, G. Arthi «Ant colony optimization for competency based learning objects sequencing in e-learning» [8], в задаче построения ИОТ чаще показывает лучшие результаты, нежели, например, генетический алгоритм (англ. Genetic Algorithm) или алгоритм оптимизации роем частиц (англ. Particle Swarm Optimization, PSO).

Целью статьи является построение ИОТ с помощью алгоритма муравьиной колонии, модифициро-

¹ Стратегия цифровой трансформации отрасли науки и высшего образования. М.: Минобрнауки России; 2021. 267 с. <https://minobrnauki.gov.ru/upload/iblock/e16/dv6edzmr0og5dm57dtm0wyllr6uwttjw.pdf>

² Постановление Совета Министров Республики Беларусь от 30 ноября 2021 года № 683 «О Концепции развития системы образования Республики Беларусь до 2030 года». <https://adu.by/images/2021/12/konsep-razv-sist-obrazov.pdf>

³ Подробнее см. соответствующие сайты: <https://katalog-kursov.ru>, <https://codeforces.com/>, <https://codechick.io/>, <https://stepik.org/catalog>.

⁴ Подробнее см. соответствующие сайты: <https://sql-ex.ru/>, <https://learndb.ru/>, <https://sql-academy.org/ru>.

⁵ Задача относится к классу NP-трудных (англ. Nondeterministic Polynomial, NP — недетерминированный полиномиально-временной), если ее невозможно решить никаким полиномиально-временным алгоритмом при допущении истинности неравенства $P \neq NP$, где P — класс детерминированных полиномиально-временных задач. Подробнее см.: Рафгарден Т. Совершенный алгоритм. Алгоритмы для NP-трудных задач. Санкт-Петербург: Питер; 2021. 304 с.

ванного за счет разделения студентов на группы в зависимости от выбранного направления подготовки и желаемого квалификационного уровня подготовки, а также подбора параметров и формул алгоритма таким образом, чтобы обеспечить высокую вероятность выбора тех заданий, основы решения которых были не полностью усвоены.

Данный алгоритм предназначен в первую очередь для построения ИОТ при изучении учебных дисциплин, имеющих значительную протяженность во времени и состоящих из слабо связанных между собой модулей. В качестве примера подобной дисциплины можно назвать курс «Базы данных», важным компонентом которого является язык SQL (аббревиатура от *англ.* Structured Query Language — «язык структурированных запросов»). SQL очень востребован работодателями в промышленном программировании и включен во все курсы изучения баз данных [9]. Реляционные системы управления базами данных (СУБД), основанные на языке SQL, много лет стабильно занимают первые позиции по популярности среди СУБД различных типов. Но, как правило, в курсе по базам данных на изучение SQL уходит только часть времени (в университете, например, не больше трех семестров, т. е. 40–65 % аудиторных часов дисциплины).

Как показывает практика, использование принципа итеративного обучения (многократного повторения информации на более высоком уровне в последующих темах) не является эффективным, поскольку повторяются чаще всего лишь отдельные фрагменты изученных тем без учета информации о том, что именно было забыто конкретным обучающимся. Поэтому *организация решения достаточного количества заданий, равномерно распределенного на протяжении всего курса, с повторением пройденных типовых задач, последовательность которых формируется для каждого студента индивидуально, является принципиально важной.*

К тому же изучение SQL осложняется тем, что этот язык является декларативным. При переходе на SQL с более привычных процедурных языков программирования возникают трудности в силу, например, необходимости наработки понимания особенностей работы с множествами и отношениями, ограничений возможности использования привычных циклических конструкций и т. п. [10, 11].

Для достижения поставленной нами цели необходимо выполнить следующие шаги:

- проанализировать существующие модификации алгоритма муравьиной колонии для задачи построения ИОТ в электронных обучающих системах;
- разработать модификацию указанного алгоритма;
- оценить повышение эффективности с точки зрения сформированности компетенций на примере курса «Базы данных» при апробации предложенной модификации.

2. Обзор подходов к формированию индивидуальной образовательной траектории на основе муравьиного алгоритма

Муравьиный алгоритм описан в статье M. Dorigo, L. M. Gambardella «Ant colonies for the travelling salesman problem» [12] для решения комбинаторных задач и модифицирован в ряде работ специально для задачи формирования ИОТ [5–8, 13–18]. Этот алгоритм описывает разнесенное во времени не прямое взаимодействие муравьев на пути от колонии к источнику еды: наиболее удачные и короткие пути муравьи маркируют большим количеством феромона, по которому новые особи ориентируются, выбирая траектории с наибольшей концентрацией феромонов, позволяющие добраться до еды за наименьшее время. Каждый раз муравьи оставляют феромонные метки, увеличивая концентрацию феромонов на оптимальном пути. А на более долгих путях, выбранных первыми муравьями, феромоны со временем испаряются, и менее используемые траектории исчезают.

В различных модификациях применения муравьиного алгоритма к образовательным задачам совокупность учебных материалов представлена графом, узлы которого представляют собой обучающие объекты (элементы теории, тесты, задачи (типичные и повышенной сложности), контрольные работы) и связаны между собой ребрами — ссылками на обучающие объекты. Студент k (аналог муравья) перемещается из узла i в узел j , имеющий наибольшую вероятность перехода $P_{ij,k}(t)$. В случае положительного прохождения узла студент «откладывает» на ребре r_{ij} «феромон» τ_{ij} , задействуя тем самым канал стигмергии¹, с помощью которого организуется разнесенное во времени не прямое взаимодействие обучающихся. Другие студенты используют «феромонную» информацию, выбирая наиболее посещаемые ребра — обучающие объекты, позволяющие успешно пройти курс за наименьшее время.

В адаптивной персонализированной системе электронного обучения, представленной в статье D. Sivakumar, R. Praveena «Determining optimized learning path for an e-learning system using ant colony optimization algorithm» [13], путь определяется интенсивностью меток на ребрах графа (т. е. максимальной частотой посещений узла графа, характеризующей заинтересованность студентов), а частота посещения узла рассчитывается как отношение числа посещений узла к общему количеству узлов графа. Такой подход к определению траектории не позволяет учесть индивидуальные особенности каждого студента при восприятии и забывании материала.

¹ Стигмергия (*др.-греч.* στίγμα — знак, метка, ἔργον — действие, работа) — асинхронное взаимодействие между субъектами, в ходе которого одни вносят изменения в окружающую среду, а другие используют эти изменения позднее. Стигмергия у муравьев реализуется с помощью феромонных меток.

В работе Ю. Ю. Дюличевой «Алгоритмы роевого интеллекта и их применение для анализа образовательных данных» [14] была описана модификация муравьиного алгоритма для задачи организации оптимальной последовательности тем проектных семинаров, при этом рекомендуемые вершины графа тем определялись на основе траекторий студентов, схожих по уровню знаний, без учета личных особенностей забывания.

«Феромон», как и в статье L.-H. Wong, C.-K. Looi «Adaptable learning pathway generation with ant colony optimization» [15], представляет собой меру сходства между атрибутами c «муравья» k и «муравьев» a , прошедших курс:

$$S(c, a) = \sqrt{\sum w_x (x_c - x_a)^2 + w_t (t_c - t_a)^2},$$

где:

x — количественное значение атрибута «муравья»;

w_x — весовой коэффициент, характеризующий значимость навыков, приобретаемых при прохождении узла;

w_t — весовой коэффициент, характеризующий время обучения, имитируя испарение «феромонов» на ребре.

S. Allach, M. B. Ahmed, A. Ghadi, M. Essaaidi в статье «Modeling of e-learning based on ant colony algorithm» [16] предлагают пометить ребро r_{ij} графа «феромоном» успеха S или неудачи F , добавляемым при верном (или неверном) решении задания в узле j . Значение S или F рассчитывается на основе механизма обратного распространения: S или F (первоначально равно оценке α за прохождение узла j) должно появиться не только на ребре, которое привело муравья к узлу j , но и на предыдущих ребрах с номерами m с уменьшением значения S или F на каждом предшествующем ребре на величину $\alpha/(m+1)$.

N. C. Benabdellah, M. Gharbi, M. Bellafkih в работе «Ant colony algorithm and new pheromone to adapt units sequence to learners' profiles» [17] предложили использовать «феромон» следования, который в случае, если посещенный узел j принадлежит корректной последовательности узлов, заданной преподавателем на основе учебной программы курса, добавляет на ребро r_{ij} положительное значение S или F , а в противном случае — отрицательное значение.

Использование механизма обратного распространения «феромонов» S и F в работах [16, 17] позволяет увеличить значимость узлов, прохождение которых способствует более успешному решению задач в последующих узлах [16]. Однако такой подход неэффективен при изучении ряда слабо связанных между собой тем в типовом учебном курсе по базам данных.

В случае, если студент не решил упражнение, I. Kamsa, R. Elouahbi, F. El Khoukhi в статье «The combination between the individual factors and the collective experience for ultimate optimization learning path using ant colony algorithm» [18] предлагают размещать на ребрах перед узлами, прохождение которых позволяет натренировать навыки решения, спасательный «феромон»:

$$Qsarc_{j,p} = 0,4 + (1 - Qsarc_{ij})0,4,$$

где:

j — набор ребер, не пройденных обучающимся и имеющих тот же узел-предшественник, что и пройденное ребро p ;

$Qsarc_{ij}$ — значение «феромона» на ребре r_{ij} , ведущем к узлу, в котором студент не решил задание.

Для сложных задач, содержащих множество различных понятий, такой подход не позволяет четко определить узел, помогающий прийти к верному решению в последующем узле.

В упомянутом выше исследовании A. P. Dharshini, S. Chandrakumarmangalam, G. Arthi [8] предложено формировать подгруппы на основе параметров, выбираемых студентами: желаемого уровня сложности курса (легкий, средний, сложный), вида учебной деятельности (тренировочный, оценка), когнитивного уровня прохождения курса (когнитивные уровни определены на основе таксономии Б. Блума¹). «Феромон» обновляется по формуле:

$$\tau_{ij} = (1 - \rho)\tau_{ij} + \frac{\rho}{M} \sum_{k \in M} F(k),$$

где:

ρ — скорость испарения «феромона»; $\rho = \overline{0,1}$;

M — набор решений, полученных на текущем шаге;

$F(k)$ — функция качества текущего решения k .

Однако формулирование в явном виде функции качества для распределенной во времени и многоаспектной задачи изучения курса «Базы данных» является трудоемким и сложным [19, 20].

В алгоритме ISACS, использование которого описывают M. Rastegarmoghadam, K. Ziarati в статье «Improved modeling of intelligent tutoring systems using ant colony optimization» [5], «феромон» разрешено добавлять только «муравьям» с наилучшими результатами за определенный период времени, что позволяет сократить работы по разведыванию и ускорить нахождение лучшего пути [21]. Изменение «феромона» описывается формулой:

$$\Delta\tau_{ij,k} = R_{ij,k} + \frac{S_{ij,k}}{S_{\max}/T_{ij,k}},$$

где:

$R_{ij,k}$ — коэффициент памяти;

$S_{ij,k}$ — оценка, полученная «муравьем» k за прохождение узла j ;

S_{\max} — наивысшая оценка среди всех оценок, полученных «муравьями», прошедшими ребро r_{ij} ;

$T_{ij,k}$ — время выполнения задания в узле j .

Забывание пройденного материала учитывает предложенный на основе кривой забывания Г. Эббингауза коэффициент памяти:

¹ Таксономия Блума (англ. Bloom's Taxonomy) — классификация целей обучения и связанных с ними когнитивных процессов, предложенная Б. Блумом в 1956 году и ставшая фундаментальным педагогическим инструментом для формирования образовательных целей и оценки учебных достижений обучающихся.

$$R_{ij,k} = \frac{1}{1 + \frac{1}{\lambda s} \ln(1 + \lambda t_{ij,k})},$$

где:

- λ — константа обучения;
- s — относительная мощность памяти;
- $t_{ij,k}$ — время перехода из узла i в j .

Каждому студенту предлагаются обучающие объекты в соответствии с его стилем обучения, определяемым на основе модели VARK¹, и индикатором типа личности Майерс—Бриггс². Однако такой подход может привести к ограничению возможностей для обучения, т. к. он не учитывает, что студенты могут использовать комбинации разных типов материалов.

Q. Liu, J. Huang, L. Wu, J. Hu, M. Hu в статье «Study on learner model in adaptive learning system based on ant colony algorithm» [22] предлагают подбирать обучающие объекты на основе шкалы стилей обучения Grasha—Riechmann³ и определять значение важности обучающего объекта на основе таблицы преобразования заданий, соответствующих разным уровням таксономии Б. Блума, в количественный коэффициент когнитивных целей φ_j , который может принимать значения от 0,5 до 3. Добавляемое количество «феромона» рассчитывается по формуле:

$$\tau_{ij,k} = R\tau_{ij,k} + \Delta\tau_{ij,k}, \quad (1)$$

где:

R — коэффициент сохранения в памяти ($0 < R \leq 1$), характеризующий объем остаточных знаний в момент времени t . R рассчитывается на основе выражения $R = e^{-tS}$, полученного с учетом исследований Г. Эббингауза [23], а S — характеристика забывания, индивидуальная для каждого обучающегося;

$\Delta\tau_{ij,k}$ — приращение «феромона», которое рассчитывается по формуле:

$$\Delta\tau_{ij,k} = \frac{\varphi_j}{\sum_{h=1}^j \varphi_h}, \quad (2)$$

где:

h — номер узла из списка узлов, переход в которые из предыдущего узла i допускается.

Таким образом, в известных на сегодняшний день модификациях муравьиного алгоритма для

задачи построения ИОТ используются различные подходы, касающиеся в основном того, как при расчете «феромонов» учитывать учебные достижения обучающегося, желаемый уровень сложности курса и забывание изученного материала. Однако эти модификации имеют следующие недостатки:

- общий характер и плохая приспособленность для учебных дисциплин со значительной расщепленностью во времени при изучении модулей, которые часто слабо связаны между собой, что требует полноценного повторения на протяжении всего курса;
- ориентация на пользователей, знакомых с каким-либо языком программирования, и формирование для них перечня задач, составленных, как правило, на основе личного опыта разработчика учебного курса или на основе таксономии Б. Блума. Такие подходы являются несовершенными, поскольку в отношении профессиональной подготовки в учебном заведении целесообразно указывать конкретное направление (специальность) с четко определенным перечнем компетенций или желаемый профессиональный уровень подготовки (например, «разработчик баз данных», «программист» и т. п.);
- попытки учесть при добавлении «феромонов» коэффициент забывания, определяемый с помощью функций на основе исследований кривой забывания Г. Эббингауза (см., например, в статьях [5, 22]), не могут представлять собой общее правило запоминания и забывания учебного материала [24].

3. Материалы и методы исследования

В качестве материала исследования используются типовые учебные программы по дисциплине «Базы данных» высших и средних специальных учебных заведений, предусматривающие изучение SQL. При разработке структуры графа электронных учебных материалов использована стратегия микрообучения, позволившая структурировать учебный материал небольшими порциями [25], количество которых зависит от объема изучаемой темы и требуемого уровня ее освоения. При построении модели адаптации учебного материала использован математический аппарат алгоритмов роевого интеллекта. Эмпирической базой исследования послужили результаты подготовки обучающихся с применением разработанной модификации алгоритма муравьиной колонии и без нее.

4. Характеристика базовых компонентов решения задачи построения индивидуальной образовательной траектории

Индивидуальная образовательная траектория, представленная в данном исследовании, выстраивается посредством взаимодействия модели предметной

¹ Модель VARK (аббревиатура, *англ.* Visual, Auditory, Reading и Kinesthetic — визуальный, аудиальный, читающий, кинестетический) — модель стилей обучения, разработанная Нилом Флемингом, построена на классификации учащихся по четырем группам в зависимости от их стиля восприятия информации.

² Индикатор типов Майерс—Бриггс (*англ.* Myers—Briggs Type Indicator, MBTI) — система психологического тестирования, которая проводится по методике самоотчета и помогает определить тип личности на основе индивидуальных предпочтений.

³ Шкала стилей обучения Grasha—Riechmann (*англ.* Grasha—Riechmann Learning Style Scales) — модель, разработанная Antony F. Grasha и Sheryl W. Riechmann в 1974 году, классифицирует стили обучения на шесть типов: независимый, зависимый, избегающий, активный, соревновательный, коллаборативный.

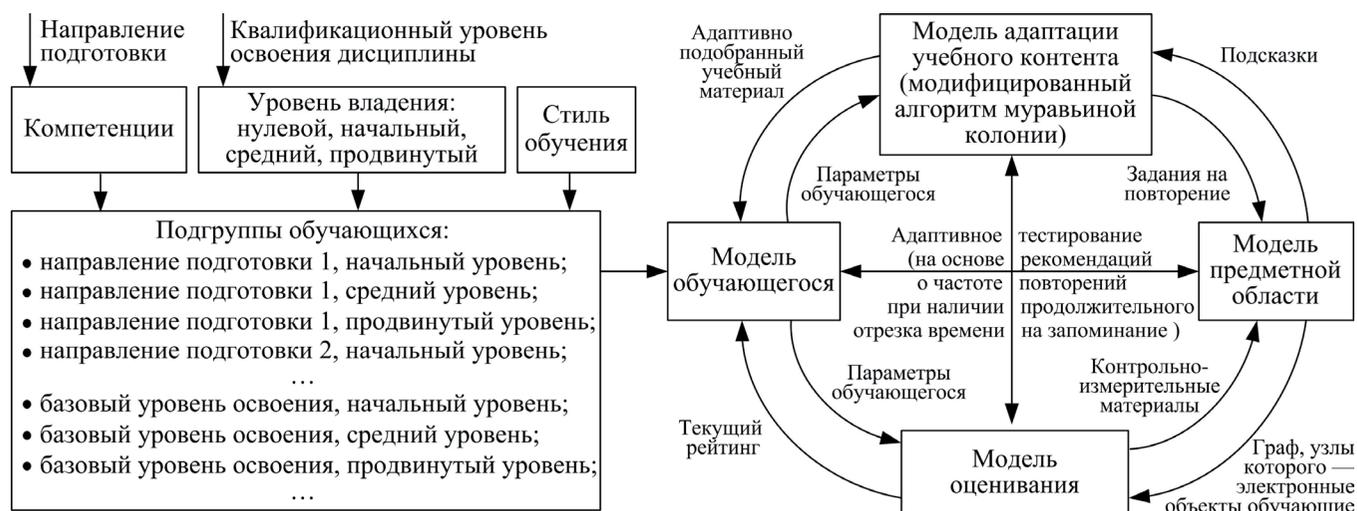


Рис. 1. Структурная схема взаимодействия базовых компонентов решения задачи построения индивидуальной образовательной траектории

Fig. 1. The structural scheme of interaction between the basic components of the solution to the problem of constructing an individual educational trajectory

области, модели обучающегося, модели адаптации и модели оценивания, которые являются базовыми компонентами любой интеллектуальной адаптивной системы обучения [2, 3]. Структурная схема взаимодействия указанных моделей показана на рисунке 1.

Модель предметной области представлена графом, узлы которого — это электронные обучающие объекты (небольшие фрагменты теории с примерами — веб-страница или файл, которые содержат несколько абзацев текста с рисунками и фрагментами кода (до 10 минут на ознакомление), задачи, тесты (до 10 вопросов), контрольные работы (до 5 задач)), сформированные на основе стратегии микрообучения и соответствующие различным квалификационным уровням освоения SQL (базовый (аналитик, маркетолог) или профессиональный (программист, разработчик баз данных)) и уровням владения SQL (начальный, средний, продвинутый).

Узлы графа связаны ребрами, для каждого из которых преподаватель указывает вес w_{ij} — число от 0 до 1, характеризующее важность предстоящего узла (задания) в данной теме. Важность типовых задач, тестов и контрольных работ имеет большие значения в сравнении со значениями важности других видов узлов (теория с примерами, сложные задачи), т. к. эти узлы в текущей теме должны быть пройдены в обязательном порядке, и только после этого будет разрешен переход к следующей теме.

Модель обучающегося («муравья» k) представлена следующим набором наиболее значимых при изучении учебной дисциплины параметров:

$$L_k = \langle C_k, U_k \rangle.$$

C_k — группа параметров, определяющих индивидуальные характеристики обучающегося, на основе которых он включается в соответствующую подгруппу:

$$C_k = \langle Q_k, V_k, St_k \rangle,$$

где:

Q_k — выбранный квалификационный уровень освоения дисциплины;

V_k — уровень владения дисциплиной (определяется по итогам предварительной контрольной работы);

St_k — стиль обучения;

U_k — группа параметров, характеризующих учебные достижения обучающегося:

$$U_k = \langle T_{ij,k}, A_{ij,k}, F_{ij,k}, H_{ij,k}, P_{ij,k}, CP_k, M_k, CT_k, RCT_k, S_{ij,k}, R_k \rangle,$$

где:

$T_{ij,k}$ — время, затрачиваемое на прохождение узлов графа;

$A_{ij,k}$ и $F_{ij,k}$ — количество успешных попыток (решение задач разными способами или повторное решение тестов и контрольных работ) и неуспешных попыток прохождения узлов, в которых предусмотрено оценивание, соответственно;

$H_{ij,k}$ — количество взятых подсказок;

$P_{ij,k}$ — количество попыток решения задания в данном узле;

CP_k — текущее положение обучающегося на графе;

M_k — память студента;

CT_k — список табу, т. е. уже решавшихся заданий;

RCT_k — перечень заданий на повторение;

$S_{ij,k}$ — оценки;

R_k — текущий рейтинг (суммарная оценка за курс).

Модель оценивания результатов обучения содержит набор различных контрольно-измерительных материалов, размещенных в узлах графа (предварительная контрольная работа, тесты, задачи типовые и повышенной сложности, итоговая контрольная работа).

Модель адаптации учебного контента содержит алгоритм построения ИОТ, основанный на алгоритме муравьиной колонии, модифицированном за счет разделения «муравьев» на группы в зависимости от выбранного направления подготовки, желаемого квалификационного уровня подготовки, а также подбора параметров и формул алгоритма таким образом, чтобы обеспечить высокую вероятность выбора тех заданий, основы решения которых были не полностью усвоены. **Алгоритм построения ИОТ обеспечивает адаптивный подбор последовательности обучающих объектов из модели предметной области на основе данных из модели обучающегося и результатов прохождения курса, получаемых с помощью модели оценивания.**

5. Модель адаптации учебного контента и модифицированный муравьиный алгоритм

При разработке модели адаптации учебного контента принимались во внимание следующие особенности поведения студента при выборе пути перемещения по графу обучающих объектов:

- студентов, которые в рамках анализа по муравьиному алгоритму выступают в роли «муравьев», можно разделить на группы в зависимости от желаемых результатов освоения курса (подобно тому, как в реальной колонии муравей выполняет определенную функцию — рабочий-строитель, солдат и т. п.). Для каждой группы студентов будут выстраиваться ИОТ различной сложности. Поэтому в качестве спецификатора муравьиной группы в модели предложено использовать желаемый квалификационный уровень освоения языка SQL или направление подготовки (специальность);
- студент k в нашей модели имеет M_k — «личную память», т. е. массив состояния, в котором сохраняются:
 - индивидуальная траектория $Tr_k(t)$ — последовательность узлов, пройденная студентом k к моменту времени t ;
 - T_k — время, затрачиваемое на прохождение узла j ;
 - количество узлов $L_k(t)$ графа и другие параметры;
- студент k обладает «зрением», которое представляет собой так называемую видимость (привлекательность) η_{ij} , характеризующую стремление посетить какой-либо из следующих доступных узлов графа. В качестве видимости предложено использовать информацию о траектории обучения наилучших студентов данной группы. Кроме того, «зрение» позволяет воспринимать подсказки — фразы, расширяющие границы понимания решаемой задачи, или указания по переходу к узлам, изучение которых позволит поработать не-

достающие знания и ускорить прохождение обязательных элементов курса;

- студент k имеет «обоняние», позволяющее определить ребро с наибольшим в данный момент времени значением «феромона» τ_{ij} , оставленным студентами данной подгруппы, решившими задание в данном узле.

Колония обучающихся имеет «общую память» — массив наилучших ИОТ студентов, завершивших курс обучения за время, не превышающее установленное для каждого квалификационного уровня освоения учебной дисциплины.

Следует отметить, что решения, основанные на кривой забывания Г. Эббингауза, в том числе метод интервальных повторений, при определении времени очередного повторения предусматривают большое количество повторений в первые дни после изучения новой темы [26]. Это в большей степени подходит для изучения иностранных языков, нежели для языка программирования, где требуется решать сложные задачи. В реальном курсе студенты, как правило, не выделяют время на решение задач по несколько раз в день и далеко не всегда занимаются ежедневно, к тому же частое решение однотипных задач не способствует углублению знаний. Курс «Базы данных», на примере которого рассматривается предложенный модифицированный алгоритм муравьиной колонии, может длиться от одного до трех семестров (от 50 до 200 аудиторных часов). За учебное занятие (два академических часа) обучающийся в зависимости от сложности изучаемой конструкции SQL обычно решает 5–15 задач, после чего продуктивность усвоения новой информации и решения задач заметно падает. Поэтому в нашей модели предложено на постоянной основе повторять только теоретические вопросы, предлагаемые студенту на каждом занятии при тестировании.

В соответствии с рекомендациями о частоте повторений при наличии продолжительного отрезка времени на запоминание [27] оптимальной считается следующая схема повторений:

- первое повторение — сразу же за первым прочтением информации (например, сразу после лекции);
- второе — примерно через полчаса после первого;
- третье — через сутки после второго;
- четвертое — спустя две-три недели после третьего;
- пятое — спустя два-три месяца после четвертого повторения.

Данная последовательность номеров повторений N и номеров дней повторений $T(N)$ использована для определения порядковых номеров дней, в которые обучающемуся предъявляется в тестах соответствующий теоретический вопрос для повторения.

Для повторения основ решения практических заданий при их забывании в данной работе предложено использовать описанную ниже модель адаптации учебного контента.

Приведем пошаговое описание модели адаптации учебного контента, основанной на использовании модифицированного алгоритма муравьиной колонии, на примере изучения SQL в курсе «Базы данных».

Шаг 1. Формирование набора параметров C_k модели обучающегося L_k .

1.1. **Выбор группы студентов, соотношенной с конкретным направлением (специальностью) с четко определенным перечнем компетенций**, например, направлением подготовки 09.03.01 «Информатика и вычислительная техника». Если речь идет об индивидуальном обучении, то указывается желаемый квалификационный уровень Q_k освоения языка SQL, выбираемый обучающимся:

- базовый уровень (аналитик, маркетолог, инженер) — подмножество операторов Data Manipulation Language языка SQL, основные операции над данными;
- профессиональный уровень (программист, разработчик баз данных, администратор баз данных) — подмножества операторов Data Manipulation Language, Data Definition Language, Data Control Language, Transaction Control Language, работа с объектами серверного компонента базы данных, обобщенные табличные выражения, оконные функции и т. д.

1.2. **Оценка уровня владения V_k языком SQL:** начальный, средний, продвинутый. Этот параметр указывается студентом и подтверждается результатами предварительной контрольной работы. Баллы за нее используются для отнесения обучающегося к соответствующей подгруппе.

1.3. **Определение на основе модели VARK стиля обучения St_k** (влияет на выбор способа представления обучающемуся теоретического материала с примерами [28], при этом у студента сохраняется возможность выбора способа представления (в виде текста, схем и таблиц или видео) в ходе изучения курса).

Шаг 2. Включение студента в соответствующую подгруппу и предложение ему траектории, выстраиваемой другими студентами для данного набора $C_k = \langle Q_k, V_k, St_k \rangle$.

Шаг 3. Инициализация набора входных параметров модифицированного алгоритма муравьиной колонии:

- множество обучающихся $S = \{s_k\}$, где k — номер обучающегося, $k = \overline{1, K}$;
- граф, структура которого аналогична структуре графа модели предметной области, содержащий значения «феромонов» τ_{ij} для каждой подгруппы студентов на ребрах между узлами;
- граф, структура которого аналогична структуре графа модели предметной области, со-

держащий значения «видимостей» узлов η_{ij} для каждой подгруппы обучающихся.

Шаг 4. Работа алгоритма муравьиной колонии (рис. 2).

Производится инициализация начального положения студента («муравья») k , который помещается в узел i с обучающим материалом (для каждой подгруппы узел i свой).

Инициализируется список доступных узлов $J_{i,k}$, которые студент k может посетить, перейдя из узла i . В список $J_{i,k}$ предложено помещать узлы как непосредственно связанные ребрами с узлом i , так и узлы из перечня на повторение RCT_k . Модель обучающегося содержит параметр CT_k — список табу, хранящий перечень уже решавшихся заданий, которые алгоритм предлагает студенту не должен. Из списка CT_k отбираются задания, при работе над которыми обучающийся k сделал более десяти неверных попыток $P_{ij,k}$ решения, и учитываются ошибки при выполнении тестов и контрольных работ, на основе чего подбираются задания из предшествующих тем для формирования перечня RCT_k . Каждое задание графа имеет связанное с ним задание из предшествующей темы либо, если предшествующей темы нет, сходное более простое задание. Без решения задач на повторение переход к рубежным тесту и контрольной работе в текущей теме невозможен.

В качестве начального малого ненулевого количества «феромона» $\tau_{ij,k0}$ на каждом ребре используется значение важности каждого ребра, установленное преподавателем для графа модели предметной области.

Шаг 5. Функционирование канала стигмергии, причем для каждого студента k определяется вероятность перехода $P_{ij,k}(t)$ из узла i в каждый узел j , принадлежащий списку $J_{i,k}$:

$$P_{ij,k}(t) = \begin{cases} \frac{\tau_{ij,k}^\alpha(t) \eta_{ij,k}^\beta(t)}{\sum_{l \in J_{i,k}} \tau_{il,k}^\alpha(t) \eta_{il,k}^\beta(t)}, & \forall j \in J_{i,k}; \\ 0, & \forall j \notin J_{i,k}, \end{cases}$$

где:

α и β — параметры, которые регулируют относительную важность «феромонного» следа в отношении видимости узла и определяются эвристически. В данном исследовании приняты $\alpha = 1$ и $\beta = 3$;

$\eta_{ij,k}(t)$ — видимость, которую предложено определять по формуле:

$$\eta_{ij,k}(t) = \frac{\sum_{b=1}^B S_{ij,b}}{\sum_{b=1}^B T_{ij,b}}$$

где:

$S_{ij,b}$ — сумма всех оценок, полученных за прохождение данного узла j студентом b , отнесенным к числу наилучших¹ для данной подгруппы, $b = \overline{1, B}$;

¹ Наилучшими считаются 20 % студентов из списка всех обучающихся, отсортированного по убыванию успеваемости.

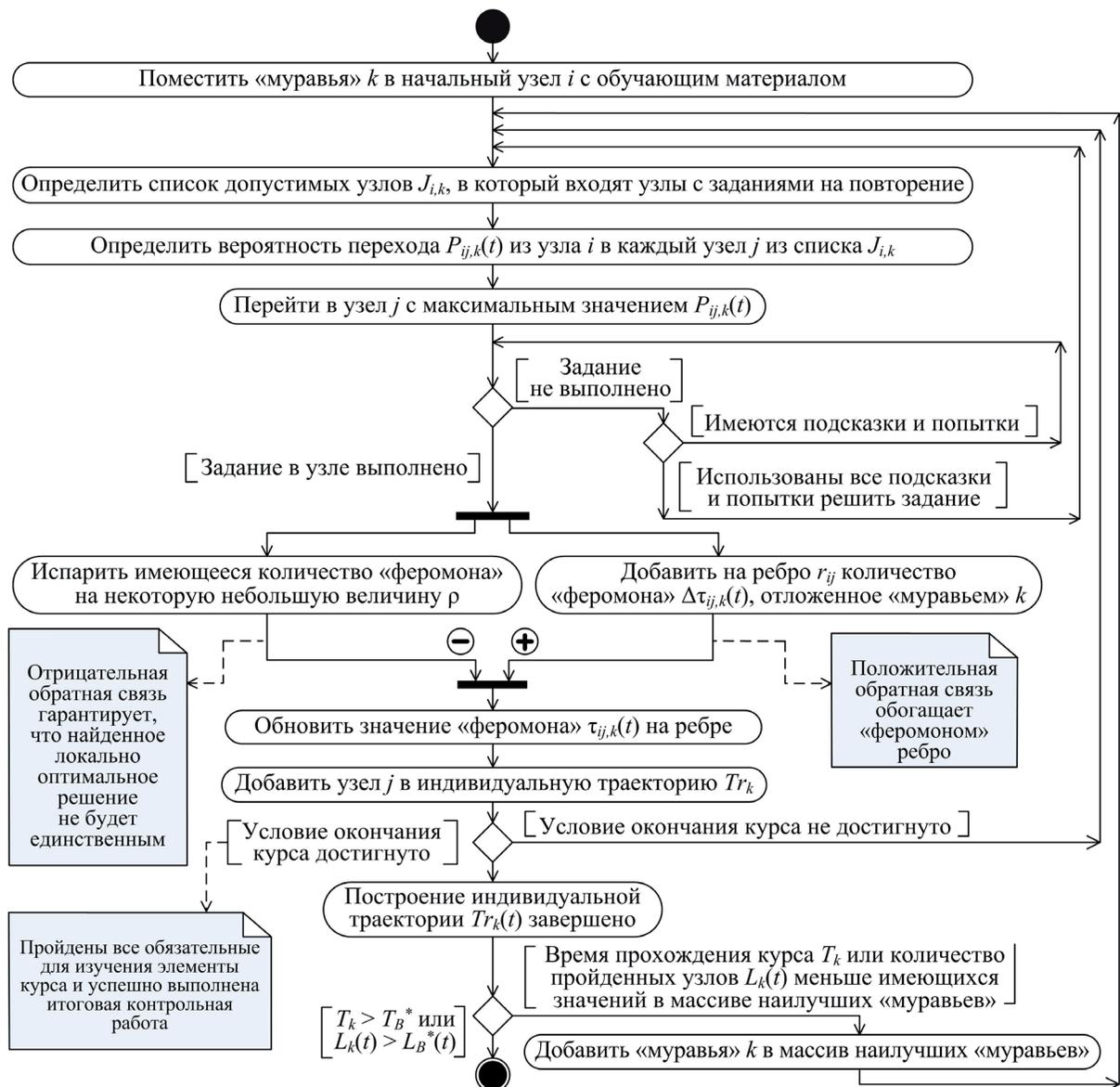


Рис. 2. Модификация муравьиного алгоритма для построения индивидуальной образовательной траектории
 Fig. 2. The modification of the ant colony optimization algorithm for constructing individual educational trajectory

$T_{ij,b}$ — суммарное время, затраченное на прохождение данного узла студентом b , отнесенным к числу наилучших для данной подгруппы, $b = \overline{1, B}$.

Шаг 6. Переход студента k в узел j с максимальным значением $P_{ij,k}(t)$ и выполнение задания в узле j .

Шаг 7. Если задание в узле j выполнено, обновляется значение «феромона» на ребре r_{ij} , соединяющем узлы i и j , в соответствии с выражением:

$$\tau_{ij,k}(t+1) = \tau_{ij,k}(t)(1-\rho) + \Delta\tau_{ij,k}(t), \forall i, j \in J_{i,k},$$

где:

ρ — величина, на которую «испаряется» имеющееся на ребре значение «феромона» ($0 < \rho \leq 1$). В данной работе предполагается, что «испарение» происходит ежедневно на протяжении курса на величину $\rho = 0,001$.

$\Delta\tau_{ij,k}(t)$ — количество «феромона», откладываемое при выполнении задания в узле j :

$$\Delta\tau_{ij,k}(t) = \begin{cases} \frac{S_{ij,k}A_{ij,k}}{S_{ijmax}T_{ij,k}} - \frac{H_{ij,k}/H_{ij}}{ZF_{ij,k}} + w_{ij,p}, & \text{если задание решено;} \\ 0, & \text{если задание не решено,} \end{cases}$$

где:

$S_{ij,k}$ — оценка за решение задания в узле j студентом k ;

S_{ijmax} — максимально возможная оценка за решение задания в узле j ;

$A_{ij,k}$ — количество успешных попыток решения задачи другим способом;

$T_{ij,k}$ — время, затраченное на решение задания в узле j студентом k ;

$H_{ij,k}$ — количество подсказок, взятых студентом k в узле j . Подсказки, как правило, содержат указа-

ние на основные изученные темы, необходимые для решения данного задания, либо краткое описание принципа решения задания, но не сам ответ [29];

H_{ij} — общее количество подсказок в узле j ;

Z — количество заданий, обязательных для решения в данной теме;

$F_{ij,k}$ — количество неуспешных попыток решить задание в узле j ;

$w_{ij,p}$ — расчетный вес задания в узле j .

Для повышения вероятности выбора алгоритмом заданий из списка на повторение RCT_k предложено расчетный вес задания вычислять по формуле:

$$w_{ij,p} = \begin{cases} 0,8 \text{ IF } R_{ij,k} > 10 \text{ OR } H_{ij,k} > H_{ij} \forall j \in RCT_k; \\ ELSE w_{ij}, \end{cases}$$

где:

$R_{ij,k}$ — количество попыток решения задания в данном узле;

w_{ij} — вес, назначаемый преподавателем.

Засчитывается только положительное решение задачи или успешное прохождение теста. Отрицательный результат не учитывается (т. е. не используется «феромон» неудачи), т. к. такой подход означает необходимость возврата на предыдущие узлы, что фактически бесполезно при изучении совершенно новой темы со слабыми связями с предыдущими темами. Если задача не решена, то студент может воспользоваться подсказкой, но это будет уменьшать оставленное количество «феромона». Если использовано максимальное количество подсказок или попыток решения задачи в текущем узле, то обучающийся перенаправляется на решение заданий на повторение из списка RCT_k .

Шаг 8. В «память» M_k студента добавляются пройденный узел j в индивидуальной траектории $Tr_k(t)$ и время $T_{j,k}$, затрачиваемое на прохождение узла j графа.

Шаг 9. Если выполнено условие окончания курса (в каждой теме пройдены все обязательные для изучения типовые задачи, тесты и контрольные работы), то построение ИОТ завершается и выполняется переход к шагу 10. В противном случае происходит возврат к шагу 5.

Шаг 10. Проверка пройденного количества узлов $L_k(t)$ и времени T_k , затраченного на прохождение курса, в индивидуальной траектории $Tr_k(t)$ для студента k на соответствие лучшему решению по сравнению с массивом соответствующих значений для наилучших $L_B^*(t)$ и $T_B^*(t)$. В случае, если $L_k(t)$ и T_k лучше, т. е. меньше, чем $L_B^*(t)$ и $T_B^*(t)$, то необходимо обновить массив $L_k^*(t)$ и $Tr_k^*(t)$.

6. Результаты использования модифицированного муравьиного алгоритма

Апробация предложенных подходов к обучению языку SQL была выполнена в Белорусско-Российском университете в рамках курса «Базы данных»

длительностью один год для студентов направления подготовки 09.03.01 «Информатика и вычислительная техника». В экспериментальной группе (17 человек) обучение велось с использованием модели обучения с предложенной модификацией муравьиного алгоритма, а в контрольной группе (19 человек) — без его использования. После предварительного тестирования все студенты были отнесены к начальному уровню.

Для оценки уровня сформированности компетенций использовался заключительный тест, оценивающий уровень подготовленности студентов по компетенциям, которые формируются в результате освоения курса «Базы данных»¹. Результаты теста были обработаны с помощью дихотомической однопараметрической модели Раша [30, 31]. Пригодность теста удостоверена рабочей программой дисциплины, а также утвержденным фондом оценочных средств. Трудность каждого тестового задания определялась отдельно для каждой компетенции [30]. Тестовые задания в наборе из 25 задач имели сопоставимую дифференцирующую способность (коэффициент дискриминации превышал 0,5) и были проверены на пригодность для принятой модели с помощью Infit MEANSQ статистики (значения статистик превышали 0,5) [32]. Полученные оценки уровня подготовленности обучающихся путем линейных преобразований были переведены в столбальную шкалу и в соответствии с принятой в рабочей программе шкалой сопоставлены с оценками «отлично», «хорошо», «удовлетворительно».

Для оценки эффективности применения предложенных подходов к изучению SQL в рамках курса «Базы данных» было проведено сравнение полученных по итогам тестирования оценок уровня подготовленности студентов в контрольной и экспериментальной группах с помощью критерия Манна—Уитни (англ. Mann—Whitney U test) [33]. В качестве основной была принята гипотеза « $H_0 = \{\text{уровень подготовленности в обеих группах одинаков}\}$ », а альтернативной — « $H_1 = \{\text{уровень подготовленности в экспериментальной группе выше}\}$ ». По результатам обработки получено критическое значение критерия Манна—Уитни $U_{кр} = 99$, эмпирическое значение $U_{эмп} = 79$. Так как $U_{кр} > U_{эмп}$, то с вероятностью 0,95 можно утверждать, что в экспериментальной группе уровень подготовленности обучающихся выше.

¹ В частности, ОПК-2 («Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности») и ОПК-8 («Способен разрабатывать алгоритмы и программы, пригодные для практического применения»). Подробнее см.: Приказ Министерства образования и науки Российской Федерации от 19 сентября 2017 года № 929 (ред. от 08.02.2021) «Об утверждении федерального государственного образовательного стандарта высшего образования — бакалавриат по направлению подготовки 09.03.01 «Информатика и вычислительная техника». https://fgosvo.ru/uploadfiles/FGOS%20VO%203++/Bak/090301_B_3_15062021.pdf

7. Заключение

Разработанная модификация муравьиного алгоритма позволяет повысить эффективность формирования компетенций при изучении различных дисциплин информатики, в первую очередь характеризующихся значительной продолжительностью и состоящих из слабо связанных между собой модулей, например, при изучении курса «Базы данных», посредством построения ИОТ на основе модели адаптации учебных материалов. Указанная модель имеет следующие отличительные особенности:

- предложено понятие спецификатора группы студентов, в качестве которого можно использовать направление подготовки (специальность) или желаемый квалификационный уровень освоения учебной дисциплины (например, для языка SQL в курсе «Базы данных» — базовый или профессиональный). Данное понятие позволяет предлагать особый набор учебных материалов для каждой группы и подгруппы (начального, среднего, продвинутого уровня), учитывающий требования конкретных профессиональных компетенций;
- организовано обязательное прохождение заданий на повторение в случае выявления пробелов в знаниях по итогам тестирования и выполнения контрольных работ, что обеспечивает более устойчивое формирование компетенций;
- для повторения изученных теоретических вопросов при прохождении тестов предложено использовать рекомендации о частоте повторений в случае наличия продолжительного отрезка времени на запоминание, что позволяет за счет частого повторения сложных в усвоении понятий сформировать прочную теоретическую базу;
- для повторения основ решения практических задач предложено из списка табу, содержащего перечень заданий, которые обучающийся пытался решать, отбирать задачи, при работе над которыми студент сделал более десяти неверных попыток, а также ошибки при выполнении тестов и контрольных работ. На этой основе подбираются сходные задачи из предшествующих тем для формирования перечня задач на повторение. Данное решение обеспечивает повторение только тех заданий, подходы к решению которых были фактически забыты или не полностью усвоены.

Основные параметры модифицированного алгоритма предложено рассчитывать следующим образом:

- «видимость» очередного узла определяется с учетом информации о траектории наилучших в данной подгруппе обучающихся, что позволяет сократить объем работ по разведыванию наилучшей ИОТ;
- количество «феромона», «оставленное» студентом при успешном решении задачи, рассчиты-

вается с учетом индивидуальных характеристик решения задачи, в первую очередь таких, как количество взятых подсказок и расчетный вес задания в узле. При превышении допустимого количества взятых подсказок или попыток решения формула для вычисления этого веса обеспечивает высокую вероятность того, что алгоритм перенаправит обучающегося к решению задач на повторение, а значит, гарантирует формирование более прочных знаний.

Список источников / References

1. Machado M. d. O. C., Bravo N. F. S., Martins A. F., Bernardino H. S., Barrere E., De Souza J. F. Metaheuristic-based adaptive curriculum sequencing approaches: A systematic review and mapping of the literature. *Artificial Intelligence Review*. 2020;54:711–754. DOI: 10.1007/s10462-020-09864-z.
2. Шершнева В. А., Вайнштейн Ю. В., Кочеткова Т. О. Адаптивная система обучения в электронной среде. *Программные системы: теория и приложения*. 2018;9(4):159–177. EDN: DNBANN. DOI: 10.25209/2079-3316-2018-9-4-3-159-177.
- [Shershneva V. A., Vainshtein Yu. V., Kochetkova T. O. Adaptive system of web-based teaching. *Program Systems: Theory and Applications*. 2018;9(4):159–177. (In Russian.) EDN: DNBANN. DOI: 10.25209/2079-3316-2018-9-4-3-159-177.]
3. Медведева С. Н. Проектирование информационных технологий обучения в профессиональной математической подготовке в инженерном образовании. Казань: Редакционно-издательский центр «Школа»; 2014. 261 с. EDN: WLJFDT.
- [Medvedeva S. N. Designing information technology of training in vocational mathematical training in engineering education. Kazan, Editorial and publishing center "Shkola"; 2014. 261 p. (In Russian.) EDN: WLJFDT.]
4. Титовская Н. В., Титовский С. Н. Методика обучения будущих IT-специалистов проектированию и разработке баз данных на основе интерактивного подхода. *Вестник Красноярского государственного педагогического университета им. В. П. Астафьева (Вестник КГПУ)*. 2019;(4(50)):75–87. EDN: CQHLYF. DOI: 10.25146/1995-0861-2019-50-4-164.
- [Titovskaya N. V., Titovsky S. N. Technique of teaching design and development of databases to future information technology experts. *Bulletin of Krasnoyarsk State Pedagogical University Named After V. P. Astafyev (Bulletin KSPU)*. 2019;(4(50)):75–87. (In Russian.) EDN: CQHLYF. DOI: 10.25146/1995-0861-2019-50-4-164.]
5. Rastegarmoghadam M., Ziarati K. Improved modeling of intelligent tutoring systems using ant colony optimization. *Educational Information Technologies*. 2016;22:1067–1087. DOI: 10.1007/s10639-016-9472-2.
6. Menai M., Alhunitah H., Al-Salman H. Swarm intelligence to solve the curriculum sequencing problem. *Computer Applications in Engineering Education*. 2018;26(5):1393–1404. DOI: 10.1002/cae.22046.
7. Bobryakov A. V., Borisov V. V., Yanukovich S. P., Zakharchenkov K. V. A method for managing engineers training processes using swarm intelligence algorithms. *Proc. 5th Int. Conf. on Information Technologies in Engineering Education (Inforino 2020)*. New Jersey, USA, Institute of Electrical and Electronics Engineers Inc.; 2020:1–5. EDN: FSUSCL. DOI: 10.1109/Inforino48376.2020.9111788.
8. Dharshini A. P., Chandrakumarmangalam S., Arthi G. Ant colony optimization for competency based learning objects sequencing in e-learning. *Applied Mathematics and Computation*. 2015;263:332–341. DOI: 10.1016/j.amc.2015.04.067.

Vestnik Chelyabinskogo Gosudarstvennogo Pedagogicheskogo Universiteta. 2017;(6):108–116. (In Russian.) EDN: ZHZYML.]

29. *Валуева Е. А., Лаптева Е. М.* Феномен подсказки при решении задач: взгляд со стороны психологии творчества. Часть 2. Эффекты подсказки в решении сложных когнитивных задач. *Психология. Журнал Высшей школы экономики*. 2012;9(3):140–162. EDN: QYXAMT.

[*Valuyeva E. A., Lapteva E. M.* The phenomenon of hint in problem solving: A creativity psychology point of view. Part 2. Hint effects in complex cognitive tasks. *Psychology. Journal of Higher School of Economics*. 2012;9(3):140–162. (In Russian.) EDN: QYXAMT.]

30. *Соколова Т. Н., Гусятников В. Н., Безруков А. И., Каюкова И. В.* Методика оценки набора компетенций на основе результатов тестирования. *Фундаментальные исследования*. 2020;(12):209–215. EDN: IHYEON. DOI: 10.17513/fr.42935.

[*Sokolova T. N., Gusyatinikov V. N., Bezrukov A. I., Kayukova I. V.* Methodology for assessing a set of competencies based on testing results. *Fundamental'nyye Issledovaniya*. 2020;(12):209–215. (In Russian.) EDN: IHYEON. DOI: 10.17513/fr.42935.]

31. *Маслак А. А.* Теория и практика измерения латентных переменных в образовании. М.: Юрайт; 2020. 255 с. EDN: RQBLAY.

[*Maslak A. A.* Theory and practice of measuring latent variables in education. Moscow, Yurayt; 2020. 255 p. (In Russian.) EDN: RQBLAY.]

32. *Родионов А. В., Братищенко В. В.* Применение IRT-моделей для анализа результатов обучения в рамках компетентностного подхода. *Современные проблемы науки и образования*. 2014;(4):1–11. EDN: STRMVH.

[*Rodionov A. V., Bratischenko V. V.* Application IRT-model for the analysis training results within the competence approach. *Sovremennye Problemy Nauki i Obrazovaniya*. 2014;(4):1–11. (In Russian.) EDN: STRMVH.]

33. *Кондрашова Е. В.* Применение методов математической статистики при проведении педагогического эксперимента. *Современные наукоемкие технологии*. 2021;(2):162–169. EDN: MOXZVO. DOI: 10.17513/snt.38512.

[*Kondrashova E. V.* Application of mathematical statistics methods in a pedagogical experiment. *Sovremennyye*

Naukoyemkiye Tekhnologii. 2021;(2):162–169. (In Russian.) EDN: MOXZVO. DOI: 10.17513/snt.38512.]

Информация об авторах

Захарченков Константин Васильевич, канд. тех. наук, доцент кафедры «Программное обеспечение информационных технологий», электротехнический факультет, Белорусско-Российский университет, г. Могилев, Беларусь; *ORCID*: <https://orcid.org/0000-0001-8185-3010>; *e-mail*: zaharchenkovkv@mail.ru

Мрочек Татьяна Владимировна, канд. тех. наук, доцент кафедры «Программное обеспечение информационных технологий», электротехнический факультет, Белорусско-Российский университет, г. Могилев, Беларусь; *ORCID*: <https://orcid.org/0000-0002-4402-6259>; *e-mail*: mrovlad@mail.ru

Кутузов Виктор Владимирович, канд. тех. наук, доцент, зав. кафедрой «Программное обеспечение информационных технологий», электротехнический факультет, Белорусско-Российский университет, г. Могилев, Беларусь; *ORCID*: <https://orcid.org/0000-0002-4767-6003>; *e-mail*: kutuzov.bru@yandex.ru

Information about the authors

Konstantin V. Zakharchenkov, Candidate of Sciences (Engineering), Associate Professor at the Department of “Information Technologies Software”, Faculty of Electrical Engineering, Belarusian-Russian University, Mogilev, Belarus; *ORCID*: <https://orcid.org/0000-0001-8185-3010>; *e-mail*: zaharchenkovkv@mail.ru

Tatiana V. Mrochek, Candidate of Sciences (Engineering), Associate Professor at the Department of “Information Technologies Software”, Faculty of Electrical Engineering, Belarusian-Russian University, Mogilev, Belarus; *ORCID*: <https://orcid.org/0000-0002-4402-6259>; *e-mail*: mrovlad@mail.ru

Victor V. Kutuzov, Candidate of Sciences (Engineering), Docent, Head of the Department of “Information Technologies Software”, Belarusian-Russian University, Mogilev, Belarus; *ORCID*: <https://orcid.org/0000-0002-4767-6003>; *e-mail*: kutuzov.bru@yandex.ru

Поступила в редакцию / Received: 29.04.24.

Поступила после рецензирования / Revised: 20.05.24.

Принята к печати / Accepted: 21.05.24.