

<https://doi.org/10.31891/2219-9365-2022-72-4-11>

УДК 004.42

Олександр ДЕНЕСЯК

Вінницький національний технічний університет

[alexdenesiak96@gmail.com](mailto:alexdenesiak96@gmail.com)

Євген ПАЛАМАРЧУК

Вінницький національний технічний університет

<https://orcid.org/0000-0002-7443-099X>

## ДОСЛІДЖЕННЯ ЗАСТОСУВАННЯ МЕТОДУ АДАПТАЦІЇ ПОВЕДІНКИ АГЕНТІВ В СИСТЕМІ ПРОКТОРИНГУ В ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЯХ АНАЛІЗУ КОНТЕКСТУ В СИСТЕМАХ ОЦІНЮВАННЯ ЗНАТЬ

*У статті на основі аналізу прокторингових систем виділено їх основні функції та розроблено математичну модель з адаптаційною складовою такої системи, що в подальшому стане частиною інформаційних технологій аналізу контексту в системах оцінювання знань. В процесі розробки системи досліджено та закладено в модель механізм, який ефективно виконує взаємодію із зовнішнім середовищем та забезпечує контроль процесу адаптації прокторингової системи протягом складання іспиту, що базується на роботі агентів, які в подальшому об'єднуються в мультиагентну систему.*

*Застосування мультиагентних систем зумовлено необхідністю генерування правил для контролю проходження тесту на кожному з моментів часу індивідуально для кожного здобувача, на основі результатів аналізу його поведінки.*

*Ключові слова: агент, мультиагентні системи, адаптація, прокторинг, системи прокторингу, математична модель, системи оцінювання знань, нечіткі множини.*

Oleksandr DENESIAK, Yevhen PALAMARCHUK

Vinnitsia National Technical University

## STUDY OF THE APPLICATION OF THE METHOD OF ADAPTATION OF THE BEHAVIOR OF AGENTS IN THE PROCTORING SYSTEM IN INFORMATION TECHNOLOGIES OF CONTEXT ANALYSIS IN KNOWLEDGE ASSESSMENT SYSTEMS

*Based on the analysis of proctoring systems, the article highlights their main functions and develops a mathematical model with an adaptive component of such a system, which will later become a part of information technologies for context analysis in knowledge assessment systems. In the process of developing the system, a mechanism that efficiently interacts with the external environment and ensures control of the process of adaptation of the proctoring system during the exam, based on the work of agents that are later combined into a multi-agent system, was researched and embedded in the model.*

*The use of multi-agent systems is due to the need to generate rules to control the passing of the test at each moment of time individually for each applicant, based on the results of the analysis of his behavior.*

*Key words: agent, multiagent systems, adaptation, proctoring, proctoring systems, mathematical model, knowledge evaluation systems, fuzzy sets.*

### Постановка проблеми у загальному вигляді та її зв'язок із важливими науковими чи практичними завданнями

Розвиток інтелектуальних технологій дає можливість застосування нових методів та інструментів у освітній галузі, які дозволяють автоматизувати і індивідуалізувати процес навчання. Важливим його етапом є об'єктивне оцінювання знань. Як правило, на цьому етапі вступає у роль людина-екзаменатор. Але сучасні технології дозволяють певною мірою автоматизувати і підвищити ефективність і цього процесу.

Системи оцінювання знань забезпечують незалежну та ефективну оцінку знань з мінімальним впливом людського фактору. Їх сучасні моделі базуються на технологіях штучного інтелекту, які в процесі використання можуть навчатися та приймати рішення, базуючись на поведінці здобувача.

Процес оцінювання знань є важливою складовою, тому що визначає рівень знань здобувачів у кожному з напрямків, що в свою чергу показує рівень підготовки фахівця. Тому забезпечення якісного контролю процесу перевірки знань є пріоритетною задачею. Контроль знань стає більш складним з переходом до дистанційного режиму, оскільки в разі потрібно збільшувати людські ресурси для моніторингу здобувачів під час процесу тестування, що не завжди є можливим та раціональним. Тому доречно буде використати універсальне рішення для аудиторних та дистанційних перевірок знань, в якому моніторинг поведінки здобувачів виконують системи оцінювання знань.

Задачею залишається автоматичне оцінювання процесу перевірки знань з урахуванням індивідуальних характеристик здобувачів та прийнятті рішень щодо продовження оцінювання знань враховуючи їх поведінку.

### Аналіз стану проблеми

Одним з рішень в даному напрямку є використання прокторингових систем, а саме автоматичних систем, які виконують аналіз та прийняття рішень без необхідності залучення людини. Існує певний спектр прокторингових систем, які забезпечують контроль процесу складання іспитів. В таблиці 1 наведені дані про найбільш відомі з них (Moodle Proctoring, Examity, Kryterion, ProctorU, Respondus, Loyalist Exam Services).

Таблиця 1.

Посилання та історична інформація про прокторингові системи

Назва системи	Веб-посилання	Дані релізу
Examity	<a href="https://www.examity.com/">https://www.examity.com/</a>	Examity вийшов на ринок онлайн-прокторингу в 2013 році.
Moodle Proctoring	<a href="https://moodle.org/plugins/">https://moodle.org/plugins/</a>	Перша версія випущена в 2003; Головний розробник Мартін Дугіамас.
Kryterion	<a href="https://www.kryteriononline.com/">https://www.kryteriononline.com/</a>	Запущено в 1999 році; Дрейк Міжнародна компанія (засновник Prometric у 1990 році).
ProctorU	<a href="https://www.proctoru.com/">https://www.proctoru.com/</a>	Заснована в 2008 році; асоціюється з Університетом Ендрю Джексона.
Respondus	<a href="https://web.respondus.com/">https://web.respondus.com/</a>	Respondus заснований у 2000 році.
Loyalist Exam Services	<a href="https://www.loyalistexamservices.com/">https://www.loyalistexamservices.com/</a>	Відділ коледжу лоялістів в Онтаріо, Канада.
TeSLA aims	<a href="https://tesla-project-eu.azurewebsites.net/op4re-by-proctorexam/">https://tesla-project-eu.azurewebsites.net/op4re-by-proctorexam/</a>	Компанія Tesla була заснована у 2003 році Мартіном Еберхардом і Марком Тарпеннінгом у Сан-Карлосі, штат Каліфорнія.
Xproctor	<a href="https://midway.libguides.com/informationtechnology/onlinelearning">https://midway.libguides.com/informationtechnology/onlinelearning</a>	Онлайн-тести в Мідвейському університеті.

Аналіз їх особливостей і характеристик показує наступні особливості. У прокторинговій системі Moodle Proctoring використовується камера. Основний принцип роботи системи базуються на автоматичному зніманні зображення кожні 30 секунд (є можливість налаштування розміру зображення та інтервалу захоплення) і його зберіганні у форматі PNG. Також є можливість перевірки зображення користувача із зображенням профіля використовуючи додаткові служби Amazon Rekognition (AWS) або Brainstation Facerecognition (BS). Дана система вимагає втручання проктора, що перевіряє будь-яку підозрілу активність та звітує. Для зручності до журналу історії прокторингової системи можна застосовувати пошукові засоби, а самі зображення зберігаються в даних Moodle як невеликі PNG зображення.

Система прокторингу ProctorU використовує мікрофон та камеру, але відносить до типу live proctoring, тобто проктори слідкують за діями здобувача протягом всього іспиту використовуючи веб-камеру. Також задачею прокторів є перевірка відсутності в наявності недозволених матеріалів перед початком іспиту та підтвердження особи студента. Обов'язком студентів є підтримання безперервного аудіо та відео зв'язку із проктором протягом усього процесу складання іспиту.

Система прокторингу Kryterion є комерційним проектом, що пропонує своє гібридне рішення, що дозволяє підвищити рівень безпеки в порівнянні з ProctorU. Запропоноване гібридне рішення доповнює автоматизований нагляд за допомогою підготовлених прокторів, які мають можливість втрутитися чи перервати тест.

Система Xproctor має функціональність, що дозволяє перевірити автентичність здобувачів та можливість постійного відстежування та контролю їх за допомогою функцій розпізнавання облич, потокового відео поведінки, аудіо та фотографічних методів.

Система прокторингу TeSLA aims методи перевірки якої базуються на біометричних властивостях. А саме на методах біометрії розпізнавання обличчя, розпізнавання голосу, аналізу натискання клавіш та відбитків пальців, що дозволяють досягти високого рівня надійності та переконатися, що відповіді під час проходження тесту дає сам здобувач.

Аналіз параметрів цих систем показує, що всі вони використовують широкий спектр функціональності, який базується на сучасних технологіях, включаючи механізми роботи з біометричними

складовими. Але такі системи прокторингу не мають адаптивної складової до кожного здобувача, яка могла приймати правила щодо подальшого проходження тесту на деяких часових інтервалах. Фактично функцію адаптації в прокторингових системах виконує людина, яку ще називають проктором.

Тобто остаточне рішення про продовження або завершення процесу складання іспиту приймає проктор. Відповідно чим більша кількість здобувачів одночасно складає той чи інший іспит, тим більшу кількість прокторів потрібно задіяти, що відображається у фінансовій та людській складових.

Агентний підхід у прокторингу

Поставлена задача вимагає використати механізм, що може забезпечити адаптаційні функції прокторингової системи оцінювання знань з урахуванням поведінки кожного здобувача у певний інтервал часу на всьому проміжку складання тесту та може на основі поведінки здобувача генерувати правила, щодо подальшого проходження тесту.

Для її реалізації пропонується використання технології агентів, які в подальшому будуть об'єднані в мультиагентну систему. Кожен з агентів може виконувати індивідуальну частину роботи, а загальні формування правил виконуються базуючись на результатах кожного з агентів.

Інтелектуальний посередник (агент) представляє собою деяку автономну обчислювальну одиницю з елементами штучного інтелекту та може взаємодіяти з навколишнім середовищем. Агенти характеризуються вмінням здійснювати дії, раціональністю, здатністю взаємодіяти з навколишнім середовищем. Тобто інтелектуальний агент володіє властивостями:

- проактивності, тобто агент показує керовану цілями поведінку, здійснюючи дії спрямовані на досягнення цілей;
- соціальності, тобто агент взаємодіє з іншими сутностями зовнішнього середовища;
- реактивності, тобто агент проявляючи ініціативу відчуває зовнішнє середовище і реагує на зміни в ньому;
- автономності, тобто здатність функціонувати без втручання у нашому випадку проктора і здійснювати контроль внутрішнього стану та своїх дій.

Мультиагентні системи (МАС) - це системи, що забезпечують спілкування між агентами, що передають між собою деяку інформацію в процесі розв'язання поставлених задач. Предметно-незалежне ядро у архітектурі МАС складається з наступних компонентів (рис. 1): служба прямого доступу, служба повідомлень, бібліотека класів агентів, спільнота агентів, онтологія. Що дозволяє при проектуванні розглядати організацію агентів як набір ролей, що мають певні відносини одне з одним.

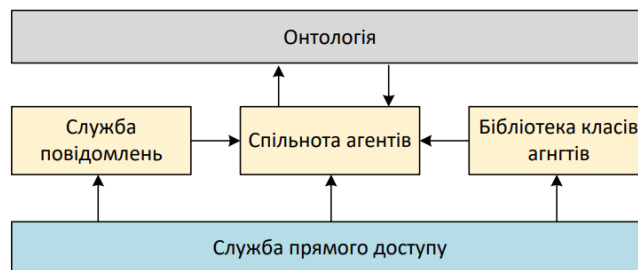


Рис 1. Архітектура ядра мультиагентної системи

Метою роботи є розробка математичної моделі з адаптаційною складовою, що дозволяє генерувати правила продовження складання тесту для здобувачів у конкретний часовий інтервал протягом всього часу проходження тесту базуючись безпосередньо на поведінці здобувачів. Для реалізації адаптаційної складової вибрано агентну технологію, за роботу якої відповідає мультиагентна система. Спроектвана система прокторингу в майбутньому стане підсистемою інформаційних технологій аналізу контексту у системах оцінювання знань.

### Проектування базової моделі прокторингу з використанням агентів

Визначимо ключові функції системи прокторингу, які працюють на всьому проміжку процесу тестування:

- розпізнавання обличчя здобувача;
- розпізнавання сторонніх осіб у кадрі;
- розпізнавання звуків у кадрі;
- моніторинг переходу між вкладками браузера;
- моніторинг клавіатурного почерку;
- відслідковування рухів очей.

Особливістю математичної моделі для поточної реалізації прокторингової системи є циклічна перевірка станів об'єкта у кожний момент часу, що в свою чергу забезпечує найвищу ефективність роботи.

Тобто функція, яка приймає деякі вхідні параметри (обличчя здобувача, активна вкладка браузера, сторонні особи та інші) на вхід і віддає певний результат в кожний момент часу. Значення параметру в певний момент часу може приймати значення 0 - якщо здобувач порушив правило та відповідно 1 - якщо порушення не було. Результат функції описується значеннями на числовому проміжку від 0 до 1. Чим ближче до 1 тим менше правил було порушено здобувачем за весь проміжок складання тесту.

Введемо наступні позначення для кожного з параметрів та відповідні позначення для відображення поточного значення стану:

- P - вхідні параметри,  $P_i$  - конкретний параметр,  $P^*$  - поточний стан вхідного параметру у кожному з одиниць часу;
- S - здобувач,  $S_i$  - конкретний (поточний) здобувач;
- B - активна вкладка браузера,  $B^*$  - поточний стан браузерної вкладки поточного здобувача;
- R - обличчя здобувача,  $R^*$  - поточний стан обличчя здобувача;
- O - фонове зображення з камери,  $O^*$  - поточний стан фонового зображення з камери;
- K - клавіатурний почерк,  $K^*$  - поточний стан клавіатурної послідовності;
- A - рух очей,  $A^*$  - поточний стан положення очей;
- L - звуки у кадрі,  $L^*$  - поточний стан сторонніх звуків у кадрі;
- T - загальний час;
- $t^i$  - одиниця поточного часу.

Потрібно врахувати той факт, що всі обчислення виконуються на певному проміжку часу T, який фактично є часом процесу складання тесту. Як зазначалося вище для досягнення поставленої задачі для системи прокторингу кожен з вхідних параметрів потрібно обчислювати на конкретних часових інтервалах (в деякі моменти часу) і відповідно обчислити загальне значення суми за весь період.

Визначимо базову формулу для конкретного вхідного параметру на прикладі параметра браузерної вкладки та сформулюємо загальну формулу обчислення для кожного з параметрів ( $P_i$ ) та зобразимо їх на рис 1:

$$B = \sum_{i=0}^n B^*(t^i) \quad (1)$$

$$P_i = \sum_{i=0}^n P^*(t^i) \quad (2)$$

де,

$i$  - одиниця часу (секунда) на проміжку  $0 \leq i \leq T$ .

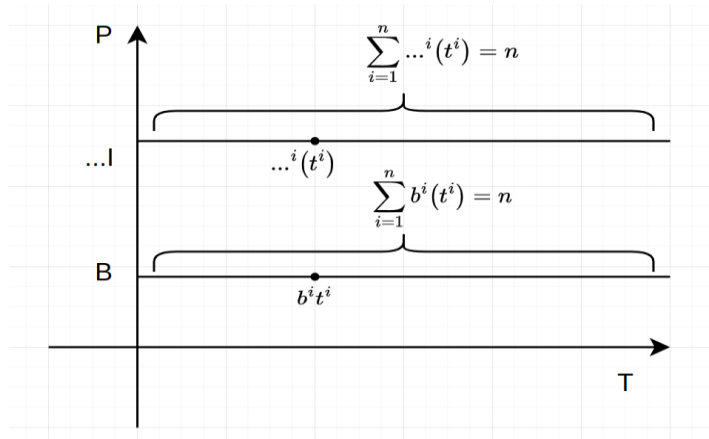


Рис 1. Зображення залежностей значень параметрів від часу.

Для загального випадку можна представити базовий приклад функції обчислення результату для конкретного здобувача виразом:

$$S_i(\dots P_i): [0; 1] = \sum_{i=1}^n \sum_{j=1}^m \frac{P_j(t_i)}{n} \quad (3)$$

У вище наведеній формулі вказано залежність конкретного здобувача ( $S_i$ ) від вхідних параметрів ( $\dots P_i$ ) та описано допустимий результат, що може повернути функція, який знаходиться на проміжку [0; 1]. Формула (3) не враховує динамічну складову та рейтинг кожного з параметрів.

Тому, для оптимального розрахунку і коректної роботи проекрованої функції потрібно задати коефіцієнти важливості та критичності, що будуть знаходитися у тих же межах, що й значення параметрів, тобто від 0 до 1. Внаслідок чого результуючий об'єкт залишиться в діапазоні значень від 0 до 1. Значення важливості та критичності параметрів будуть обчислюватися на основі таблиць, які базуються на експертних даних. В базовому випадку достатньо, щоб експертами стали викладачі конкретної установи де проводиться тестування. А в подальшому з збільшенням використання системи оцінювання буде зростати база даних про коефіцієнти важливості та критичності параметрів. В таблиці 2 приставлений приклад експертних даних для коефіцієнтів важливості та критичності параметрів. Також використання коефіцієнтів дозволить вести систему рейтингування для вхідних параметрів.

Таблиця 2.

Експертні дані про коефіцієнти параметрів

Номер	Назва параметру	Значення важливості параметру [0;1]	Значення критичності параметру [0;1]
1	Моніторинг переходу між вкладками браузера	0.5	0.6
2	Розпізнавання обличчя здобувача	0.9	0.9
3	Розпізнавання сторонніх осіб у кадрі	0.3	0.1
4	Розпізнавання звуків у кадрі	0.4	0.2
5	Моніторинг клавіатурного почерку	1	1
6	Відслідковування рухів очей	0.2	0.1

Враховавши коефіцієнти рейтингування параметрів базова формула (3) прийме наступний вигляд:

$$S_i(T) = \sum_{i=1}^n \sum_{j=1}^m \frac{P_j(t_i)}{n} k_{ef} k_c \quad (4)$$

Наступним кроком добавимо адаптивну складову, яка буде відповідати за генерування правил продовження складання тесту на певних часових інтервалах. Тобто замінимо класичну функцію проктора програмними агентами. Для цього виділемо окремі агенти для кожного з параметрів  $P_i$ , що матимуть ізольовану область та окрему відповідальність за дії конкретного параметра.

Функції агентів в прокторинговій системі реалізуємо базуючись на механізмі синтезу нечітких правил на основі існуючих. В нашому випадку суть агента полягає у аналізі стану ( $P_i$ ) в кожний з моментів часу та прийняті рішення, враховуючи поведінку здобувача. Тобто генеруванні правил про продовження проходження тесту для існуючих станів моделі ( $P^*$ ).

Базовий список правил наведено нижче, який може бути розширений відповідно до потреб навчальних закладів:

- дозвіл на проходження;
- текстове попередження відповідно до дій здобувача;
- надсилання повідомлення адміністрації;
- критичний вихід з системи;
- блокування проходження.

Оскільки вхідних параметрів у нашому випадку визначена кількість  $n$ , то виникає необхідність переходу до використання МАС з  $n$  кількістю агентів. Що, в свою чергу, має забезпечити якісну роботу системи з генеруванням відповідних правил проходження.

Для кожного з станів моделі, що задається поточним значенням  $P^*$  і представляє собою характеристику агента задамо два граничних стани  $p_1$  та  $p_2$  та відповідні функції приналежності  $\mu_1$  та  $\mu_2$ .

Запишемо формулу оцінки значення стану  $P^*$ , який базується на визначенні ступеня близькості  $\omega$  поточного стану до станів  $p_1$  та  $p_2$  у конкретний момент часу, задавши  $\omega = 0$  для  $p_1$ , і  $\omega = 1$  відповідно  $p_2$  отримаємо формулу:

$$\mu^*(t_i) = \mu_1(1 - \omega) + \mu^2\omega \quad (5)$$

або її графічне представлення для вхідного параметру  $B$  на рис. 2.

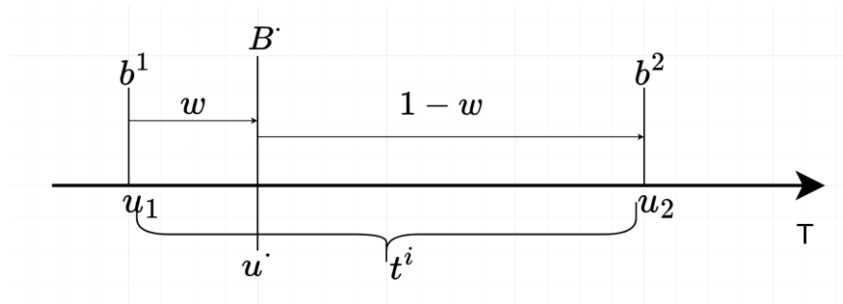


Рис. 2. Графічне представлення механізму адаптації нечіткого правила

Оскільки функція приналежності  $\mu^*$  пропорційна зміні параметру стану агента  $P^*$  від  $p_1$  до  $p_2$  вище зазначену формулу (4) можна записати у наступному вигляді:

$$S_i = \sum_{i=1}^n \sum_{j=1}^m \mu_j^*(t_i) k_{ef} k_c \quad (6)$$

$$S_i(t_i) = \sum_{j=1}^n P_j(t_i) k_{ef} k_c = \sum_{j=1}^n \mu_j^*(t_i) k_{ef} k_c \quad (7)$$

де,

$k_{ef}$  – коефіцієнт важливості параметру,

$k_c$  – коефіцієнт критичності параметру,

$\mu_j^*$  – функція приналежності  $j$  параметру в  $i$  момент часу,

$t_i$  – певний момент часу,

$S_i(t_i)$  – поведінка здобувача в певний момент часу

Процес обчислення мультиагентною системою для поточного агента в конкретний момент часу закладено у формулі (6), яка в свою чергу для досягнення найвищого коефіцієнта довіри використовує коефіцієнти важливості та критичності.

Для забезпечення плавності роботи прокторингової систему інформаційних технологій аналізу контексту в системах оцінювання знань, мультиагентна система формує правила проходження тесту базуючись сумую обчислень на певних часових блоках, наприклад, п'ять часових блоків, часовий інтервал кожного з яких 3 секунди та містить відповідну вхідну кількість параметрів. Коефіцієнт критичності дозволяє застосувати рівень важливості для кожного з вхідних параметрів, тобто його критичність при аналізі стану, що враховує грубе порушення правил проходження тесту, яке може призвести до негайного завершення тесту.

Таким чином результатом роботи агентної системи є формування правил проходження тесту та прийняття рішення відповідно до їх порушення здобувачем. МАС може, як оцінити загальний процес проходження тесту, тобто видати результат довіри до процесу проходження, що лежить на проміжку від 0 до 1, так і екстрено завершити складання тесту.

#### Висновки з даного дослідження і перспективи подальших розвідок у даному напрямі

В статті запропоновано адаптивну математичну модель для прокторингової системи інформаційних технологій аналізу контекста у системах оцінювання знань. Модель базується на головних функціях прокторингової системи, а саме розпізнаванні обличчя здобувача, моніторинг переходу між вкладками браузера, відслідковуванні клавіатурного почерку, виявленні сторонніх звуків, відслідковуванні рухів очей здобувача та розпізнавання сторонніх осіб у кадрі.



Адаптаційну складову моделі забезпечує мультиагентна система, а самий процес адаптації в даному випадку безпосередньо залежить від дій кожного здобувача в певних часових інтервалах на всьому проміжку складання тесту. Математична модель спроектована таким чином, що дозволяє розширення п кількістю вхідних параметрів не змінюючи архітектуру. Важливим елементом є підтримання розробленою моделлю системи рейтингування кожного з вхідних правил, що включає важливість та критичність параметру.

Запропонована математична модель прокторингової системи має забезпечити ефективний контроль на всьому часовому проміжку складання тесту, генеруючи правила проведення іспиту індивідуально для кожного здобувача відповідно до його поведінки.

### Література

1. Palamarchuk, Y.A., Kovalenko, O.O. Optimization of Electronic Test Parameters in Learning Management Systems/Y.A.Palamarchuk.–CEUR Workshop Proceedings, Proceedings of the 2nd International Workshop on Information-Communication Technologies & Embedded Systems (ICTES 2020), Mykolaiv, Ukraine (online), November 12, 2020.Vol-2762, p.98-109. – Режим доступу: <http://ceur-ws.org/Vol-2762/paper6.pdf>.
2. Bisikalo, Oleg, Models of Behavior of Agents in the Learning Management System / Oleg Bisikalo, Olena Kovalenko, Yevgen Palamarchuk // Матеріали XIV-ої Міжнародної науково-технічної конференції “Комп’ютерні науки та інформаційні технології (CSIT -2019)”. Том 3. – Львів, 2019. – С. 222-227.
3. Y. Palamarchuk, “METHODS OF BUILDING MICROSERVICE ARCHITECTURE OF E-LEARNING SYSTEMS”, ITKI, vol. 53, no. 1, pp. 43–54, Feb. 2022.
4. Bugaychenko, D. Y. MASL: A logic for the specification of multiagent real-time systems. // Proc. 5th International Central and Eastern European Conference on Multi-Agent Systems.- Leipzig (Germany): Springer-Verlag, 2017.- Pp. 183–192.
5. Müller J., Fisher K. Application Impact of Multi-agent Systems and Technologies: A Survey // Agent-oriented software engineering. 2014. pp. 27–53.
6. J.-P. Briot, C. Felicissimo and C. J. de Lucena, «A Norm-Based Approach for the Modeling of Open Multiagent Systems,» in Int. Conf. on Agents and Artificial Intelligence ICAART, Porto, 2009, pp. 540-546.
7. Rahimi Shahram. A multi-agent framework for stock trading / Shahram Rahimi, Raju Tatikunta, Raheel Ahmad, Bidyut Gupta // International Journal of Intelligent Information and Database Systems 2009. – Vol. 3, No. 2 pp. 203 – 227.
8. Mukhin, V., Kuchuk, N., Kosenko, N., Kuchuk, H. and Kosenko, V. (2020), “Decomposition Method for Synthesizing the Computer System Architecture, Advances in Intelligent Systems and Computing”, AISC, vol. 938, pp 289-300, DOI:[https://doi.org/10.1007/978-3-030-16621-2\\_27](https://doi.org/10.1007/978-3-030-16621-2_27).
9. Kuchuk N. Method for calculating of R-learning traffic peakedness / N. Kuchuk; O. Mozhaiev, M. Mozhaiev; H. Kuchuk // 2017 4th International Scientific-Practical Conference Problems of Infocommunications Science and Technology, PIC S and T 2017. – 2017. – P. 359 – 362. URL:<http://dx.doi.org/10.1109/INFOCOMMST.2017.8246416>.
10. Leitao P., Vrba P. Recent Developments and Future Trends of Industrial Agents // Holonic and Multi-Agent Systems for Manufacturing. 2011. LNCS 6867. pp. 15–28.
11. J.-P. Briot, C. Felicissimo and C. J. de Lucena, «A Norm-Based Approach for the Modeling of Open Multiagent Systems,» in Int. Conf. on Agents and Artificial Intelligence ICAART, Porto, 2009, pp. 540-546.
12. О. І. Денесяк і Є. А. Паламарчук, «КОМПЛЕКСНА СИСТЕМА ПРОКТОРИНГУ В ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЯХ АНАЛІЗУ КОНТЕКСТУ В СИСТЕМАХ ОЦІНЮВАННЯ ЗНАТЬ», Вісник ВПІ, вип. 6, с. 93–99, Груд. 2021
13. Катренко А.В. Системний аналіз об’єктів та процесів комп’ютеризації: Навчальний посібник / А.В. Катренко. – Львів: Новий світ, 2003. – 424 с.
14. В. М. Павленко і Ю. Ю. Свіріна, «Мультиагентний підхід при побудові системи технічного обслуговування і ремонту автомобіля,» Новітні технології розвитку автомобільного транспорту, 2018, с. 187.

### References

1. Palamarchuk, Y.A., Kovalenko, O.O. Optimization of Electronic Test Parameters in Learning Management Systems/Y.A.Palamarchuk.–CEUR Workshop Proceedings, Proceedings of the 2nd International Workshop on Information-Communication Technologies & Embedded Systems (ICTES 2020), Mykolaiv, Ukraine (online), November 12, 2020.Vol-2762, p.98-109. – Rezhym dostupu: <http://ceur-ws.org/Vol-2762/paper6.pdf>.
2. Bisikalo, Oleg, Models of Behavior of Agents in the Learning Management System / Oleg Bisikalo, Olena Kovalenko, Yevgen Palamarchuk // Materialy XIV-oi Mizhnarodnoi naukovo-tekhnichnoi konferentsii “Komp’iuterni nauky ta informatsiini tekhnologii (CSIT -2019)”. Tom 3. – Lviv, 2019. – S. 222-227.
3. Y. Palamarchuk, “METHODS OF BUILDING MICROSERVICE ARCHITECTURE OF E-LEARNING SYSTEMS”, ITKI, vol. 53, no. 1, pp. 43–54, Feb. 2022.
4. Bugaychenko, D. Y. MASL: A logic for the specification of multiagent real-time systems. // Proc. 5th International Central and Eastern European Conference on Multi-Agent Systems.- Leipzig (Germany): Springer-Verlag, 2017.- Pp. 183–192.

5. Müller J., Fisher K. Application Impact of Multi-agent Systems and Technologies: A Survey // Agent-oriented software engineering. 2014. pp. 27–53.
6. J.-P. Briot, C. Felicissimo and C. J. de Lucena, «A Norm-Based Approach for the Modeling of Open Multiagent Systems,» in Int. Conf. on Agents and Artificial Intelligence ICAART, Porto, 2009, pp. 540-546.
7. Rahimi Shahram. A multi-agent framework for stock trading / Shahram Rahimi, Raju Tatikunta, Raheel Ahmad, Bidyut Gupta // International Journal of Intelligent Information and Database Systems 2009. – Vol. 3, No. 2 pp. 203 – 227.
8. Mukhin, V., Kuchuk, N., Kosenko, N., Kuchuk, H. and Kosenko, V. (2020), "Decomposition Method for Synthesizing the Computer System Architecture, Advances in Intelligent Systems and Computing", AISC, vol. 938, pp 289-300, DOI: [https://doi.org/10.1007/978-3-030-16621-2\\_27](https://doi.org/10.1007/978-3-030-16621-2_27).
9. Kuchuk N. Method for calculating of R-learning traffic peakedness / N. Kuchuk; O. Mozhaiev, M. Mozhaiev; H. Kuchuk // 2017 4th International Scientific-Practical Conference Problems of Infocommunications Science and Technology, PIC S and T 2017. – 2017. – R. 359 – 362. URL: <http://dx.doi.org/10.1109/INFOCOMMST.2017.8246416>.
10. Leitao P., Vrba P. Recent Developments and Future Trends of Industrial Agents // Holonic and Multi-Agent Systems for Manufacturing. 2011. LNCS 6867. pp. 15–28.
11. J.-P. Briot, C. Felicissimo and C. J. de Lucena, «A Norm-Based Approach for the Modeling of Open Multiagent Systems,» in Int. Conf. on Agents and Artificial Intelligence ICAART, Porto, 2009, pp. 540-546.
12. O. I. Denesiak i Ye. A. Palamarchuk, «KOMPLEKSNA SYSTEMA PROKTORYNHU V INFORMATSIYNYKH TEKHOLOGIYAKH ANALIZU KONTEKSTU V SYSTEMAKH OTSNIUVANNIA ZNAN», Visnyk VPI, vyp. 6, s. 93–99, Hrud. 2021
13. Katrenko A.V. Systemnyi analiz ob'ektiv ta protsesiv kompiuteryzatsii: Navchalnyi posibnyk / A.V. Katrenko. – Lviv: Novyi svit, 2003. – 424 s.
14. V. M. Pavlenko i Yu. Yu. Svirina, «Mulyahentnyi pidkhid pry pobudovi systemy tekhnichnoho obsluhovuvannia i remontu avtomobilia,» Novitni tekhnologii rozvytku avtomobilnoho transportu, 2018, s. 187.