

УДК 004.94:330.342

DOI: 10.31673/2786-8362.2024.026791

Вишнівський В.В., д.т.н.; Кравчук П.О., PhD;
Гуменний Д.О., к.т.н.; Чичур А.І., к.т.н.;
Волуйко І.В.

АГЕНТНО-ОРІЄНТОВАНЕ ТА СИСТЕМНО-ДИНАМІЧНЕ МОДЕЛЮВАННЯ В УПРАВЛІННІ СКЛАДНИМИ СИСТЕМАМИ

Vyshnivskiy V.V., Kravchuk P.O., Humennyi D.O., Chychur A.I., Volyuko I.V. **Agent-oriented and system-dynamic modeling in the management of complex systems.** The article is devoted to the creation of agent-oriented and system-dynamic models, and means of developing compatible problem solvers of intelligent systems capable of solving complex problems. The main problem considered in the work is the low consistency of the principles underlying the implementation of various models of problem solving. As a result, it is significantly difficult to simultaneously use different models of problem solving in a single system during the execution of the same complex task, it is practically impossible to reuse technical solutions implemented in any system, in addition, there are actually no complex methods and tools capable of ensure the process of developing problem solvers at all stages. It is proposed as a basis for solving the problem of the compatibility of problem solvers to implement a multi-agent approach, and to divide the process of solving any problem into logically atomic actions that will ensure the compatibility and modifiability of the solvers. It is proposed to consider the solver as a hierarchical system consisting of several interconnected levels, which allows for the possibility of independent design, debugging and verification of components of different levels. The proposed models, methods and tools were used in the development of a number of protot prescription product types of intelligent educational systems, as well as a prototype of the automation system of on.

Keywords: automated design system, modeling, task solver, intelligent agent, database

Вишнівський В.В., Кравчук П.О., Гуменний Д.О., Чичур А.І., Волуйко І.В. **Агентно-орієнтоване та системно-динамічне моделювання в управлінні складними системами.** У статті запропоновано використовувати мультиагентний підхід для вирішення проблеми сумісності розв'язувачів завдань. Рекомендовано розглядати вирішувач як ієрархічну систему, яка складається з кількох взаємопов'язаних рівнів, що дозволяє забезпечити можливість незалежного проектування, налагодження та верифікації компонентів різних рівнів. Запропоновані моделі, методики та засоби використані при розробці ряду прототипів інтелектуальних систем навчального призначення, а також прототипу системи автоматизації рецептурного виробництва.

Ключові слова: система автоматизованого проектування, моделювання, вирішувач завдань, інтелектуальний агент, база даних

Вступ.

Сучасні технологічні виклики в галузі управління складними системами вимагають від інженерів не тільки глибоких теоретичних знань, але й уміння ефективно застосовувати їх у реальних сценаріях. У цьому контексті комп'ютерне моделювання стає ключовим інструментом для навчання, яке спеціалізується на складних системах управління. У світі цифрових технологій імітаційне моделювання допомагає досліджувати та вирішувати різноманітні завдання шляхом скорочення часу та спрощення самого процесу роботи. Розробники з різних країн створюють програмні продукти, орієнтовані на вирішення постановлених перед ними завдань і надання допомоги професіоналам різних галузей.

Постановка проблеми. Одним із ключових компонентів інтелектуальної системи (ІС) є вирішувач завдань (ВЗ), що забезпечує можливість вирішувати різні завдання, пов'язані як безпосередньо з основною функціональністю системи, так і з забезпеченням ефективності роботи та автоматизації розвитку самої системи. Можливість використання різних моделей для вирішення завдань у межах однієї системи дозволить декомпозувати комплексне завдання на підзавдання, кожне з яких, у свою чергу, можна вирішити одним із відомих системі способів. Завдяки комбінації різних моделей число класів вирішуваних системою завдань буде значно більшим, ніж сумарне число класів задач, розв'язуваних декількома системами, кожна з яких реалізує лише одну з інтегрованих моделей.

Сучасні ІС, орієнтовані на тимчасове використання різних видів знань та різних моделей вирішення завдань, будуються за принципом гібридних комп'ютерних систем [1].

Аналіз останніх досліджень. Методи комп'ютерного моделювання репрезентують широкий спектр інструментів для ефективного навчання у галузі складних систем управління. Математичні моделі відіграють ключову роль у цьому процесі, оскільки забезпечують формалізацію динаміки системи. Дж. Бенкс та Дж.С. Карсон у своїй роботі «Discrete-Event System Simulation» [2] підкреслюють значення математичного моделювання для аналізу дискретних подій систем. Згідно з [3] імітаційне моделювання є ефективним методом, що дозволяє створювати віртуальні сценарії та реалістичні умови роботи системи в реальному часі. Цей підхід активно застосовується у навчальних програмах, оскільки учні можуть взаємодіяти з віртуальними системами, покращуючи розуміння динамічних процесів.

Сучасні інженерні програми, такі як Simulink і LabVIEW, дають можливість створювати складні симуляції. М. Стефанович та ін. [4] розглядають застосування LabVIEW у процесі навчання та виконання завдань. Важливе значення у комп'ютерному моделюванні має візуалізація даних. Л. Чен та С. Лі у статті «Adaptive E-Learning System for Control Engineering Education Based on Item Response Theory» [5] підкреслюють значення візуалізації для поглибленого розуміння результатів моделювання. Усі ці методи вимагають не лише технічних навичок у галузі комп'ютерного моделювання, але й глибокого розуміння можливості їх застосування у контексті управління складними системами.

М. Гарсія та Г. Лопес [6] акцентують увагу на ефективності використання математичних моделей для аналізу та проектування складних систем управління. Ще одним значущим аспектом у комп'ютерному моделюванні є імітаційне моделювання. Р. Суарес та Н. Бермудес [7] досліджують застосування імітаційних моделей для створення віртуальних сценаріїв, що дає можливість студентам взаємодіяти з системою в режимі реального часу, покращуючи їх практичні навички і розуміння функціонування систем управління. Спеціалізовані програмні середовища, такі як Simulink та LabVIEW, дають потужні інструменти створення детальних симуляцій [8]. Л. Чен та С. Лі [9] у своєму дослідженні розглядають обмеження таких програмних середовищ у контексті навчання управлінню складними системами. Д. Сміт та Т. Браун [10] підкреслюють важливість візуалізації даних для полегшення розуміння структури та динаміки складних систем управління. Огляд літератури підтверджує, що ефективно комп'ютерне моделювання у процесі навчання системам управління вимагає комплексного підходу, що інтегрує різні методи, включаючи агентно-орієнтовані та системно-динамічні моделі.

У статті будуть розглянуті перспективи застосування комп'ютерного моделювання складних систем, зокрема агентно-орієнтованих та системно-динамічних моделей. Тому необхідно дослідити сучасні підходи до комп'ютерного моделювання таких систем.

Метою роботи є підвищення ефективності комп'ютерного моделювання складних систем, включаючи агентно-орієнтовані та системно-динамічні моделі.

Виклад основного матеріалу дослідження.

Кінець ХХ ст. характеризуються стрімким розвитком комп'ютерної техніки та комп'ютерних технологій, їх поширенням на сферу проєктних робіт у різних галузях економіки. Ускладнення проєктних задач вимагало комплексного підходу до процесу проєктування і створення складних комп'ютерних систем. Перші системи уподібнювалися автоматизованій роботі інженера-конструктора. Тільки замість кульмана, паперу, рейсшини та олівця використовувалися екран монітора, мишка та відповідні інструменти графічної системи. З плином часу процеси автоматизованого проєктування, включаючи агентно-орієнтовані та системно-динамічні моделі, проникли у всі сфери людського життя: будівництво, машинобудування, радіоелектроніку, технологічні процеси, програмні продукти тощо. Розглянемо основні програмні продукти.

Система AnyLogic – універсальна система імітаційного моделювання, яка пропонує процесно-орієнтовані (дискретно-подійні), системно-динамічні та агентно-орієнтовані

технології. Її бібліотека, інструменти та графічний інтерфейс дозволяють створювати імітаційні моделі для вирішення цілого ряду завдань: від бізнес-процесів та логістики до виробництва і розвитку ринку. AnyLogic широко використовується в освіті та у транснаціональних компаніях для бізнес-моделювання.

Actor Pilgrim – спеціалізована система для створення та налагодження імітаційних моделей трудомістких процесів, включаючи часову, просторову та економічну динаміку. Використовується для аналізу проєктів у галузі енергетики, регіональної економіки, транспорту та логістики, цивільної оборони та бізнесу. Система дозволяє працювати з багатошаровими імітаційними моделями, у яких вузли представляють процеси, а актори – динамічні одиниці, що виконують функції, аналогічні транзакції GPSS. Система реалізує два типи зорієнтованих дуг для міграції суб'єктів та транзакцій моделей, а також компактну та ресурсомістку геоінформаційну підсистему [11].

Aimsun 6.0 – комплексний набір інструментів для аналізу транспортних потоків та транспортування, що полегшує планування, моделювання та дослідження транспортної діяльності. Програмне забезпечення може імпортувати та обробляти інформацію з різних геоінформаційних систем, включаючи ESRI, Tele Atlas, NAVTEQ та ін. Aimsun 6.0 підтримує формати даних з інших програм, таких як EMME/2, CONTRAM, SATURN, TRANSYT-7F, TRANSYT/12 та VSPLUS, спрощуючи завдання редагування та подання проєктної документації. Aimsun 6.0 гарантує завантаження з датчиків інформації, що зберігається в базах даних, або отримана в режимі реального часу для цілей моделювання, планування чи візуалізації.

Enterprise Dynamics – основна програмна платформа для моделювання бізнес-операцій, що надає засоби для аналізу та оптимізації поточного і майбутнього функціонування будь-якої системи чи інфраструктури з урахуванням конкретних матеріалів, обладнання та інших ресурсів, задіяних у різних галузях промисловості. У співпраці з фахівцями ринку INCONTROL розроблено бібліотеку конкретних функціональних об'єктів, які можна використовувати для різних цілей. Бібліотека включає такі продукти: ED Logistics, ED Plato, ED Airport, ED Transport, ED Warehouse, ED Educational та ShowFlow. Один з продуктів, ED Airport, є інтегрованим програмним двигуном моделювання, який розроблений спеціально для аеропортів. Він використовує достовірні та точні дані для моделювання довгострокових подій, виявлення вузьких місць, оцінки кадрового потенціалу, прогнозування потенційних результатів.

Flexsim – програмна система, яка моделює та візуалізує бізнес-процеси. Використовуючи Flexsim, можна аналізувати ефективність виробництва, виявляти вузькі місця та тестувати нові методи планування. Система генерує 3D-анімацію в кожній імітаційній моделі. Flexsim також дозволяє користувачам створювати моделі та підмоделі за допомогою мови програмування C++, використовуючи бібліотеки моделювання та графіки.

NetLogo – програмний інструмент, призначений для моделювання природних та соціальних явищ, що особливо корисно при моделюванні складних систем, які змінюються в часі. Дозволяє моделювати сценарії, у яких сотні чи тисячі незалежних «агентів» працюють паралельно. NetLogo – програмне забезпечення з відкритим вихідним кодом, яке підтримує Національний науковий фонд США. Працює на кількох платформах.

Підсумки аналізу програмних продуктів для створення імітаційних та агентно-орієнтованих і системно-динамічних моделей у різних сферах наведені в таблиці 1.

Математичне моделювання складних систем, включаючи агентно-орієнтовані та системно-динамічні моделі. Комп'ютерне моделювання є ефективним інструментом в складних процесах, що збагачує їх практичним досвідом і дозволяє глибше зрозуміти складні концепції управління системами [12]. Математичні моделі, імітаційне моделювання та спеціалізовані програмні середовища створюють реалістичні симуляції, сприяючи формуванню у майбутніх інженерів практичних навичок. Зростання системи можна визначити за рівнянням 1.

Таблиця 1

Порівняльний аналіз програмних продуктів для створення агентно-орієнтованих та системно-динамічних моделей

Програмні продукти	Переваги	Недоліки
AnyLogic	Середовище багатопідходного моделювання. Анімація та візуалізація. Галузеві бібліотеки. Професійний інструмент для агентного моделювання. Інтеграція з ГІС-картами. Запуск моделей у хмарі. Великий вибір експериментів. Робота з даними. Експорт та інтеграція моделей. Розширювана та настроювана платформа.	Існуючі шейп-файли можуть бути не актуальні.
Actor Pilgrim	Стандартні засоби системи для спостереження та аналізу експериментатором часової, просторової, фінансової динаміки. Наявність спеціалізованих бібліотек.	Недостатній обсяг інформації у загальному доступі.
Aimsun	Повнофункціональний комплекс інструментів аналізу у сфері транспорту. Дозволяє на етапі планування будівельних проєктів моделювати міські перехрестя будь-якої складності та типу регулювання. Дозволяє аналізувати пропускну здатність транспортних систем та тестувати схеми транзитних пріоритетів.	Недостатній обсяг інформації у загальному доступі.
Enterprise Dynamics	Віртуальна оптимізація. Можливість управляти великими системами. Планування інвестицій.	Недостатній обсяг інформації у загальному доступі.
Flexsim	Підтримує три технології створення імітаційних моделей: процесно-орієнтовану (дискретно-подієву), системно динамічну та агрегатну. Зручний графічний інтерфейс, інструменти та бібліотеки. Наочність та точність.	Неможливість отримання стандартних відхилень часових показників модельованих об'єктів.
NetLogo	Ефективне моделювання складних систем. Унаочнення процесів, вимірювань та розрахунків у вигляді графіків. Наявність спеціалізованих бібліотек елементів моделювання та готових моделей у різних предметних областях.	Неможливість побудови імітаційних моделей за їх структурними моделями. Необхідність програмування.

$$DR = \sum \left\{ \begin{array}{l} g = \frac{(\alpha_{12} \times \alpha_{21})^2}{\sqrt{2}} \\ DR_i(\varepsilon + 1) = \frac{\alpha_i DR_i(\varepsilon)}{1 + \sum_{i=1}^2 \alpha_{ij} DR_i(\varepsilon)} \end{array} \right\} (1)$$

Де: DR – зростання систем моделювання, DR_i – зростання i -тої системи, α – кут зростання системи.

Заходи з розвитку системного рівня комп'ютерного моделювання в межах складної системи зображені на рисунку 1.

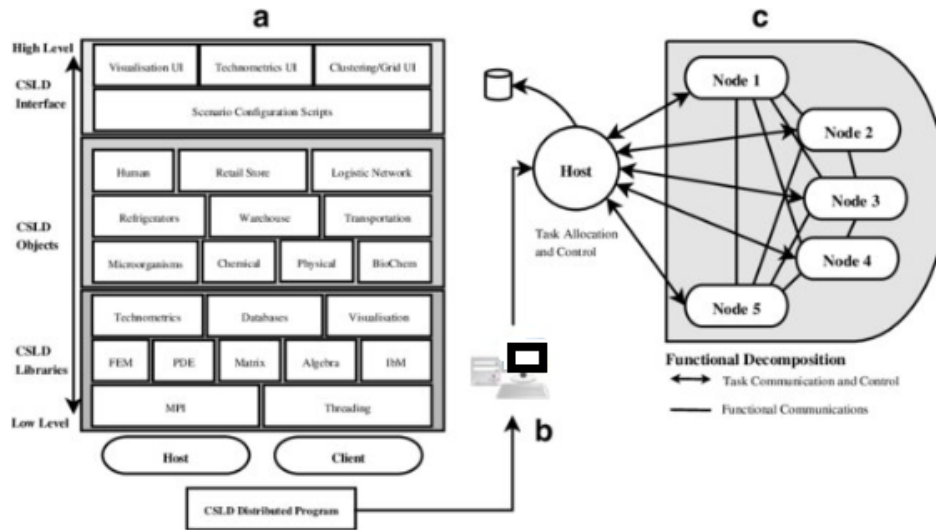


Рис. 1. Архітектура прототипу системи моделювання складних систем: а) багатомасштабна організація класів, б) доступ до кластера/сітки, с) розподілена функціональна декомпозиція модифіковано з [13]

Відповідність між правилом рішення агентів у складній системі та прямим проходом AgentNet. Є на обох панелях змінна стану кожного агента s_i^t взаємодіє зі змінними стану інших агентів із силою взаємодії a_{ij}^t . На відміну від GAT, AgentNet декодує кожну змінну окремо, щоб накласти інформацію про силу взаємодії змінних. Інші функції, а саме h_{self} і f , можуть бути захоплені як модулями кодувальника, так і модулями декодера [14].

Якщо x_i і y_{ij} демонструють відсоткове зростання, то зниження i -х змінних сприятиме інтеграції системи, відповідно y_{ij} означатиме значну частку i -го фактора, на який безпосередньо впливає j -й фактор, але також $i = 1$ і $j = 2$ представляють рівні підсистем. Тоді базове рівняння (2) можна подати у варіаційному численні:

$$DR = \sum \left\{ \begin{array}{l} \frac{mn_1}{m\epsilon} = x_1 DR_1 - y_{11} DR_1^2 - y_{12} DR_1 DR_2 \\ \frac{mn_2}{m\epsilon} = x_2 DR_2 - y_{22} DR_2^2 - y_{21} DR_1 DR_2 \end{array} \right\} (2)$$

Спрощення в одне рівняння:

$$DR = \sum \frac{m(DR_i)}{m\epsilon} = DR_i(c_i - y_{ij} DR_j) - \sum h_{ij} DR_i DR_j (3)$$

Категорія конкуренції між підсистемами визначається, виходячи з того, чи y_{ij} і y_{ji} більші за 0:

- (1) Якщо y_{ij} та y_{ji} більші за нуль, то i^{th} та j^{th} змінні є справжніми суперниками. Зростання одного гальмується зростанням іншого.
- (2) Якщо y_{ij} і y_{ji} менші за нуль, то дві змінні сприяють зростанню одна одної.
- (3) Якщо y_{ij} і y_{ji} дорівнюють нулю, то два множники не впливають один на одного.
- (4) Якщо $y_{ij} = y_{ji}$, відповідно, i^{th} -фактор пригнічує прогресію j^{th} -фактора.
- (5) Якщо $y_{ij} > 0$ і $y_{ji} > 0$, то існує аменсалізм між двома змінними.

З погляду засобів синхронізації агентно-орієнтованих та системно-динамічних моделей можна виділити три класи sc-агентів:

– sc-агенти інтерпретації scp-програм, реалізовані на рівні платформи інтерпретації sc-моделей, одним із завдань яких є забезпечення механізму синхронізації; принципи

синхронізації агентів цього класу більш тривіальні, ніж у випадку програмних sc-агентів, і описуються окремо;

- програмні sc-агенти, забезпечують основну функціональність системи, що дає можливість вирішувати ті чи інші завдання;

- sc-метаагенти, завданням яких є координація діяльності програмних sc-агентів, зокрема, для вирішення проблеми взаємоблокувань.

Така ієрархія рівнів забезпечує можливість, по-перше, поетапного проектування з поступовим підвищенням ступеня деталізації від верхнього рівня до нижнього, а, по-друге, проектування, налагодження та верифікацію компонентів на різних рівнях, що суттєво спрощує завдання побудови та модифікації складних комп'ютерних систем. Крім того, запропонований підхід до побудови агентно-орієнтованих та системно-динамічних моделей дозволяє забезпечити їх модифікованість та можливість узгодженого використання [15].

Висновки

У статті представлено низку програмних продуктів, що дозволяють користувачам створювати імітаційні, агентно-орієнтовані та системно-динамічні моделі, що застосовуються у різних сферах діяльності. Кожна система має свої переваги та недоліки. Для них регулярно випускають нові версії з розширеними функціями для різних платформ та форматів, а також нові системи моделювання, що відповідають сучасним вимогам. Основною особливістю розглянутих засобів побудови та модифікації таких систем є їх реалізація на основі комп'ютерних технологій. Ця особливість дозволяє забезпечити модифікованість самих засобів, тобто легкість нарощування їх функціональності, в тому числі за рахунок використання компонентів з постійної бібліотеки, що поповнюється.

Запропоновано:

- використовувати мультиагентний підхід як основу для вирішення проблеми сумісності розв'язувачів завдань, а процес розв'язання будь-якого завдання поділяти на логічно атомарні дії, які дозволять забезпечити сумісність і модифікованість вирішувачів;

- розглядати вирішувач як ієрархічну систему, яка складається з кількох взаємопов'язаних рівнів, що дозволяє забезпечити можливість незалежного проектування, налагодження та верифікації компонентів різних рівнів.

Запропоновані моделі, методики та засоби використані при розробці ряду прототипів інтелектуальних систем навчального призначення, а також прототипу системи автоматизації рецептурного виробництва. В майбутньому практично моделювання перетворюється на програмування, тільки замість коду використовуються фізичні параметри середовища, самого об'єкта моделювання та вимог до його експлуатації.

Список використаної літератури:

1. Бідніченко О. Г. Сучасні тенденції розвитку систем автоматизованого комп'ютерного моделювання. *Управління розвитком складних систем*. 2022. № 49. С. 59-65. URL: <https://doi.org/10.32347/2412-9933.2022.49.59-65>.
2. Discrete-event System Simulation / J. Banks et al. 5th ed. Pearson, 2009. 648 p.
3. Kim S., Chen D. Virtual Reality Simulations in Control Systems Education: A Case Study. *IEEE Transactions on Education*. 2019. Vol. 62(4). P. 309-316.
4. A Labview-based Remote Laboratory Experiments for Control Engineering Education / M. Stefanovic et al. *Computer Applications in Engineering Education*. 2011. Vol. 19 (3). P. 538-549. URL: <https://doi.org/10.1002/cae.20334>.
5. Chen L., Li S. Adaptive E-Learning System for Control Engineering Education Based on Item Response Theory. *IEEE Access*. 2019. Vol. 7. P. 60745-60754.
6. Garcia M., Lopez G. Evaluating the Quality of Computer Models in Control Systems Engineering Education. *Journal of Engineering Education*. 2020. Vol. 109(1). P.48-69.
7. Suarez R., Bermudez N. Challenges in Real-Time Simulation for Engineering Education. *International Journal of Electrical Engineering Education*. 2018. Vol. 55(3). P. 237-253.

8. Kim C., Chen J. Virtual Reality Applications in Engineering Education: A Review. *Journal of Professional Issues in Engineering Education and Practice*. 2019. Vol. 145(1). P. 04018014.
9. Chen L., Li S. Adaptive E-Learning System for Control Engineering Education Based on Item Response Theory. *IEEE Access*. 2019. Vol. 7. P. 60745-60754.
10. Smith D., Brown T. Artificial Intelligence in Control Systems Education: A Review and Case Study. *IEEE Transactions on Education*. 2021. Vol. 64(1). P. 26-36. URL: <https://doi.org/10.3389/fsufs.2021.756584/>
11. Actor Pilgrim system. URL: <http://simulation.su/static/actor-pilgrim-full-info.html> (date of access 13.10.2024).
12. Three-dimensional Trajectories and Network Analyses of Group Behaviour Within Chimney Swift Flocks During Approaches to the Roost / D. J. Evangelista et al. *Royal Society B: Biological Sciences*. 2017. Vol. 284, no 1849. URL: <https://doi.org/10.1098/rspb.2016.2602>.
13. Williams R. J., Zipser D. A Learning Algorithm for Continually Running Fully Recurrent Neural Networks. *Neural computation*. 1989. Vol. 1. P. 270-280.
14. Hoshen Y. Vain: Attentional Multi-agent Predictive Modeling. *31st Conference on Neural Information Processing Systems (NIPS 2017)*, Long Beach, CA, USA. 2017. P. 2701-2711.
15. Ha S., Jeong H. Uncovering Hidden Interactions in Complex Systems Using Deep Learning. *Scientific Reports*. 2021. No 12804. URL: <https://doi.org/10.1038/s41598-021-91878-w>.

Автори статті

Вишнівський Віктор – доктор технічних наук, професор, Державний університет інформаційно-комунікаційних технологій, Київ, Україна.

ORCID: 0000-0003-1923-4344

Кравчук Петро – PhD, доцент, Державний університет інформаційно-комунікаційних технологій, Київ, Україна.

ORCID: 0009-0000-6412-1010

Гуменний Дмитро – кандидат технічних наук, доцент, Київський національний університет будівництва і архітектури, Київ Україна.

ORCID: 0000-0001-6736-0543

Чичур Андрій – кандидат технічних наук, доцент, Державний університет інформаційно-комунікаційних технологій, Київ, Україна.

ORCID: 0009-0004-9949-0958

Волюйко Ігор – студент, Державний університет інформаційно-комунікаційних технологій, Київ, Україна.

ORCID: 0009-0003-3035-2840

Authors of the article

Vyshnivskiy Viktor – Doctor of Science (technic), Professor, State University of Information and Communication Technologies, Kyiv, Ukraine.

ORCID: 0000-0003-1923-4344

Kravchuk Petro – PhD, Associate Professor, State University of Information and Communication Technologies, Kyiv, Ukraine.

ORCID: 0009-0000-6412-1010

Humennyi Dmytro – Candidate of Science (technic), Associate Professor, Kyiv National University of Construction and Architecture, Kyiv, Ukraine.

ORCID: 0000-0001-6736-0543

Chychur Andrii – Candidate of Science (technic), Associate Professor, State University of Information and Communication Technologies, Kyiv, Ukraine.

ORCID: 0009-0004-9949-0958

Volyuko Ihor – student, State University of Information and Communication Technologies, Kyiv, Ukraine.

ORCID: 0009-0003-3035-2840