

А. О. КАРГІН

доктор технічних наук, професор,
професор кафедри комп'ютерних наук
ДВНЗ «Приазовський державний технічний університет»
ORCID: 0000-0003-2885-9071

О. І. ПРОНІНА

кандидат технічних наук, доцент,
доцент кафедри комп'ютерних наук
ДВНЗ «Приазовський державний технічний університет»
ORCID: 0000-0001-7085-8027

О. Є. П'ЯТИКОП

кандидат технічних наук, доцент,
доцент кафедри комп'ютерних наук
ДВНЗ «Приазовський державний технічний університет»
ORCID: 0000-0002-7731-3051

Є. А. ПЕРЦЕВ

магістр кафедри комп'ютерних наук
ДВНЗ «Приазовський державний технічний університет»
ORCID: 0009-0009-4155-9031

ТЕХНОЛОГІЯ ПРОЕКТУВАННЯ ШТУЧНОГО ІНТЕЛЕКТУ, ЩО ПІДТРИМУЄ АВТОНОМНІ РОЗПОДІЛЕНІ СИСТЕМИ

Потреба в більш досконалих автономних системах підтверджується зростаючим попитом не лише військової сфери, а й цивільної спільноти. Зусилля розробників автономних систем на базі Штучного Інтелекту (ШІ) зосереджені на підвищенні рівня автономності майбутніх систем. Відомі моделі штучного інтелекту не можуть комплексно вирішити проблему автономності через розрив між двома парадигмами «обробка даних від датчиків» та «прийняття рішень на основі знань експерта в предметній області». В роботі обговорюється нова модель ШІ (Feeling AI, FAI), як гібридний ШІ, що базується на когнітивних моделях, запозичених у живих істот. Так як База Знань (БЗ) FAI представлена набором незалежних універсальних Гранул Знань (ГЗ), що взаємодіють за моделлю подій, то подієво-керована архітектура (Event-Driven Architecture, EDA) є найбільш придатною для реалізації обчислювань у FAI. Основною проблемою застосування EDA є необхідність формування реакції в режимі реального часу компонентами FAI на основі сенсорних даних, отриманих асинхронно у різний час і з різними характеристиками старіння. У FAI використана нова когнітивна модель врахування старіння даних, яка підтримує поступове забування подій з часом та оцінює їхню поточну актуальність за допомогою нечіткого фактору впевненості (Certainty Factor, CF). Проектування системи сприйняття FAI за технологією EDA розглядається на прикладі обслуговування автономної системи розосереджених об'єктів (automatic Garage Complex, GC). До складу GC входить колісний робот що отримує додатково до стаціонарної СС дані щодо стану окремого боксу. Наведено окремі етапи логічного проектування, починаючи з експертних знань, їх формалізації у вигляді множини ГЗ, відображення структури знань у БЗ на апаратне забезпечення, та уявлення знань, якими оперує Event Broker EDA у таблиці параметрів. На етапі фізичного проектування розроблено програмну реалізацію ядра FAI мовою C++ з використанням бібліотеки ZeroMQ для забезпечення високопродуктивного асинхронного обміну повідомленнями. Запропонована архітектура реалізує механізм диспетчеризації подій через компоненти-маршрутизатори та обчислювальні потоки, що дозволяє масштабувати систему шляхом розпаралелювання виконання функцій обробки гранул знань.

Ключові слова: штучний інтелект, когнітивна модель старіння даних, автономні системи, нечіткий фактор впевненості, гранула знань, подієво-керована архітектура.

A. O. KARGIN

Doctor of Technical Sciences, Professor,
Professor at the Department of Computer Science
PSTU "Pryazovskyi State Technical University"
ORCID: 0000-0003-2885-9071

O. I. PRONINA

Candidate of Technical Sciences, Associate Professor,
Associate Professor at the Department of Computer Science
PSTU "Pryazovskyi State Technical University"
ORCID: 0000-0001-7085-8027

O. YE. PIATYKOP

Candidate of Technical Sciences, Associate Professor,
Associate Professor at the Department of Computer Science
PSTU "Pryazovskyi State Technical University"
ORCID: 0000-0002-7731-3051

YE. A. PERTSEV

Master at the Department of Computer Science
PSTU "Pryazovskyi State Technical University"
ORCID: 0009-0009-4155-9031

DESIGN TECHNOLOGY OF THE ARTIFICIAL INTELLIGENCE THAT SUPPORTS AUTONOMOUS DISTRIBUTED SYSTEMS

The need for more advanced autonomous systems is confirmed by the growing demand not only from the military domain but also from the civilian community. The efforts of developers of autonomous systems based on Artificial Intelligence (AI) are focused on increasing the level of autonomy of future systems. Known AI models cannot comprehensively solve the problem of autonomy due to the gap between the two paradigms of "processing data from sensors" and "making decisions based on expert knowledge in verbal form." The paper discusses a new model of AI, known as Feeling AI (FAI), which is a hybrid AI based on cognitive models borrowed from living beings. Since the FAI's Knowledge Base (KB) is represented by a set of independent universal Knowledge Granules (KG) that interact according to the event model, the Event-Driven Architecture (EDA) is most suitable for implementing the processing in FAI. The primary challenge of using EDA is that its components must process data in real-time when they are received at different times and with varying aging characteristics. FAI employs a novel cognitive model of data aging, which supports the gradual forgetting of events over time and estimates their current relevance using a fuzzy Certainty Factor (CF). The design of an FAI's perception system, based on EDA technology, is considered using the example of servicing an autonomous system of distributed objects, such as the Automatic Garage Complex (AGC). The AGC includes a wheeled robot that receives data on the state of a separate box in addition to the stationary perception system. Separate stages of logical design are presented, starting with expert knowledge, its formalization in the form of a set of KGs of the KB, mapping the structure of the KB into hardware deployment, and representing the knowledge that the Event Broker EDA operates with in the parameter table. At the physical design stage, a software implementation of the FAI core was developed in C++ using the ZeroMQ library to provide high-performance asynchronous messaging. The proposed architecture implements an event dispatching mechanism through router components and computational threads, which allows the system to be scaled by parallelizing the execution of the knowledge granule processing function.

Key words: artificial intelligence, cognitive data aging model, autonomous systems, fuzzy certainty factor, knowledge granule, event-driven architecture.

Постановка проблеми

Один із напрямів використання інтелектуальних інформаційних технологій пов'язаний з автономними системами [1, 2, 3]. Особливо актуальними є системи, в яких завдяки ШІ досягається безпека об'єктів. Сучасний рівень розвитку Інтернету Речей і ШІ сприяє їх інтеграції при створенні автономних розосереджених об'єктів. Централізований збір всіх даних не завжди є ефективним рішенням для певного класу об'єктів, тому застосовуються гібридні системи де окремі компоненти сенсорної підсистеми можуть бути реалізовані за допомогою мобільних пристроїв (колісні роботи, дрони), які оснащені потрібними сенсорами для отримання даних. Головне питання що виникає при створенні такого класу систем це як об'єднати окремі компоненти системи, а точніше моделі ШІ що їх підтримують, у цілісну однорідну структуру знань [4]. Створення ШІ для автономних технічних систем у тому числі з розосередженими об'єктами які задовольняють певному рівню автономності [5] на сьогодні є актуальним завданням.

Аналіз останніх досліджень і публікацій

Сучасний стан та тенденції ШІ, які виявляються зараз (Spatial computing, Polyfunctional robots) і є такими що націлені на об'єднання фізичного та цифрового світів, створюючи нове обчислювальне середовище, де віртуальні елементи інтегровані у реальне оточення, добре підходять для автоматизації послуг, що виконуються у віртуальному світі Інтернету [6]. Для вирішення аналогічної проблеми у фізичному світі запропановано RT-X проект метою якого є створення загального робототехнічного мозку (GRB) шляхом збору даних, ресурсів та коду, навичок, яким роботи вже навчені [7, 8]. Планується, що GRB матиме підтримку GenAI [9, 10]. Такий ШІ із символічним механізмом прийняття рішень та попередньо навченим набором різних навичок у фізичному світі є одним із типів гібридного ШІ, спрямованого на масштабні роботизовані застосування. Використання цього підходу для вирішення згаданої проблеми обмежене з двох причин. По-перше, це обмежує автономність через постійне підключення до Інтернету та велику базу знань (БЗ), необхідну для охоплення всіх можливих ситуацій [11]. По-друге, на сьогодні автономні роботизовані системи не мають потужних комп'ютерних засобів [12, 13]; отже, вони не можуть виконувати програмне забезпечення гібридного ШІ класу GRB. У статтях [4, 14, 15] пропонується інша модель гібридного ШІ для робототехніки (Feeling AI, FAI), яка базується на когнітивних моделях, що запозичені у живих істот. На відміну від GRB, FAI спрямований на невеликі роботизовані системи, що підтримуються звичайним комп'ютерним обладнанням та програмним забезпеченням, які традиційно використовуються на периферійному рівні Інтернету речей та Розумних систем [16, 17, 18]. Різні питання організації та впровадження когнітивних моделей як основи FAI обговорювалися в попередніх публікаціях: сприйняття, що підтримується механізмом старіння даних [19], прийняття рішень на основі когнітивних моделей [4, 14, 15]. Оскільки архітектура FAI складається з незалежних компонентів, які взаємодіють за допомогою повідомлень один з одним, то відкривається можливість об'єднання компонентів на основі (Event-Driven Architecture, EDA) [20-22]. Теоретичні аспекти застосування EDA у FAI розглянуті в роботі [23].

Формулювання мети дослідження

Стаття присвячена дослідженню інформаційної технології проектування підсистеми сприйняття FAI для автономних систем із розосередженими об'єктами.

Викладення основного матеріалу дослідження

Теоретичним підґрунтям технології є модель FAI запозичує те, як нейронна мережа найпростіших живих істот, як біологічна форма представлення та обробки знань, формує поведінку [4, 14, 15]. БЗ живих істот представляє знання про реакції істоти на всі можливі ситуації. Для формування кожної окремої реакції використовуються три види знань: Сенсорний Шаблон (СШ), Дії Шаблон (ДШ) та Конструктор Реакцій (КР). Об'єднані разом, вони представляють Прототип Реакцій (ПР), як показано на рис. 1. СШ усіх ПР FAI образують Систему Сприйняття (СС) FAI.

Для універсального представлення всіх трьох компонентів було запроваджено модель ГЗ, яка моделює окремих нейрон живих істот. Це універсальний елементарний фрагмент знань, подібний до цеглинок у будівництві [4]. На рис. 1 ГЗ показані колами з цифрами всередині. ГЗ виконує дві ролі: представлення та обробку. Модель представлення визначає сенс ГЗ через сенс інших ГЗ нижчого рівня абстракції, використовуючи параметризовані семантичні зв'язки.

$$\langle N, know, \{ \langle M, (a, b, v, g) \rangle, M_i \in \Omega_N \} \rangle \quad (1)$$

де N – ідентифікатор ГЗ; $know$ – знакова модель ГЗ; $\Omega_N = \{ M_i \}_{i=1, 2, \dots, L}$ – набір ГЗ нижчих рівнів концептуалізації, які використовуються для визначення сенсу гранули N ; M_i – ідентифікатор ГЗ нижнього рівня; a – впевненість, b – часова затримка, v – швидкість старіння даних, g – повнота знань.

Отже, СШ – це структурований набір ГЗ (рис. 1), де ГЗ нульового рівня отримують безпосередньо дані від датчиків. ГЗ верхніх рівнів вже представляють концептуалізовані знання й отримують на вході значення впевненості і на виході також обчислена впевненість, що входні дані ГЗ відповідають шаблону, знання про який зберігає ця ГЗ. ГЗ має L входів та один вихід, значення яких є нечіткими факторами впевненості (Certainty Factor, CF): це нечітке число з гауссовою функцією належності [4].

$$m_x = \exp(-(x-a)^2/2\beta^2), \forall x \in [-q, +q] \quad (2)$$

з параметрами: $-1, 0 \leq a \leq +1, 0$ – значення впевненості; β – часовий інтервал, що минув з моменту останньої зміни значення впевненості; $|q| > +1$. Для полегшення обчислень було введено числову оцінку CF, яка виражає результат старіння даних з часом відповідно до швидкості старіння v [4, 14, 15].

$$cf = \alpha \cdot k_\beta, \quad k_\beta = 1 - v \cdot \frac{\sum_{x \in [-q, +q]} m_x(x)}{Card([-q, +q])} \quad (3)$$

Модель обробки ГЗ, використовуючи (2) та (3), обчислює оцінки того, як поточні входні CF відповідають сенсу ГЗ, представленому в (1). Отже, вихід KG такий що

$$cf_{out}(t) = f_{KG}(cf_{in_1}(t), cf_{in_2}(t), \dots, cf_{in_L}(t)). \tag{4}$$

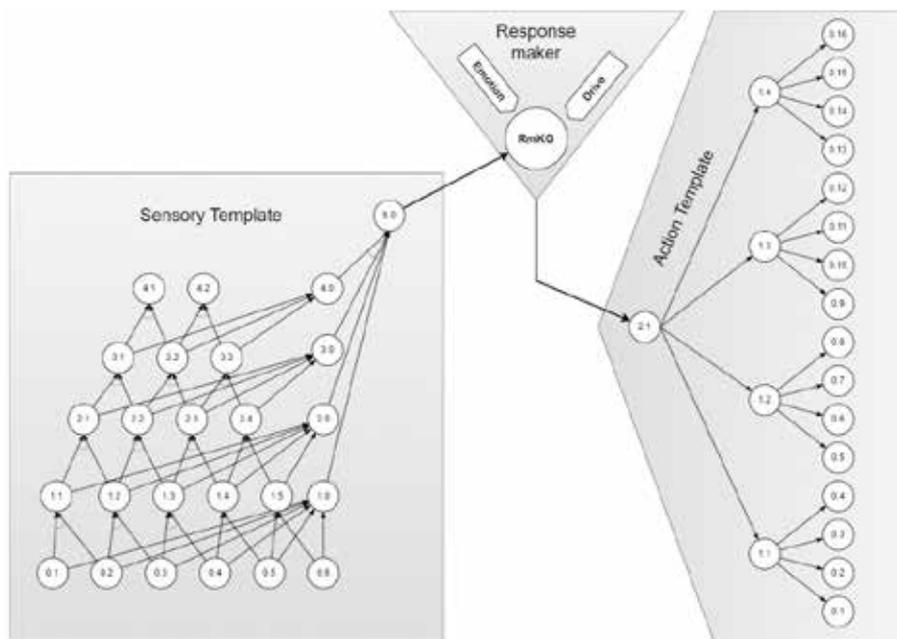


Рис. 1. Приклад структури прототипу одиночної реакції [23]

Математична модель та обчислювальний алгоритм функції f_{KG} у (4) наведено в [4].

Проектування СС FAI за технологією EDA розглянемо на прикладі системи обслуговування автоматичного (Garage Complex, GC) – одно або багато рядову лінійку гаражних боксів, кожен із яких має автоматичні ворота. Основні параметри кожного окремого боксу, такі як стан гаражних воріт, наявність автомобіля, пожежебезпека, можливо температура й вологість контролюються автоматично і ця інформація збирається й обробляється автоматичною системою GC. До складу автоматичного GC ще входить колісний робот що виконує періодичний моніторинг стану GC шляхом об'їзду й обстеження кожного боксу або виконує додаткове обстеження стану окремого боксу, коли надходить інформація про ті чи інші порушення нормального режиму функціонування його автоматичних воріт. Колісний робот оснащений набіром датчиків для контролю різних параметрів воріт (відкриті чи закриті, чи нормально відкриті і таке інше). Дані від датчиків робота, інтегровані разом із даними від датчиків кожного окремого боксу, використовуються СС FAI, яка надає модель поточної ситуації по кожному боксу GC на різних рівнях абстрагування. Поточна ситуація формується у реальному часі й використовується системою прийняття рішень FAI для формування різних команд у реальному часі системам автоматички гаражних воріт, роботу й сервісній службі. Таким чином GC уявляє собою автономну систему, обслуговування якої підтримує ІІІ, а саме, FAI. В даній роботі розглядаються питання проектування СС FAI на прикладі фрагмента бази знань необхідного для обслуговування автоматичних гаражних воріт.

Проектування СС FAI, як і іншої інформаційної системи, розбивається на два великих етапа: проектування логічної моделі СС FAI і створення фізичної моделі EDA, яка втілює логічну модель СС FAI.

На першому кроці логічного проектування експертні знання щодо реакції на різні ситуації, які можуть виникнути при обслуговуванні автоматичної GC, формуються у вигляді множини правил чи ПР

ЯКЩО *ситуація* ТОДІ *реакція*,

де *ситуація* і *реакція* представлені у вербальній формі. Завдання експертів побудувати структуру знань у вербальній формі, також, де дається визначення ситуації через набір більш деталізованих підситуацій (1) таким чином щоб досягти рівня деталізації на якому значення ситуації вже є безпосередньо дані від датчиків. Наприклад, ситуація «*There is a normal situation with the compartment gate*» це ситуація коли і «*There is a normal state of the opened gate*» і «*There is a normal state of the closed gate*». Потім визначаються ці дві ситуації, наприклад, для першої можна сказати «*Normal opened gate position received by robot's sensors*» і «*The automatic gate system indicates the open position of the door*». Таким чином послідовна деталізація опису ситуацій продовжується поки не досягається

до даних від сенсорів що описують саме ситуацію. Нижче у (5) наведено фрагмент знань, параметризованих відповідно до (1), щодо визначення ситуацій для прикладу що розглядається. Для наступних пояснень технології, структура знань (5) наведена ще у графічному вигляді на Рис. 2.

1. $\langle 5.1, \text{There is a normal situation with compartment } N \text{ } 0.4, \{ \langle 4.1, (0.75, 0, 0.1, 0.5) \rangle, \langle 0.4, (0.75, 0, 0.1, 0.5) \rangle \} \rangle$;
2. $\langle 5.2, \text{There is a malfunction in the automatic gate system of compartment } N \text{ } 0.4, \{ \langle 4.2, (0.75, 0, 0.1, 0.5) \rangle, \langle 0.4, (0.75, 0, 0.1, 0.5) \rangle \} \rangle$;
3. $\langle 4.1, \text{There is a normal situation with the compartment gate, } \{ \langle 3.1, (0.75, 0, 0.1, 1.0) \rangle, \langle 3.2, (0.75, 0, 0.1, 1.0) \rangle \} \rangle$;
4. $\langle 4.2, \text{There is a malfunction in the automatic gate system, } \{ \langle 3.3, (0.75, 0, 0.1, 1.0) \rangle, \langle 3.4, (0.75, 0, 0.1, 1.0) \rangle \} \rangle$;
5. $\langle 3.0, \text{There is a normal state of the open gate and no service personnel in the compartment, } \{ \langle 2.1, (0.75, 0, 0.1, 0.5) \rangle, \langle 1.7, (-0.75, 0, 0.1, 0.5) \rangle \} \rangle$;
6. $\langle 3.1, \text{There is a normal state of the opened gate, } \{ \langle 2.1, (0.75, 0, 0.1, 0.5) \rangle, \langle 2.3, (0.75, 0, 0.1, 0.5) \rangle \} \rangle$;
7. $\langle 3.2, \text{There is a normal state of the closed gate, } \{ \langle 2.2, (0.75, 0, 0.1, 0.5) \rangle, \langle 2.4, (0.75, 0, 0.1, 0.5) \rangle \} \rangle$;
8. $\langle 3.3, \text{There is a malfunction of the automatic gate system in the open position of the gate, } \{ \langle 2.1, (0.75, 0, 0.1, 0.25) \rangle, \langle 2.4, (0.75, 0, 0.1, 0.25) \rangle, \langle 1.6, (0.75, 0, 0.1, 0.25) \rangle, \langle 1.7, (-0.75, 0, 0.1, 0.25) \rangle \} \rangle$;
9. $\langle 3.4, \text{There is a malfunction of the automatic gate system in the closed position of the gate, } \{ \langle 2.2, (0.75, 0, 0.1, 0.33) \rangle, \langle 2.3, (0.75, 0, 0.1, 0.33) \rangle, \langle 1.6, (0.75, 0, 0.1, 0.33) \rangle \} \rangle$;
10. $\langle 2.1, \text{Normal opened gate position received by robot's sensors, } \{ \langle 1.1, (0.75, 0, 0.1, 1.0) \rangle, \langle 0.1k, (0.75, 0, 0.1, 1.0) \rangle \} \rangle$;
11. $\langle 2.2, \text{Normal closed gate position, } \{ \langle 1.3, (0.75, 0, 0.1, 0.5) \rangle, \langle 1.4, (0.75, 0, 0.1, 0.5) \rangle \} \rangle$;
12. $\langle 2.3, \text{The automatic gate system indicates the door open position, } \{ \langle 1.4, (0.75, 0, 0.1, 0.5) \rangle, \langle 1.5, (-0.75, 0, 0.1, 0.5) \rangle \} \rangle$;
13. $\langle 2.4, \text{The automatic gate system indicates the door closed position, } \{ \langle 1.4, (-0.75, 0, 0.1, 0.5) \rangle, \langle 1.5, (0.75, 0, 0.1, 0.5) \rangle \} \rangle$;
14. $\langle 1.1, \text{Gate is not skewed, } \{ \langle 0.11, (0.75, 0, 0.1, 0.5) \rangle, \langle 0.12, (0.75, 0, 0.1, 0.5) \rangle \} \rangle$;
15. $\langle 1.2, \text{The space underneath the opened gate is clear, } \{ \langle 0.1k+1, (-0.75, 0, 0.1, 0.5) \rangle, \langle 0.1n, (-0.75, 0, 0.1, 0.5) \rangle \} \rangle$;
16. $\langle 1.3, \text{The doorway of the gate is filled, } \{ \langle 0.1k+1, (0.75, 0, 0.1, 0.5) \rangle, \langle 0.1n, (0.75, 0, 0.1, 0.5) \rangle \} \rangle$;
17. $\langle 1.4, \text{The door's sensors indicate the open position of the door, } \{ \langle 0.21, (0.75, 0, 0.1, 0.5) \rangle, \langle 0.21', (-0.75, 0, 0.1, 0.5) \rangle \} \rangle$;
18. $\langle 1.5, \text{The door's sensors indicate the closed position of the door, } \{ \langle 0.22', (-0.75, 0, 0.1, 0.5) \rangle, \langle 0.22, (0.75, 0, 0.1, 0.5) \rangle \} \rangle$;
19. $\langle 1.6, \text{The automatic gate systems are powered up, } \{ \langle 0.23, (0.75, 0, 0.1, 0.5) \rangle, \langle 0.24, (0.75, 0, 0.1, 0.5) \rangle \} \rangle$;
20. $\langle 1.7, \text{There are human beings in the compartment, } \{ \langle 0.31, (-0.75, 0, 0.1, 1/k) \rangle, \dots, \langle 0.3k, (-0.75, 0, 0.1, 1/k) \rangle \} \rangle$;
21. $\langle 0.11, \text{Normal height on the right, } \{ \langle (0.0, 249.5, 250.0, 250.5, 251.0, 300.0) \rangle \} \rangle$;
22. $\langle 0.12, \text{Normal height on the left, } \{ \langle (0.0, 249.5, 250.0, 250.5, 251.0, 300.0) \rangle \} \rangle$;
23. $\langle 0.13, \text{The space underneath the gate is clear in sector } 1, \{ \langle (0.0, 0.0, 0.5, 299.5, 300.0, 300.0) \rangle \} \rangle$;
24. $\langle 0.1n+3, \text{The space underneath the gate is clear in sector } n, \{ \langle (0.0, 0.0, 0.5, 299.5, 300.0, 300.0) \rangle \} \rangle$;
25. $\langle 0.21, \text{The } 1^{\text{st}} \text{ door's sensor indicates the open position of the door, } \{ \langle (0.0, 0.0, 0.5, 0.95, 1.0, 1.0) \rangle \} \rangle$;
26. $\langle 0.22k, \text{The } k^{\text{st}} \text{ door's sensor indicates the open position of the door, } \{ \langle (0.0, 0.0, 0.5, 0.95, 1.0, 1.0) \rangle \} \rangle$;
27. $\langle 0.21k+1, \text{The } 1^{\text{st}} \text{ door's sensor indicates the closed position of the door, } \{ \langle (0.0, 0.0, 0.5, 0.95, 1.0, 1.0) \rangle \} \rangle$;
28. $\langle 0.22, \text{The } k^{\text{st}} \text{ door's sensor indicates the closed position of the door, } \{ \langle (0.0, 0.0, 0.5, 0.95, 1.0, 1.0) \rangle \} \rangle$;
29. $\langle 0.23, \text{Normal state of electricity supply of the automatic control system, } \{ \langle (0.0, 210, 215, 225, 230, 250) \rangle \} \rangle$;
30. $\langle 0.24, \text{Normal state of electricity supply of the automatic gate drives' system, } \{ \langle (0.0, 210, 215, 225, 230, 250) \rangle \} \rangle$;
31. $\langle 0.31, \text{There aren't humans being in sector } 1 \text{ of the compartment, } \{ \langle (0.0, 0.0, 0.5, 0.95, 1.0, 1.0) \rangle \} \rangle$;
32. $\langle 0.31k, \text{There aren't humans being in sector } k \text{ of the compartment, } \{ \langle (0.0, 0.0, 0.5, 0.95, 1.0, 1.0) \rangle \} \rangle$;
33. $\langle 0.4, \text{The ID of compartment, } \{ \langle (k) \rangle \} \rangle$.

(5)

В (5) в кожній строчці наведено опис однієї окремої ГЗ відповідно до (1). Перша цифра ідентифікатора ГЗ відображає рівень узагальнення ситуації. Так, цифра «0» вказує на нульовий рівень, де сенс ГЗ розкривається даними від сенсорів. Наприклад, ГЗ з ідентифікатором 0.11 має вербальне визначення «Normal height on the right» сенс якого задається чисельним діапазоном висоти нормально піднятих воріт. На етапі обробки даних чисельні показники дачика Lidar, наприклад, VL53L0X-V2 фазифікується відповідно (4) на основі знань (5) (значень параметрів в (1) які вказані в визначенні цієї ГЗ). Результат обробки сенсорних даних ГЗ є нечітка впевненість в тому, що входні сенсорні дані відповідають діапазону вказаному у визначенні гранули. Нечітка впевненість оцінюється чисельно як Fuzzy Certainty Factor (FCF) (2) чи (3). Таким чином, ГЗ нульового рівня виконують роль містка між даними від сенсорів й знаннями у вербальному вигляді й ліквідують розрив між двома парадигмами: дані від сенсорів й міркування на вербальному рівні. Завдяки цьому, ГЗ вищих рівнів узагальнення, першого, другого і так далі, отримують від ГЗ нижчих рівнів входні дані тільки у вигляді значень впевненості FCF й обчислюють на виході також впевненість, що входні дані відповідають вербальному опису наведеному у цій ГЗ. Отже, Рис. 2 показує потік даних у вигляді FCF від сенсорів до ГЗ найвищого рівня, де типові ситуації для прийняття рішень мають оцінки впевненості в тому, що наочна ситуація отримана від сенсорів відповідає їм.

На другому кроці логічного проектування отримана структура знань у вигляді (5) та Рис. 2 відображається на апаратне забезпечення автономної системи. Традиційно для цього використовують UML діаграми розміщення [24], але для більшої наочності технології, що наведено у статті наведено у вигляді Рис. 3.

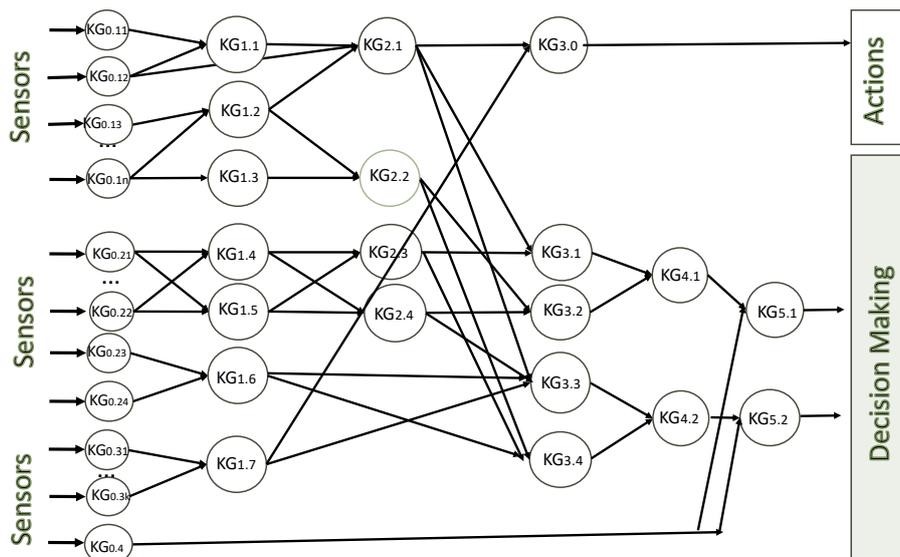


Рис. 2. Графічне зображення структури знань (5)

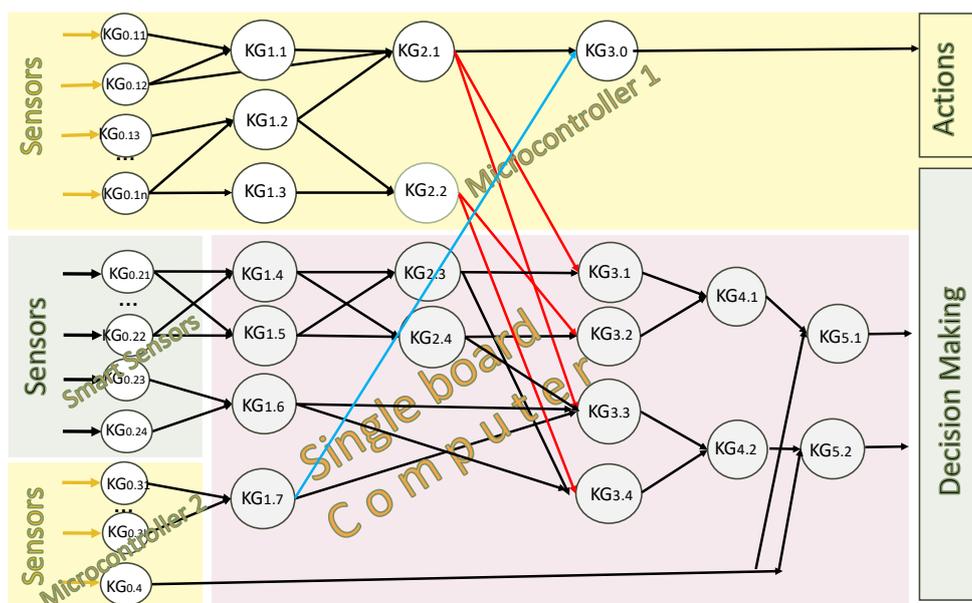


Рис. 3. Відображення структури знань (5) на апаратне забезпечення

Із рисунка видно що база знань (Рис. 2) розподілена між двома мікроконтролерами (Microcontroller 1, Microcontroller 2), чотирма розумними сенсорами, які на виході обчислюють FCF й представлені як окремі ГЗ KG0.21, ..., KG0.24, кожен з яких базується на своєму мікроконтролері, і одноплатним комп'ютером. Обидва мікроконтролера розташовані на мобільному роботі, додатково до тих що підтримують автономність функціонування робота, й обслуговують кожен свої сенсори: лазерні датчики відстані VL53L0X-V2 і матричний ІЧ датчик MLX90640 (тепловізійна камера), відповідно.

З'єднання між зовнішніми компонентами здійснюється за допомогою TCP. Робота між внутрішніми компонентами здійснюється через внутрішньопроцесорну або міжпроцесорну взаємодію в межах однієї системи. Зв'язки між ГЗ вищих, крім нульового, рівнів розташованих на різних апаратних платформах вказані червоним й блакитним кольорами на Рис. 3.

Основні знання що потрібні для прийняття рішень автономною системою зосередженні в одній компоненті, а саме, на одноплатному комп'ютері, як видно з Рис. 3. Частина знань, що потрібна на нижніх рівнях обробки даних від сенсорів, якби «приєднується» до глобальної структури знань під час коли робот зосередився на

отримані даних від конкретного гаражного боксу. Ці знання є загальними для обробки інформації від будь яких боксів. Коли робот під'їхав й спозиціонував своє положення перед воротами боксу, сканував дані від своїх сенсорів в тому числі й ідентифікатор боксу, тоді фрагмент структури знань що базується на контролері робота як би приєднується тимчасово до основної структури знань на час обробки. Це можливо завдяки однорідності уявлення знань за допомогою запропонованої моделі ГЗ, а також однорідності обробки даних від різного типу сенсорів завдяки введеному нечиткому фактору впевненості FCF і використанню одного ш того ж алгоритму обчислювання для гранул всіх рівнів. Таку технологію можна розглядати як *технологію приєднаної реальності*, яка на відміну від розширеної (віртуальної чи доповненої) реальності створює повну модель ситуації шляхом приєднання тільки реальних фрагментів фізичного світу. Це перша особливість. А друга полягає в тому, що приєднана реальність створює структуру знань для ШІ, який на цій основі реалізує когнітивну функцію міркувань, а не навпаки: ШІ створює розширену реальність для людини доповнюючи реальне сприйняття фізичного світу цифровими віддумками.

На третьому кроці логічного проектування розподілена структура знань Рис. 3 відображається у EDA. Для нашого прикладу це дві пов'язані між собою EDA. В цій статті розглядається технологія на прикладі однієї EDA для реалізації на одноплатном комп'ютері наведено бузкокольором на Рис. 3. Тобто структура (множина ГЗ і зв'язки між ними), а також параметри ГЗ (6) повинні бути представленими у вигляді компонентів EDA, як показано на рис. 4.

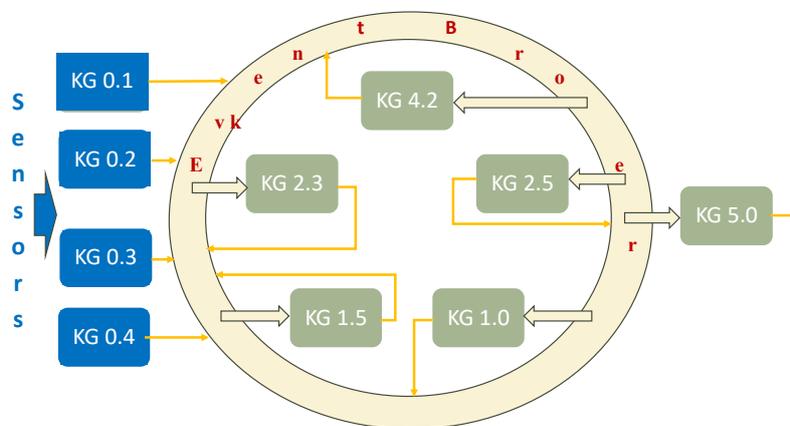


Рис. 4. Компоненти EDA

Для уявлення знань щодо структури Рис. 3, якими оперує Event Broker (EB) на Рис. 4, використовується дві таблиці (Таблиця 1 і Таблиця 2), які містять інформацію про зовнішні зв'язки EDA з ГЗ що належать іншим платформам і внутрішні зв'язки між ГЗ EDA, відповідно. Нижче в Таблиці 1 наведені всі зовнішні зв'язки. У першому стовбчику перелічені усі ГЗ, які надсилають повідомлення через EB, коли отримують свіжі вхідні дані й результат обчислення вихідного FCF задовольняє умовам що вказані в якості параметрів уявлення знань цієї ГЗ. Так як ці ГЗ є зовнішніми, то в таблиці вказано не тільки оримувачів цих повідомлень, а ще і канал через який EB отримує повідомлення. Друга Таблиця 2 містить аналогічну інформацію щодо ГЗ приналежних до розглядаємої EDA: перелічені всі ГЗ як продюсери повідомлень та список отримувачів для кожного продюсера. Крім цього, якщо отримувач не належить до EDA, тоді додатково вказуються ідентифікатор каналу через який відправляється повідомлення. Нагадамо, що повідомлення, яке отримує EB містить інформацію тільки о продюсері (ID гранули відправника й значення FCF). Тому організація даних у таблицях налаштована на такий тип EDA.

Фізичне проектування на основі створенної логічної моделі виконано на мові програмування C++. Серед існуючих бібліотек обрані ZeroMQ та STL, як ті які задовольняють вимогам створення автономних ситем без застосування Інтернет та хмарних обчислень й можуть бути розгорнутими на одноплатних комп'ютерах і контролерах. Бібліотека ZeroMQ [25] дозволяє реалізувати обмін повідомленнями, створити власну event-driven архітектуру, мало важить, і підтримується Linux подібними операційними системами. Для організації міжгранульної взаємодії застосовано набір комунікаційних протоколів, що забезпечують узгоджену та масштабовану передачу даних, зокрема, протокол iprocs. Для реалізації передачі даних у розподіленому середовищі застосовується протокол TCP, що функціонує на основі стандартного стеку TCP/IP.

Програмна реалізація системи Рис. 3 організована як узгоджена сукупність взаємодіючих чотирьох мікро EDA. Кожна з них має власний ідентифікаційний номер, набір потоків-виконавців, сховище вузлів (множина ГЗ) та маршрутизатор (EB), відповідальний за переспрямування повідомлень.

Таблиця 1
Зовнішні зв'язки EDA

Event producer		Event consumers (KG's ID/input's N)
KG's ID	External Channel	
KG021	Ch 1	KG ₁₄ /1, KG ₁₅ /1
KG022		KG ₁₄ /2, KG ₁₅ /2
KG023		KG ₁₆ /1
KG024		KG ₁₆ /2
KG031	Ch 2	KG ₁₇ /1,
KG03k		KG ₁₇ /k
KG04		KG ₅₁ /2, KG ₅₂ /2
KG21	Ch 3	KG ₃₁ /1, KG ₃₁ /1
KG22		KG ₃₂ /1, KG ₃₄ /1

Таблиця 2
Внутрішні зв'язки EDA

Event producer (KG's ID)	Event consumers (Channel:KG's ID/input's N)	
	Internal	External Channel
KG14	KG ₂₃ /1, KG ₂₄ /1	
KG15	KG ₂₃ /2, KG ₂₄ /2	
KG16	KG ₃₃ /1, KG ₃₄ /3	
KG17	KG ₃₄ /4	Ch 1:KG ₃₀ /2
KG23	KG ₃₁ /2, KG ₃₄ /2,	
KG24	KG ₃₂ /2, KG ₃₃ /2	
KG31	KG ₄₁ /1	
KG32	KG ₄₁ /2	
KG33	KG ₄₂ /1	
KG34	KG ₄₂ /2	
KG41	KG ₅₁ /1	
KG42	KG ₅₂ /1	
KG51		Ch 4
KG52		Ch 4

ЕВ, отримав повідомлення від будь якої ГЗ, використовуючі дані таблиць розсилає повідомлення відповідним отримувачам й активізує для кожної гранули-отримувача виконання алгоритму, вказаному у Таблиці 3, та передає йому відповідні параметри із цієї же таблиці. Параметри ГЗ наведені в таблиці 3.

Таблиця 3

Параметри ГЗ

KG's ID	KG's processing algorithm	KG's input parameters (c _i /a _i , b _i , v _i , g _i)				KG's output parameters (c/a)
		Input i=1	Input i=2	Input i=3	Input i=4	
1.4	2SNO	Up/0.75, 0, 0.1, 0.5	Up/0.75, 0, 0.1, 0.5			Up/0.85
1.5	2SNO	Up/0.75, 0, 0.1, 0.5	Up/0.75, 0, 0.1, 0.5			Up/0.85
1.6	2SNO	Up/0.75, 0, 0.1, 0.5	Up/0.75, 0, 0.1, 0.5			Up/0.85
1.7	2SNO	Up/0.75, 0, 0.1, 0.5	Up/0.75, 0, 0.1, 0.5			Up/0.85
2.3	2SNO	Up/0.75, 0, 0.1, 0.5	Down/-0.75, 0, 0.1, 0.5			Up/0.85
2.4	2SNO	Down/-0.75, 0, 0.1, 0.5	Up/0.75, 0, 0.1, 0.5			Up/0.85
3.1	2SNO	Up/0.75, 0, 0.1, 0.5	Up/0.75, 0, 0.1, 0.5			Up/0.85
3.2	2SNO	Up/0.75, 0, 0.1, 0.5	Up/0.75, 0, 0.1, 0.5			Up/0.85
3.3	2SNO	Up/0.75, 0, 0.1, 0.25	Up/0.75, 0, 0.1, 0.25	Up/0.75, 0, 0.1, 0.25	Down/-0.75, 0, 0.1, 0.25	Up/0.85
3.4	2SNO	Up/0.75, 0, 0.1, 0.33	Up/0.75, 0, 0.01, 0.33	Up/0.75, 0, 0.01, 0.33		Up/0.85
4.1	2SNO	Up/0.75, 0, 0.1, 1.0	Up/0.75, 0, 0.1, 1.0			Up/0.85
4.2	2SNO	Up/0.75, 0, 0.1, 1.0	Up/0.75, 0, 0.1, 1.0			Up/0.85
5.1	2SNO	Up/0.75, 0, 0.1, 0.5	Up/0.75, 0, 0.1, 0.5			Up/0.85
5.2	2SNO	Up/0.75, 0, 0.1, 0.5	Up/0.75, 0, 0.1, 0.5			Up/0.85

Програмна реалізація окремої мікро EDA включає хеш-таблиці типу std::unordered_map в оперативній пам'яті що забезпечує високу швидкодію. В них зберігаються дані вузлів: номер ГЗ, функція обробки, дані з Таблиці 3, та сигнал, який випускається при завершенні виконання з передачею повідомлення. Сховище вузлів (NodeStore) організоване на основі двох хеш-структур. Перша відповідає за швидкий доступ до екземплярів вузлів за їхніми номерами, друга фіксує односторонні зв'язки між вузлами: використовуючи номер вузла-джерела як ключ, система миттєво отримує перелік вузлів-приймачів, з якими він пов'язаний. Маршрутизатор побудований на основі ZeroMQ для прослуховування повідомлень від ГЗ. Для цього асинхронно слухається вказана черга повідомлень (наприклад, inproc://outputs). У разі надходження даних від будь-якої ГЗ він визначає відправника, звертається до NodeStore, отримує список вузлів-призначень і перенаправляє повідомлення у відповідний канал для виконання.

Потік-виконавець (Worker) використовує сокет zeromq для прослуховування повідомлень у власній черзі (наприклад, `inproc://tasks`). Після надходження повідомлення він визначає, до якого вузла воно належить, передає йому дані й запускає вказану функцію обробки. Для довготривалого збереження стану системи ще розроблено спеціалізований файловий формат `grpdf`, який містить повний набір параметрів, необхідних для відтворення системи.

Висновки

FAI, як ІШ для управління на основі сенсорних даних автономними системами, має бути розгорнутим на мікроконтролерах й одноплатних комп'ютерах без застосування потужностей хмарних обчислень і знань з Інтернет. Модель FAI дозволяє всі потрібні експертні знання щодо прийняття рішень уявляти у вигляді структурованої множини однорідних елементів – ГЗ, та розглядати їх як елементи EDA. Така структура знань у вигляді множини ГЗ легко відображається на апаратне забезпечення FAI і компонується в окремі мікро EDA. Когнітивна модель FAI, яка підтримує поступове забування з часом подій через старіння даних, дозволяє подолати основну проблему застосування EDA – необхідність формування реакції в режимі реального часу компонентами FAI на основі сенсорних даних, отриманих асинхронно. У майбутньому планується розробка моделі навчання FAI в операційному режимі з метою автоматичного налаштування параметра швидкості старіння ν CF відповідно до динамічних властивостей середовища FAI.

Список використаної літератури

1. Unmanned Systems. NovAtel, 2022 [Електронний ресурс]. Режим доступу: <https://novatel.com/industries/unmanned-systems>
2. The Future of Autonomy. Isn't Human-Less. It's Human More. Lockheed Martin, 2022 [Електронний ресурс]. Режим доступу: <https://www.lockheedmartin.com/en-us/capabilities/autonomous-unmanned-systems.html>
3. Chen J., Sun J., Wang G. From Unmanned Systems to Autonomous Intelligent Systems // *Engineering*. 2022. Vol. 12. P. 16–19. DOI: 10.1016/j.eng.2021.10.007
4. Kargin A., Petrenko T. Knowledge Distillation for Autonomous Intelligent Unmanned Systems // In: Pedrycz W., Chen S.-M. (eds.) *Advancements in Knowledge Distillation: Towards New Horizons of Intelligent Systems*. Studies in Computational Intelligence, vol. 1100. Springer, 2023. P. 193–230.
5. Joseph L., Mondal A. K. (eds.) *Autonomous Driving and Advanced Driver-Assistance Systems (ADAS). Applications, Development, Legal Issues, and Testing*. 1st ed. Boca Raton : CRC Press, 2021. DOI: 10.1201/9781003048381
6. Gartner: 2025 Top Strategic Technology Trends [Електронний ресурс]. Режим доступу: <https://www.gartner.com/en/articles/top-technology-trends-2025>
7. Open X-Embodiment: Robotic Learning Datasets and RT-X Models [Електронний ресурс]. Режим доступу: <https://robotics-transformer-x.github.io/>
8. The TED AI Show: How AI robots learn like babies – but a million times faster w/ NVIDIA's Rev LeBaredian. 3 Dec 2024 [Електронний ресурс]. Режим доступу: <https://www.ted.com/podcasts/how-ai-robots-learn-just-like-babies-rev-lebaredian-transcript>
9. Levine S., Hausman K. The global project to make a general robotic brain // *IEEE Spectrum*. Jan 2024 [Електронний ресурс]. Режим доступу: <https://spectrum.ieee.org/global-robotic-brain>
10. McKinsey Global Survey. The state of AI in 2023: Generative AI's breakout year. 2023 [Електронний ресурс]. Режим доступу: <https://www.mckinsey.com/capabilities/quantumblack/our-insights/the-state-of-ai-in-2023-generative-AIs-breakout-year>
11. Yasumoto K., Yamaguchi H., Shigeno H. Survey of Real-time Processing Technologies of IoT Data Streams // *Journal of Information Processing*. 2016. Vol. 24, no. 2. P. 195–202. DOI: 10.2197/ipsjip.24.195
12. Winfield A. T. et al. IEEE P7001: A Proposed Standard on Transparency // *Frontiers in Robotics and AI*. 2021. DOI: 10.3389/frobt.2021.665729
13. Siemens. Next-Gen AI Manufacturing / Industrial Sector. 2021 [Електронний ресурс]. Режим доступу: <https://siemens.com/innovation>
14. Kargin A., Petrenko T. Feeling Artificial Intelligence for AI-Enabled Autonomous Systems // *Proc. Global Conf. on Artificial Intelligence and Internet of Things (GCAIoT)*. IEEE, Alamein New City, Egypt, 2022. P. 88–93.
15. Kargin A., Petrenko T. Feeling Artificial Intelligence. Cognitive Decision-making Model Borrowed from Living Beings // *International Symposium on Networks, Computers and Communications*. IEEE, Washington DC, USA, 2024. P. 1–6.
16. Reis J., Cohen Y., Melao N., Costa J., Jorge D. High-Tech Defense Industries: Developing Autonomous Intelligent Systems // *Applied Sciences*. 2021. Vol. 11(11). Art. 4920. DOI: 10.3390/app11114920
17. Ploennigs J., Cohn J., Stanford-Clark A. The future of IoT // *IEEE Internet of Things Magazine*. 2018. Vol. 1(1). P. 28–33.
18. Internet of Things: Understanding the Adventure. SAS Institute Inc., 17 Mar. 2020 [Електронний ресурс]. Режим доступу: <https://www.sas.com/sas/offers/20/iot-understanding-adventure.html>

19. Kargin A., Petrenko T. Method of Using Data from Intelligent Machine Short-Term Memory in Fuzzy Logic System // 2021 IEEE 7th World Forum on Internet of Things (WF-IoT). DOI: 10.1109/wf-iot51360.2021.9594918
20. Gupta H., Nayak A. Publish-subscribe system security requirement: A case study for V2V communication // IEEE Open Journal of the Computer Society. 2024. Vol. 5. P. 389–405. DOI: 10.1109/OJCS.2024.3442921
21. Rocha H. F. O. Practical Event-Driven Microservices Architecture: Building Sustainable and Highly Scalable Event-Driven Microservices. APRESS, 2025. 449 p. DOI: 10.1007/978-1-4842-7468-2
22. Islam G. Z., Motakabber S. M. A. A Comprehensive Review on the Internet of Things Network // Journal of Communications. 2025. Vol. 20, no. 1. P. 84–98. DOI: 10.12720/jcm.20.1.84-98
23. Kargin A., Petrenko T. Event-Driven Architecture of Feeling Artificial Intelligence Supported by Data Aging Model // In: Mejdoub Y., Elamri A., Kardouchi M. (eds.) Connected Objects, Artificial Intelligence, Telecommunications and Electronics Engineering (COCIA 2025). Lecture Notes in Networks and Systems, vol. 1584. Springer, Cham, 2025. P. 178–183. DOI: 10.1007/978-3-032-01536-5_27
24. Unified Modeling Language [Електронний ресурс]. Режим доступу: <https://www.uml.org/>
25. ZeroMQ. [Електронний ресурс]. Режим доступу: <https://zeromq.org/>

References

1. (2022) Unmanned Systems. NovAtel. URL: <https://novatel.com/industries/unmanned-systems>
2. (2022) The Future of Autonomy. Isn't Human-Less. It's Human More. Lockheed Martin. URL: <https://www.lockheedmartin.com/en-us/capabilities/autonomous-unmanned-systems.html>
3. Chen J., Sun J., Wang G. (2022) From Unmanned Systems to Autonomous Intelligent Systems // Engineering. Vol. 12. P. 16–19. DOI: 10.1016/j.eng.2021.10.007
4. Kargin A., Petrenko T. (2023) Knowledge Distillation for Autonomous Intelligent Unmanned Systems // In: Pedrycz W., Chen S.-M. (eds.) Advancements in Knowledge Distillation: Towards New Horizons of Intelligent Systems. Studies in Computational Intelligence, vol. 1100. Springer. P. 193–230.
5. Joseph L., Mondal A. K. (eds.) (2021) Autonomous Driving and Advanced Driver-Assistance Systems (ADAS). Applications, Development, Legal Issues, and Testing. 1st ed. Boca Raton : CRC Press. DOI: 10.1201/9781003048381
6. Gartner: 2025 Top Strategic Technology Trends URL: <https://www.gartner.com/en/articles/top-technology-trends-2025>
7. Open X-Embodiment: Robotic Learning Datasets and RT-X Models. URL: <https://robotics-transformer-x.github.io/>
8. The TED AI Show: How AI robots learn like babies – but a million times faster w/ NVIDIA's Rev LeBaredian. 3 Dec 2024. URL: <https://www.ted.com/podcasts/how-ai-robots-learn-just-like-babies-rev-lebaredian-transcript>
9. Levine S., Hausman K. The global project to make a general robotic brain // IEEE Spectrum. Jan 2024. URL: <https://spectrum.ieee.org/global-robotic-brain>
10. (2023) McKinsey Global Survey. The state of AI in 2023: Generative AI's breakout year. URL: <https://www.mckinsey.com/capabilities/quantumblack/our-insights/the-state-of-ai-in-2023-generative-AIs-breakout-year>
11. Yasumoto K., Yamaguchi H., Shigeno H. (2016) Survey of Real-time Processing Technologies of IoT Data Streams // Journal of Information Processing. Vol. 24, no. 2. P. 195–202. DOI: 10.2197/ipsjip.24.195
12. Winfield A. T. et al. (2021) IEEE P7001: A Proposed Standard on Transparency // Frontiers in Robotics and AI. DOI: 10.3389/frobt.2021.665729
13. Siemens. Next-Gen AI Manufacturing / Industrial Sector. 2021. URL: <https://siemens.com/innovation>
14. Kargin A., Petrenko T. (2022) Feeling Artificial Intelligence for AI-Enabled Autonomous Systems // Proc. Global Conf. on Artificial Intelligence and Internet of Things (GCAIoT). IEEE, Alamein New City, Egypt, P. 88–93.
15. Kargin A., Petrenko T. (2024) Feeling Artificial Intelligence. Cognitive Decision-making Model Borrowed from Living Beings // International Symposium on Networks, Computers and Communications. IEEE, Washington DC, USA, P. 1–6.
16. Reis J., Cohen Y., Melao N., Costa J., Jorge D. (2021) High-Tech Defense Industries: Developing Autonomous Intelligent Systems // Applied Sciences. Vol. 11(11). Art. 4920. DOI: 10.3390/app11114920
17. Ploennigs J., Cohn J., Stanford-Clark A. (2018) The future of IoT // IEEE Internet of Things Magazine. Vol. 1(1). P. 28–33.
18. Internet of Things: Understanding the Adventure. SAS Institute Inc., 17 Mar. 2020. <https://www.sas.com/sas/offers/20/iot-understanding-adventure.html>
19. Kargin A., Petrenko T. Method of Using Data from Intelligent Machine Short-Term Memory in Fuzzy Logic System // 2021 IEEE 7th World Forum on Internet of Things (WF-IoT). DOI: 10.1109/wf-iot51360.2021.9594918
20. Gupta H., Nayak A. (2024) Publish-subscribe system security requirement: A case study for V2V communication // IEEE Open Journal of the Computer Society. Vol. 5. P. 389–405. DOI: 10.1109/OJCS.2024.3442921
21. Rocha H. F. O. (2025) Practical Event-Driven Microservices Architecture: Building Sustainable and Highly Scalable Event-Driven Microservices. APRESS, 449 p. DOI: 10.1007/978-1-4842-7468-2

22. Islam G. Z., Motakabber S. M. A. (2025) A Comprehensive Review on the Internet of Things Network // Journal of Communications. Vol. 20, no. 1. P. 84–98. DOI: 10.12720/jcm.20.1.84-98

23. Kargin A., Petrenko T. (2025) Event-Driven Architecture of Feeling Artificial Intelligence Supported by Data Aging Model // In: Mejdoub Y., Elamri A., Kardouchi M. (eds.) Connected Objects, Artificial Intelligence, Telecommunications and Electronics Engineering (COCIA 2025). Lecture Notes in Networks and Systems, vol. 1584. Springer, Cham, P. 178–183. DOI: 10.1007/978-3-032-01536-5_27

24. Unified Modeling Language. URL: <https://www.uml.org/>

25. ZeroMQ. URL: <https://zeromq.org/>

Дата першого надходження рукопису до видання: 22.11.2025

Дата прийнятого до друку рукопису після рецензування: 17.12.2025

Дата публікації: 31.12.2025