

А. О. Каргін^{1,2}, Д. О. Гієвський², Д. Г. Олійник¹

¹ Харківський національний університет радіоелектроніки, Харків, Україна

² Український державний університет залізничного транспорту, Харків, Україна

БЕЗПЕРЕРВНЕ ПЛАНУВАННЯ І СИТУАЦІЙНЕ УПРАВЛІННЯ ЯК ЗАВДАННЯ ШТУЧНОГО ІНТЕЛЕКТУ ЩО ВІДЧУВАЄ

Анотація. Мотивація. Незважаючи на значний прогрес у галузі створення Безлюдних Систем (БС), забезпечення необхідного рівня їх автономності залишається актуальним завданням. Важливу роль у його вирішенні відіграє штучний інтелект. Особливості безлюдних систем породили нову модель «Штучний інтелект що відчуває» (ШІВ), що підтримує автономність. **Метою даної роботи** є розробка алгоритму що підтримує модель безперервного планування і ситуаційного управління імплементовану у систему Goal-Directed Control ШІВ. **Предметом дослідження** є методи та моделі управління автономними мобільними роботами на підставі даних від різноманітних сенсорів. **Результати.** Особливістю задачі управління автономними мобільними роботами є те, що для прийняття рішення в режимі реального часу щодо поточних дій використовують стан виконання плану, поточну ситуацію та можливість виконання частини плану дій, що залишилася, для досягнення мети. Наведено структуру багаторівневої розподіленої системи нечітких правил управління у поєднанні з системою продукційних правил планування. Розглянуто модифікований механізм нечіткого висновку, який завдяки введенню фактора впевненості здатний обробляти як факти щодо стану оточення так і стану виконання плану. Наведено алгоритм безперервного планування й приклади контрольних розрахунків. **Висновок.** Показано, що модифікація традиційного механізму логічного виведення в нечітких логічних системах, по-перше, шляхом введення контекстної пам'яті, що містить контекстні факти, по-друге, уявлення стану фактів, як і значень вхідних змінних у вигляді нечітких факторів впевненості, дозволяє застосувати їх для управління БС та використовувати усі переваги нечіткого управління щодо обробки невизначеності. Розробка традиційної нечіткої системи, спрямованої на управління реалізацією плану дій для автономного робота з урахуванням вищезазначених умов, що характеризується значною кількістю вхідних числових змінних від датчиків, є нерозв'язним завданням. Запропонована модель, яка складається з компонентів двох типів систем нечітких систем з лінгвістичними змінними та нечітких продукційних систем з факторами впевненості долає перелічені проблеми та зберігає переваги традиційних нечітких систем у повожденні з невизначеністю. **Напрямок подальших досліджень.** Майбутні кроки можуть бути присвячено розвитку моделі та інструментарію для інших класів безлюдних систем.

Ключові слова: автономні інтелектуальні безлюдні системи, штучний інтелект що відчуває, безперервне планування, ситуаційне управління.

Вступ

При створенні нових поколінь безпілотних автомобілів [1], автономних транспортних засобів військового призначення [2] та інших автономних систем основна увага приділяється проблемі підвищення рівня їхньої автономності [3]. Промислові БС складаються з автоматичних ліній і верстатів з програмною системою ЧПУ, до яких підключені транспортні та роботизовані системи [4]. У залізничній галузі БС створюють на базі розумних поїздів [5]. Розрізняють три класи БС залежно від рівня автономності: програмовані автоматичні БС, інтелектуальні БС та Автономні Інтелектуальні БС (АІБС) [4, 6, 7]. Перший тип має обмеження: БС може працювати лише за програмою і не може адаптуватися до будь-яких змін навколишнього середовища. Другий тип має певні можливості сприйняття, прийняття рішень і контролю, і може пристосовуватися до змін у навколишньому середовищі. АІБС має високий рівень автономності, може автономно приймати рішення в широкому діапазоні невизначеності [8]. ШІ є важливим компонентом, який забезпечує такий рівень автономії.

Апаратні ресурси АІБС встановлюють основу для автономії, а інтелект, наданий технологіями ШІ, визначає рівень самостійності прийняття рішень у цій структурі. Для створення БС обговорюються три типи моделей штучного інтелекту, які підтримують

автономію БС – механічний, мислячий та ШІ що Відчуває (ШІВ) [7, 9]. Зараз ШІВ є актуальним для АІБС і обговорюються різні аспекти від соціальних наслідків до архітектурних проектів [8, 10].

Метою цієї роботи є розробка алгоритму що підтримує модель безперервного планування і ситуаційного управління, який є імплементованим у системі Goal-Directed Control ШІВ [11-13].

Основна частина

Проблема автономності АІБС на прикладі колісного роботу. Нехай місією АІБС буде повторюване перевезення певного вантажу роботом з позиції А на позицію В (рис.1). Для досягнення мети робот має подолати певні етапи, тобто досягти виконання локальних цілей. В даному випадку, локальні цілі АІБС можуть бути представлені певними етапами плану реалізації місії, тобто послідовністю станів системи «Оточення-АІБС», які система має виконати завдяки своїм діям. Наведемо приклад такої послідовності станів, що відображає план реалізації місії: «Завантаження АІБС на позиції В», «Переміщення АІБС на позицію 5», «Переміщення АІБС на позицію 3», «Переміщення АІБС на позицію А», «Розвантаження вантажу АІБС на позиції А», «Переміщення АІБС на позицію 1», «Переміщення АІБС на позицію 4», «Переміщення АІБС на позицію В» і далі повторити описаний процес завантаження та переміщення вантажу (рис. 1):

$$\begin{aligned} &\rightarrow A^{Load} \rightarrow 1^{Move} \rightarrow 4^{Move} \rightarrow 5^{Move} \rightarrow B^{Unload} \rightarrow 5^{Move} \\ &\rightarrow 4^{Move} \rightarrow 1^{Move} \rightarrow A^{Load} \rightarrow \end{aligned} \quad (1)$$

Варто зауважити, що процес досягнення кожної локальної цілі може бути підданий подальшій декомпозиції і бути представленим у формі більш деталізованого плану дій.

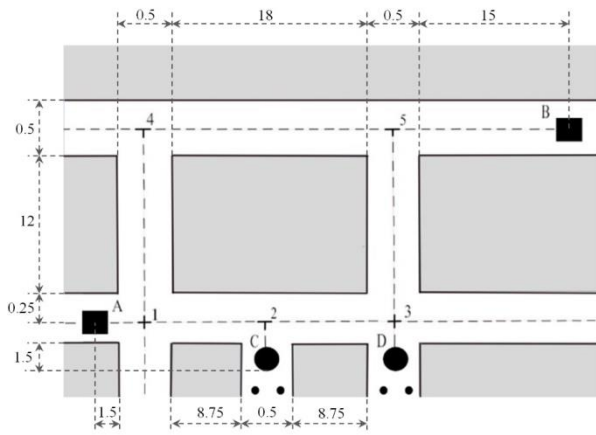


Рис. 1. Модель оточення АІБС

Також важливо враховувати, що сама місія являє собою не лише виконання переміщення між певними позиціями, але й включає в себе подолання перешкод. Через це план реалізації місії не є сталим, мусить включати в себе додаткові етапи, що допомагають долати перешкоди, перебудовувати за необхідності план, та сприяти досягненню кінцевої мети місії. Прикладами перешкод є пошкодження оточення, на яке спирається АІБС при виконанні місії (мітки, розмітка тощо), виникнення об'єкту на путі АІБС, що перешкоджає подальшому руху, або нестача заряду батареї, яка живить роботу.

Управління реалізацією плану дій у визначених умовах для досягнення кінцевої мети є завданням ШПВ, що розглядається в роботах [25, 26]. Однак відзначається головний недолік нечітких логічних систем для вирішення завдання ситуаційного управління поєднаного з безперервним плануванням. Це відсутність втримання контексту, через що немає розуміння в якому конкретно стані знаходиться система в визначений момент часу і які дії мають бути виконані на поточному кроці задля виконання місії. Довільний вибір траєкторії не сприяє виконанню плану дій, оскільки визначена планом послідовність станів системи потребує руху за певною заданою траєкторією. Визначене обмеження нечітких систем пропонується подолати шляхом імплементації контекстної залежності в множину незалежних нечітких правил БЗ [25], яка дозволяє додати в БЗ послідовність локальних цілей етапів плану, та при цьому зберегти незалежність правил одне від одного.

В статті розглядається реалізація запропонованої моделі та апробація нового алгоритму безперервного планування, який є імplementованим у системі Goal-Directed ШПВ.

Контекстно залежна модель управління що направляється ціллю. Реалізація контекстної залежності потребує певних змін у структурі нечіткої системи у тому числі фактів й правил. Для демонст-

рації прикладу використаємо визначений підхід [25]. Його реалізація потребує розділення множини фактів F на дві підмножини, де перша містить факти F^{plan} , на яких визначають етапи плану (локальні цілі), а друга – факти F^{sit} стосовно ситуації, в якій перебуває оточення АІБС.

$$F = \{F^{plan} = \{f_i, i=1,2,\dots,n\} = \{A^{Load}, B^{Unload}, 1^{Move}, 2^{Move}, 3^{Move}, 4^{Move}, 5^{Move}\}, F^{sit} = \{g_j, j=1,2,\dots,m\}\}. \quad (2)$$

Далі для збереження стадій виконання плану і їх відображення в поточний контекст введемо для кожного факту з F^{plan} (2) додатковий контекстний факт й помістимо їх в Контекстну Пам'ять (КП). Відповідно до кожного сенсорного факту f_i з (2) позначимо цей контекстний факт як $*f_i$. Таким чином, фактам за F^{plan} відповідатимуть наведені в КП контекстні факти з множини F .

$$F = \{ *A^{Load}, *B^{Load}, *1^{Move}, *2^{Move}, *3^{Move}, *4^{Move}, *5^{Move} \} \quad (3)$$

Стан Нечіткої Логічної Системи (НЛС) на довольному етапі реалізації плану визначає вектор CF значень Факторів Впевненості (ФВ) у базі фактів:

$$CF = (cf(f_i), cf(*f_i), i=1,2,\dots,n, cf(g_j), j=1,2,\dots,m). \quad (4)$$

Значення ФВ контекстних фактів оновлюється в результаті виконання правил НЛС шару управління реалізацією плану, що відслідковують контекст. Значення ФВ фактів, що є складовими підмножин F^{plan} та F^{sit} , визначаються на підставі даних, що надходять від сенсорів АІБС [27, 28].

Знання АІБС цієї досвід у вигляді фрагментів траєкторії поведінки, що призвела до досягнення мети при виконанні місії або усунення перешкод. Ці фрагменти являють собою, серед іншого, різні варіанти реалізації планів дій. Кожен фрагмент представляє структуру знань у вигляді рис. 2.

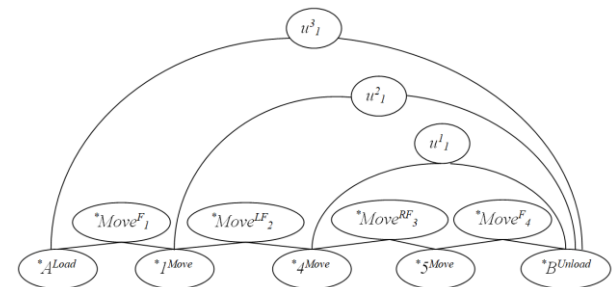


Рис. 2. Структура частини знань бази знань, що відповідає правилам R1-R7

На нижньому рівні структури це є елементарні знання що даються у формі «факт₁-дія-факт₂», які відображають, що за наявності факту₁ акт дії призводить до появи факту₂. Нижче в (5) та (6) наведено набір правил, які представляють фрагменти реалізації місії в оточенні (рис. 1), що потребується для завантаження грузу в точці А та розвантаженні в точці В. Структура відображає не тільки елементарні причинно-наслідкові зв'язки, а й стійкі більш тривалі ланцюги зв'язків. Наприклад, мета $*B^{Unload}$ досягається (можна отримати факт $*B^{Unload}$), якщо за наявності факту $*A^{Load}$ реалізується дія u^3_1 , яка є узагальненням послідовності дій $*Move^F_1$ і u^2_1 , де верхній індекс ви-

значає характер руху (F – рухатися прямо, LF – повернути наліво та рухатись прямо, RF – повернути направо та рухатись прямо), а нижній індекс є ідентифікаційним номером дії. Стосовно позначок дій високого рівня $u^n - l$ позначає рівень ієрархії правила, n – номер сценарію руху, що реалізується. Так само мета $*B^{Unload}$ досягається, якщо за наявності факту $*I^{Move}$ реалізується дія $*u^2_1$, яка є узагальненням послідовності дій $*4^{Move}$ ТА u^1_1 . В свою чергу, u^1_1 є уособленням послідовності $*Move^{RF}_3$ та $*Move^F_4$. Ці знання представлені в Базі Знань (БЗ) правилами R_5 , R_6 і R_7 , відповідно.

R1: ЯКЩО	$*A^{Load}$ ТА	$*Move^F_1$	ТО	$*I^{Move}$	($cf=0.9$)
R2: ЯКЩО	$*I^{Move}$ ТА	$*Move^{LF}_2$	ТО	$*4^{Move}$	($cf=0.9$)
R3: ЯКЩО	$*4^{Move}$ ТА	$*Move^{RF}_3$	ТО	$*5^{Move}$	($cf=0.9$)
R4: ЯКЩО	$*5^{Move}$ ТА	$*Move^F_4$	ТО	$*B^{Unload}$	($cf=0.9$)
R5: ЯКЩО	$*A^{Load}$ ТА	u^3_1	ТО	$*B^{Unload}$	($cf=0.7$)
R6: ЯКЩО	$*I^{Move}$ ТА	u^2_1	ТО	$*B^{Unload}$	($cf=0.75$)
R7: ЯКЩО	$*4^{Move}$ ТА	u^1_1	ТО	$*B^{Unload}$	($cf=0.8$) (5)

Оскільки рух реалізується за умови наявності перешкод, в якості прикладу, наведемо одну з них, що є нестачею заряду батареї робота. АІБС має вирушити до зарядної станції С або D, і лише потім повернутися до виконання плану.

R8: ЯКЩО	$*A^{Load}$ ТА	$*Move^F_1$	ТО	$*I^{Move}$	($cf=0.9$)
R9: ЯКЩО	$*I^{Move}$ ТА	$*Move^F_5$	ТО	$*2^{Move}$	($cf=0.9$)
R10: ЯКЩО	$*2^{Move}$ ТА	$*Move^{RF}_6$	ТО	$*C^{Charge}$	($cf=0.9$)
R11: ЯКЩО	$*A^{Load}$ ТА	u^2_2	ТО	$*C^{Charge}$	($cf=0.8$)
R12: ЯКЩО	$*I^{Move}$ ТА	u^2_2	ТО	$*C^{Charge}$	($cf=0.85$)
R13: ЯКЩО	$*A^{Load}$ ТА	$*Move^F_1$	ТО	$*I^{Move}$	($cf=0.9$)
R14: ЯКЩО	$*I^{Move}$ ТА	$*Move^F_5$	ТО	$*2^{Move}$	($cf=0.9$)
R15: ЯКЩО	$*2^{Move}$ ТА	$*Move^F_7$	ТО	$*3^{Move}$	($cf=0.9$)
R16: ЯКЩО	$*3^{Move}$ ТА	$*Move^{RF}_8$	ТО	$*D^{Charge}$	($cf=0.9$)
R17: ЯКЩО	$*A^{Load}$ ТА	u^3_3	ТО	$*D^{Charge}$	($cf=0.7$)
R18: ЯКЩО	$*I^{Move}$ ТА	u^2_3	ТО	$*D^{Charge}$	($cf=0.75$)
R19: ЯКЩО	$*2^{Move}$ ТА	u^1_3	ТО	$*D^{Charge}$	($cf=0.85$)
R20: ЯКЩО	$*C^{Charge}$ ТА	$*Move^F_2$	ТО	$*2^{Move}$	($cf=0.9$)
R21: ЯКЩО	$*2^{Move}$ ТА	$*Move^F_7$	ТО	$*3^{Move}$	($cf=0.9$)
R22: ЯКЩО	$*3^{Move}$ ТА	$*Move^{LF}_9$	ТО	$*5^{Move}$	($cf=0.9$)
R23: ЯКЩО	$*5^{Move}$ ТА	$*Move^{RF}_{10}$	ТО	$*B^{Unload}$	($cf=0.9$)
R24: ЯКЩО	$*C^{Charge}$ ТА	u^3_4	ТО	$*B^{Unload}$	($cf=0.75$)
R25: ЯКЩО	$*2^{Move}$ ТА	u^2_4	ТО	$*B^{Unload}$	($cf=0.8$)
R26: ЯКЩО	$*3^{Move}$ ТА	u^1_4	ТО	$*B^{Unload}$	($cf=0.85$)
R27: ЯКЩО	$*D^{Charge}$ ТА	$*Move^F_{11}$	ТО	$*3^{Move}$	($cf=0.9$)
R28: ЯКЩО	$*3^{Move}$ ТА	$*Move^F_{12}$	ТО	$*5^{Move}$	($cf=0.9$)
R29: ЯКЩО	$*5^{Move}$ ТА	$*Move^{RF}_9$	ТО	$*B^{Unload}$	($cf=0.9$)
R30: ЯКЩО	$*D^{Charge}$ ТА	u^2_5	ТО	$*B^{Unload}$	($cf=0.75$)
R31: ЯКЩО	$*3^{Move}$ ТА	u^1_5	ТО	$*B^{Unload}$	($cf=0.8$) (6)

У випадку жорсткої нестачі енергії обирається маршрут 2, за яким АІБС рухається до точки С для виконання зарядки з виконанням узагальнюючих дій u^2_2 та u^2_1 (правила R_8 - R_{12}). Після активації зарядки далі може бути реалізований маршрут 4, за яким робот може досягти кінцевої дії $*B^{Unload}$ (правила R_{20} - R_{26}). Подібний сценарій також реалізовано для випадку помірної нестачі енергії, коли АІБС спочатку рухається до точки D для виконання зарядки, і вже потім до мети $*B^{Unload}$.

Goal-Directed механізм ШІВ реалізує нечітке логічне виведення з комбінацією методів прямого та зворотного ланцюгового висновку з ФВ. Прямий ланцюговий висновок використовує ФВ з бази фактів F^{plan} та F^{sit} а зворотній ланцюговий висновок ви-

користує факти з КП. Перед активацією механізму виведення ФВ фактів з F^{plan} та F^{sit} описують реальний стан середовища та етапів плану. Факти з КП відображають стани, що характеризують можливість реалізації дій, що ведуть до досягнення локальних цілей. Наприклад, $cf^{Re} = 0,9$ факту $*Move^F_4$ відображає можливість реалізації переходу від $*5^{Move}$ до $*B^{Unload}$. Інше значення $cf^{Re} = -0,9$ свідчить про неможливість досягнення локальної мети $*A^{UnLd}$ таким шляхом (правило R_4 в (5)). Ці значення змінюються відповідно до змін стану навколишнього середовища в реальному часі.

Алгоритм безперервного планування. Алгоритм представляє собою інтерпретацію методу зворотного виводу, який дозволяє отримати усі можливі шляхи розрахунку заданого цільового факту з розрахованими ФВ кожного знайденого шляху та отримати комбінований фактор впевненості (КФВ).

Алгоритм базується на використанні ідеї алгоритму пошуку в ширину та створенні дерева шляхів розрахунку цільового факту. Кожен вузол дерева представляє собою правило, яке необхідно розрахувати, щоб отримати цільовий факт або один з фактів, що необхідні для його розрахунку.

Структура дерева має стандартний вигляд: дерево має корінь, гілки, листя. Однак, структура вузла дерева була модифікована для можливості реалізації алгоритму.

Формальне визначення вузла дерева має вигляд:

$$\langle N, R_N, P_N, \{c_i: \forall c_i \in C_N\}, D_N, CF_N, \exists CF_B \leftrightarrow C_N \in \emptyset \rangle \quad (7)$$

де N є ідентифікатором вузла; R_N є ідентифікатором правила, за яким розраховується вузол N ; P_N є батьківським вузлом N ; C_N є множиною дочірніх вузлів N ; D_N є множиною залежностей, які необхідно мати для розрахунку вузла N за правилом R_N ; CF_N є поточним розрахованим ФВ вузла N ; CF_B є поточним розрахованим ФВ шляху (гілки дерева), до якої входить вузол N за умови, що вузол N є листям дерева.

Формальне визначення множини залежностей D_N вузла у (7) має вигляд:

$$\{ \langle f_j, \{T_{jk}\} \rangle : \forall f_j \in F_{RN} \} \quad (8)$$

де F_{RN} є множина необхідних фактів, що наявні у правилі R_N вузла N ; $\{T_{jk}\}$ є множиною кортежів зі структурою (9).

Формальне визначення кортежу T_f має вигляд:

$$\langle r, CF \rangle : r \in KB_f \quad (9)$$

де KB_f є множина правил для визначення факту f ; CF є поточним ФВ правила r .

Глобальні залежності (GD) є структурою, яка надає можливість відслідковувати вже знайдені правила, які необхідні для розрахунку кінцевої цілі, що прибирає необхідність повторного пошуку вже знайденого правила.

Формальне визначення структури GD має вигляд:

$$\{ \langle f_j, \{r_k: \forall r_k \in KB_f\} \rangle, \forall f_j \in F \} \quad (10)$$

де F є множиною всіх фактів, що використовуються у системі; KB_f є множина правил для визначення факту f .

Вхідними даними алгоритму є: БЗ, що містить у собі правила формату «ЯКЦО-ТО»; База Даних (БД), що містить відомі факти на момент початку використання алгоритму; цільовий факт (goal fact) – факт, для якого необхідно знайти шляхи його розрахунку.

Ідея алгоритму наступна: з черги отримується факт для пошуку відповідних правил для розрахунку. У випадку, коли отриманий факт – цільовий, створюється нове дерево шляхів розрахунку цільового факту, використовуючи знайдені правила як корені дерев. Інакше відбувається пошук вузлів дерев, які потребують отриманий факт для розрахунку, та їх доповнення знайденими правилами.

Після доповнення вузлів відбувається уточнення ФВ відповідних шляхів.

Шляхи розрахунку цільового факту отримуються із використанням листя дерев, за умови, що всі їх залежності були знайдені. Якщо хоча б до одної залежності не було знайдено відповідного правила, то шлях вважається недійсним й не бере участь у розрахунку КФВ.

Таким чином, результатом роботи даного алгоритму є:

1) множина шляхів для розрахунку цільового факту з розрахованим ФВ. Кожен шлях є впорядкованим списком правил, які необхідно послідовно розрахувати для отримання цільового факту, проте на момент завершення роботи алгоритму кожен шлях матиме розрахований ФВ цільового факту;

2) КФВ, який розраховано на основі ФВ кожного дійсного шляху.

В залежності від структури правил у БЗ розрахунок ФВ відбувається за (11) чи (12). В обох випадках КФВ розраховується за (13).

$$cf(H, E_1 \wedge E_2 \wedge \dots \wedge E_n) = \min[cf(E_1), cf(E_2), \dots, cf(E_n)] * cf \quad (11)$$

$$cf(H, E_1 \vee E_2 \vee \dots \vee E_n) = \max[cf(E_1), cf(E_2), \dots, cf(E_n)] * cf \quad (12)$$

де H є гіпотезою, або ж фактом, що розраховується; E_n є множина необхідних для розрахунку фактів; cf є базовим ФВ гіпотези H ; $cf(E_i)$ є ФВ факту E_i .

$$cf(cf_1, cf_2) = \begin{cases} cf_1 + cf_2 * (1 - cf_1) & cf_1 > 0 \text{ and } cf_2 > 0 \\ \frac{cf_1 + cf_2}{1 - \min[|cf_1|, |cf_2|]} & cf_1 < 0 \text{ or } cf_2 < 0 \\ cf_1 + cf_2 * (1 + cf_1) & cf_1 < 0 \text{ and } cf_2 < 0 \end{cases} \quad (13)$$

Алгоритм модифікованого зворотного виводу також використовує низку допоміжних алгоритмів-функцій.

До них входять наступні:

- GetLeaves,
- RestorePath,
- UpdateAncestors.

Рекурсивний алгоритм GetLeaves, використовуючи початковий вузол, повертає множину листків піддерева, де коренем є початковий вузол. Він використовується у випадках, коли необхідно до поточно-го вузла додати один або декілька нових одного рівня

(було знайдено декілька правил для розрахунку факту), проте поточний вузол може мати дочірні вузли. Тому, щоб зберегти коректні шляхи розрахунку цільового факту, необхідно додавати їх до листків піддерева.

Ідея алгоритму-функції наступна: рекурсивно переходити до дочірніх вузлів, доки S_N поточного вузла не дорівнюватиме порожній множині – це кінцева точка рекурсії.

Результатом є множина листків піддерева.

У момент знаходження правила для розрахунку поточного факту з черги, може трапитися ситуація, коли серед залежностей є ті правила, що вже було знайдено, тому, щоб уникнути повторного знаходження вже знайдених правил, було розроблено алгоритм-функцію RestorePath.

Цей рекурсивний алгоритм на вході отримує правило, з якого буде відбуватися відновлення шляхів. Відновлення відбуватиметься до тих пір, доки не відновиться правило, у якому всі залежності – дані з БД, або відповідні правила для необхідних фактів ще не були знайдені.

Ідея полягає у наступному: відбувається співставлення залежностей правила з GD. У разі, коли значення факту у GD є множина $\{0\}$, відбувається отримання його значення ФВ з БД. Інакше, коли значення множини відрізняється та не дорівнює порожній множині, відбувається відновлення кожного вузла з множини.

Результатом алгоритму-функції є оновлений вузол, з якого почалося відновлення шляху.

Після додавання нового вузла або відновлення шляху відбувається уточнення ФВ починаючи з листів піддерева.

Ідея полягає у наступному: через те, що кожен факт може мати декілька правил для розрахунку, використовується множина кортежів, що має структуру (9) для ідентифікації відповідного правила у залежностях вузла для оновлення. Починаючи з вузла-листя, відбувається поступовий перехід до батьківських вузлів оновлюючи їх залежності та доповнюючи множину кортежів.

Результатом є ФВ цільового факту використовуючи відповідний шлях розрахунку.

Висновки

Для дослідження можливостей нового напрямку штучного інтелекту, а саме штучний інтелект що відчуває, призначеного для підтримки автономності безлюдних систем, створюється модель ситуаційного управління поєднана з безперервним планування дій.

Показано, що модифікація традиційного механізму логічного виведення в нечітких логічних системах, по-перше, шляхом введення контекстної пам'яті, що містить контекстні факти відбиваючи стан виконання плану дії, по-друге, уявлення стану фактів, як і значень вхідних змінних у вигляді нечітких факторів впевненості, дозволяє застосувати їх для управління АІБС та використовувати усі переваги нечіткого управління щодо обробки невизначеності.

Розробка традиційної нечіткої системи, спрямованої на управління реалізацією плану дій для автономного робота з урахуванням вищезазначених умов, що характеризуються значною кількістю вхідних числових змінних від датчиків, є нерозв'язним завданням.

Запропонована модель, яка складається з компонентів двох типів нечітких систем з лінгвістичними змінними та фактами продукційних систем завдяки введенню факторів впевненості долає перелічені проблеми та зберігає переваги традиційних нечітких систем у поводженні з невизначеністю.

СПИСОК ЛІТЕРАТУРИ

1. L. Joseph, A.K. Mondal, Eds, *Autonomous Driving and Advanced Driver-Assistance Systems (ADAS). Applications, Development, Legal Issues, and Testing*. 1st edn. CRC Press, Boca Raton, 2021, doi:10.1201/9781003048381.
2. "Sikorsky and DARPA's Autonomous Black Hawk Flies Logistics And Rescue Missions Without Pilots On Board," Lockheed Martin Corporation, 2022, USA, Accessed: Aug. 10, 2023. [Online]. Available: <https://news.lockheedmartin.com/2022-11-02-Sikorsky-and-DARPA-Autonomous-Black-Hawk-R-Flies-Logistics-and-Rescue-Missions-Without-Pilots-on-Board>
3. J. Deichmann et al., "Autonomous driving's future: Convenient and connected," McKinsey Center for Future Mobility. Report, Jan. 2023, Accessed: August 15, 2023. [Online]. Available: <https://www.mckinsey.com/>
4. H. Chen et al., "From Automation System to Autonomous System: An Architecture Perspective," *J. of Marine Sci. and Eng.*, vol. 9, no. 6, Jun. 2021, doi: 10.3390/jmse9060645.
5. Rail Technical Strategy. *Innovating across Britain's railway*. Oct. 2022. Accessed: Aug. 10, 2023. [Online]. Available: <https://railtechnicalstrategy.co.uk/wp-content/uploads/2022/10/The-Rail-Technical-Strategy.pdf>
6. T. Zhang et al., "Current trends in the development of intelligent unmanned autonomous systems," *Frontiers Inf. Technol. Electron. Eng.*, vol. 18, Feb. 2017, pp. 68–85, doi: 10.1631/FITEE.1601650.
7. J. Reis, Y. Cohen, N. Melao, J. Costa, and D. Jorge, "High-Tech Defense Industries: Developing Autonomous Intelligent Systems," *Appl. Sci.*, vol. 11, 4920, 2021, doi: 10.3390/app11114920.
8. J. Chena, J. Sun, and G.Wang, "From Unmanned Systems to Autonomous Intelligent Systems," *Engineering*, vol.12, May 2022, pp. 16-19, doi: 10.1016/j.eng.2021.10.007.
9. M. Czerwinski, J. Hernandez, D. McDuff, "Building an AI that feels," *Appl. Sci.*, vol.11, 4920, Apr. 2021, doi: 10.3390/app11114920.
10. M. Huang and R. Rust, "Artificial Intelligence in Service," *J. of Service Res.*, vol. 21(2), Feb. 2018, pp.155-172, doi: 10.1177/1094670517752459.
11. A. Kargin, T. Petrenko, "Feeling Artificial Intelligence for AI-Enabled Autonomous Systems" in *Conf. Proc. of 2022 IEEE Global Conference on Artificial Intelligence and Internet of Things (GCAIoT) Alamein New City, Egypt, 18-21 December 2022*, P.88-93.
12. A. Kargin and T. Petrenko, "Spatio-Temporal Data Interpretation Based on Perceptual Model," in *Advances in Spatio-Temporal Segmentation of Visual Data. Studies in Computational Intelligence*, V. Mashtalir, I. Ruban, V. Levashenko, Eds., vol. 876, Springer, Cham, 2020, pp. 101-159.
13. A. Kargin and T. Petrenko, "Multi-level Computing With Words Model to Autonomous Systems Control," in *Proc. 9th Int. Conf. Inf. Control Sys.&Tech (ICST-2020)*, A. Pakštis, T. Hovorushchenko, V. Vychuzhanin, H. Yin, N. Rudnichenko. Eds. Odessa, Ukraine, Sep. 24–26, 2020, *CEUR Workshop Proceedings*, vol. 2711, pp. 16-30. [Online]. Available: <http://ceur-ws.org/Vol-2711/>

Received (Надійшла) 12.02.2024

Accepted for publication (Прийнята до друку) 17.04.2024

Continuous planning and situational control as an Feeling Artificial Intelligence task

Anatolii Kargin, Dmytro Hiiievskiy, Dmytro Oliinyk

Abstract. Motivation Despite significant progress in the field of creating unmanned systems, ensuring the necessary level of their autonomy remains an urgent task. Artificial intelligence plays an important role in its solution. Features of unmanned systems have given rise to a new model of Feeling Artificial Intelligence (FAI) that supports autonomy. **The goal of this work** is to develop an algorithm that supports the model of continuous planning and situational control implemented in the Goal-Directed Control system of FAI. **The object of research** is the methods and models of controlling autonomous mobile robots based on data from various sensors. **Results.** The peculiarity of the task of autonomous mobile robots control is that they use the state of execution of the plan, the current situation and the possibility of executing the remaining part of the action plan to achieve the goal in order to make a decision in real time regarding the current actions. The structure of a multi-level distributed system of fuzzy rules in combination with a system of production planning rules is given. A modified mechanism of fuzzy inference is considered, which, thanks to the introduction of a certainty factor, is able to process both facts about the state of the environment and the state of plan execution. The continuous planning algorithm and examples of control calculations are given. **Conclusions.** It is shown that the modification of the traditional mechanism of logical reasoning in fuzzy logic systems, firstly, by introducing a context memory containing contextual facts, and secondly, the representation of the state of facts, as well as the values of the input variables in the form of fuzzy certainty factors, allows applying them for autonomous intelligence unmanned system control and take full advantage of fuzzy control in handling uncertainty. The development of a traditional fuzzy system aimed at managing the implementation of an action plan for an autonomous robot, taking into account the above conditions, characterized by a significant number of input numerical variables from sensors, is an intractable task. The proposed model, which consists of components of two types of systems, fuzzy systems with linguistic variables and fuzzy production systems with certainty factors, overcomes the listed problems and preserves the advantages of traditional fuzzy systems in dealing with uncertainty.

Keywords: Autonomous intelligent unmanned systems, Feeling Artificial Intelligence, continuous planning, situational control.