

ФОРМАЛЬНА МОДЕЛЬ ОПРАЦЮВАННЯ ЗНАНЬ У СИСТЕМАХ ІЗ СИТУАЦІЙНОЮ ОБІЗНАНІСТЮ

© Буров Є. В., Микіч Х. І., 2017

Розглянуто моделі, методи формалізації та моделювання процесу прийняття рішень у системах із ситуаційною обізнаністю. Наведено математичну формалізацію понять ситуації. Подано алгебраїчну модель, яка дає змогу використовувати різні методи моделювання ситуаційної обізнаності, що ґрунтується на фреймворку “Алгебра систем”. Проаналізовано поняття дескриптивної логіки та її переваги і недоліки. Розроблено відображення між алгебраїчною моделлю, дескриптивною логікою та моделлю інтерпретованих систем.

Ключові слова: ситуаційна обізнаність, формальна модель, фреймворк, алгебра систем, дескриптивна логіка, інтерпретовані системи.

In the article are described models and methods of formalizing and modeling process of decision making in systems with situation awareness (SAW systems). The definitions of mathematical formalization of situation are discussed. Also, the unifying algebraic model, allowing the usage of different tools for situation awareness modeling, based of Algebra of Systems was proposed. We analyze the term of description logic and its advantages and disadvantages. The mappings between algebraic model, description logic and interpreted systems are proposed.

Key words: situation awareness, formal model, framework, Algebra of systems, description logic, interpreted systems.

Вступ та постановка проблеми

Сьогодні значного розвитку набули класичні методології підтримки прийняття рішень, що реалізують вибір рішення з множини альтернатив для добре визначених та структурованих задач. Водночас сфера неструктурованих та недостатньо визначених задач потребує подальших досліджень. Складовою частиною проблематики підтримки прийняття рішення для складних задач є виявлення та формалізація знань експерта про процес прийняття рішень. Отже, актуальним є завдання розроблення і впровадження інтелектуальних систем підтримки прийняття рішень, що використовують попередній досвід з ідентифікації проблемних ситуацій.

Необхідною умовою для побудови автономних інтелектуальних систем є формування та підтримка в них ситуаційної обізнаності (situation awareness) – здатності отримувати інформацію про стан зовнішнього середовища та формування на її основі та з використанням знань про предметну область висновків про необхідні дії або майбутні стани середовища.

Проблематика систем із ситуаційною обізнаністю (СО) доволі довго розвивалася для людино-машинних застосувань в аерокосмічній та військовій галузях [1]. Сьогодні напрям систем із ситуаційною обізнаністю отримав новий поштовх для розвитку в зв'язку з розробленням нових методів подання та опрацювання знань для вирішення завдання побудови повністю автономних інтелектуальних систем.

Важливим завданням є розроблення формальних моделей та фреймворків для подання та опрацювання знань у системах із СО. Сьогодні для подання предметної області та розв'язання задач у системах із СО використовуються різні методи. Зокрема, дедалі більшого поширення набувають онтології для подання предметної області [2] та дескриптивна логіка [2] для формування логічних висновків. Для розгляду динамічних станів використовують апарат інтерпретованих систем [3].

Аналіз останніх досліджень і публікацій

У роботі [3] наведено класифікацію формальних методів, що використовуються у системах із СО, переваги та недоліки кожного з підходів. Водночас відсутній спільний, об'єднувальний математичний фреймворк, який дає змогу використовувати різні методи для розв'язання різних задач у межах єдиної системи із СО. В цій роботі запропоновано такий фреймворк на основі алгебраїчної моделі подання знань про предметну область.

Для дослідження СО користуються моделями, в яких деталізуються складові частини СО та їх взаємозалежності. Вибір конкретного типу моделі залежить від постановки задачі дослідження, особливостей предметної області, вимог до системи з СО.

У літературі [3] запропоновано три типи моделей СО – процесні, функціональні та формальні моделі. Кожен з цих типів має переваги та недоліки і сферу застосування.

Функціональні та процесні моделі СО є концептуальними моделями. На відміну від них, формальні моделі дають змогу специфікувати предметну область та процес отримання СО, задіяти математичні методи для подання та повторного використання знань, реалізувати механізми логічного виведення, здійснити валідацію моделей. Аналізуючи математичні моделі знань та даних про предметну область, дослідники отримують нові знання про предметну область та про процес досягнення СО. Побудова формальної моделі є необхідною умовою для кращого розуміння вимог до системи із СО та основою для подальшого розроблення фреймворків та архітектур систем із СО.

Запропоновану класифікацію формальних моделей [3] наведено на рис. 1.

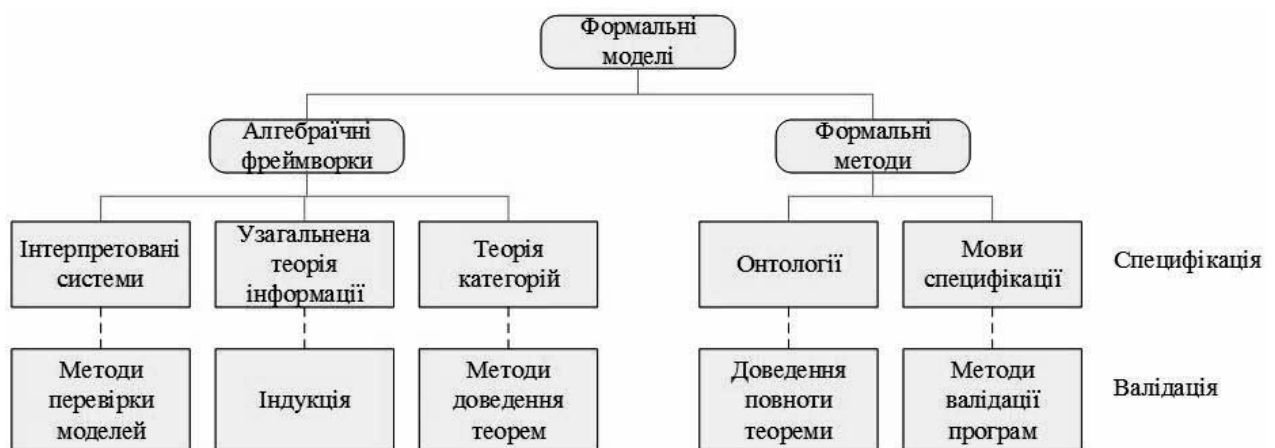


Рис. 1. Класифікація формальних моделей СО

Тобто сьогодні використовують:

- Теорію категорій, яка досліджує процеси об'єднання інформації, використовує традиційний підхід доведення теорем.
- Узагальнену теорію інформації – для моделювання процесу опрацювання інформації та невизначеностей на всіх рівнях JDL моделі (модель синтезу даних). Основана насамперед на використанні індукції;
- Інтерпретовані системи – новий підхід, що ґрунтується на семантиці інтерпретованих систем. Використовувався спочатку для аналізу розподілених систем, оснований на ефективній методиці перевірки моделі.

Відомі методи формалізації ситуації дають змогу відображати лише певні підзадачі СО. Проте процес прийняття рішення та ідентифікації ситуації у реальному світі потребує поєднання декількох методів, тобто необхідно розробити такий цілісний підхід, що дає змогу в межах однієї системи поєднувати різні методи моделювання СО. Цього можна досягти розробленням основної формальної моделі та побудовою переходів від неї до інших методів та моделей.

У математичній моделі ситуацію можна змоделювати як ситуацію:

- статичних станів;
- наближених станів – груп станів як одного стану;
- сусідніх станів та граничних станів;
- динаміки переходів між станами;
- траєкторій руху між станами, асоціативних та наслідкових зв'язків;
- використання онтологій для подання предметної області та можливості логічного виведення (дескриптивна логіка, інші форми логіки, використання моделей);
- можливість врахування нечіткостей та розуміння про нечіткості.

Для вирішення завдання ідентифікації проблемних ситуацій потрібно створити метод, що дає змогу формалізувати знання експерта про ознаки проблемних ситуацій, накопичувати та повторно використовувати досвід щодо прийняття рішень в аналогічних ситуаціях. Процес використання знань у СО системах відображено на рис. 2.

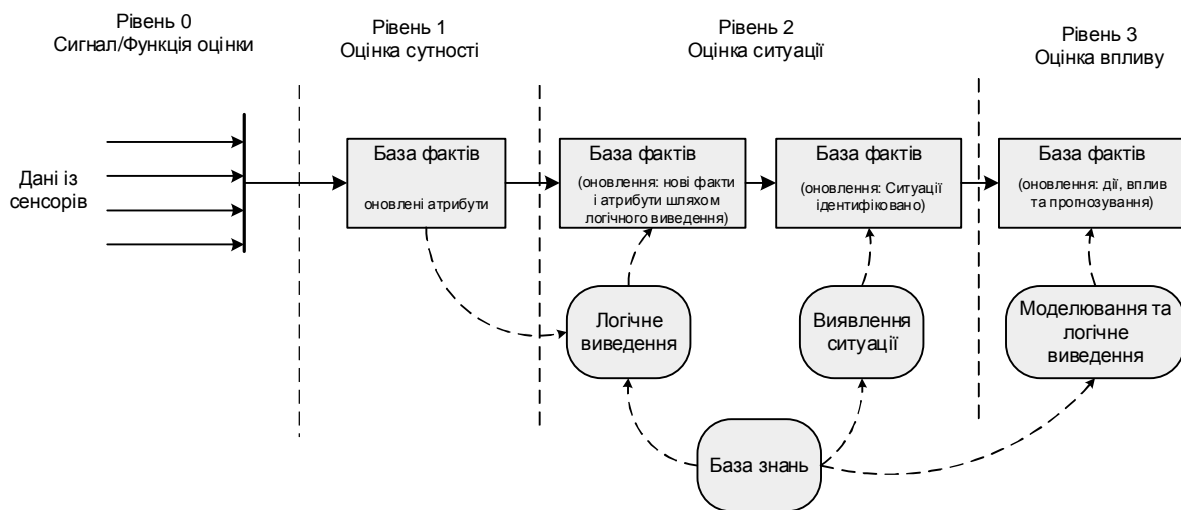


Рис. 2. Процес опрацювання знань у системах із СО

Сенсори накопичують дані, що надходять із зовнішнього середовища. Ці дані належать до об'єктів у базі фактів (перший рівень JDL моделі). Дані у базі фактів опрацьовують за допомогою логічних висновків, з використанням доступних знань для формування висновку. Як результат, база фактів оновлюється новими фактами та значеннями атрибутів. Далі здійснюється процес моделювання ситуації, для того щоб ідентифікувати проблемну ситуацію. Він також використовує знання про правила ідентифікації ситуацій та властивості самих ситуацій. Ці дії відповідають другому рівню JDL моделі. На третьому рівні JDL моделі, основаному на ідентифікації ситуації та контекстному знанні про предметну область, плануються необхідні дії, створюються впливи та здійснюють прогнозування [4].

Формулювання мети статті

Метою цієї роботи є розроблення формальних моделей та фреймворків для подання та опрацювання знань у системах із СО (зокрема алгебраїчних фреймворків), а також побудова відображення із алгебраїчної моделі в інші моделі та використання різних методів у межах єдиної СО системи.

Виклад основного матеріалу

Використання формальних методів подання знань, в основу яких покладена строга математична теорія, є доволі поширеним явищем. Класичними прикладами формальних методів є обчислення висловлювань, числення предикатів першого порядку, теорія множин, формальні методи, основані на алгебраїчному підході.

Подання даних у вигляді алгебраїчних структур є природним для багатьох спеціалістів у галузі інформаційних технологій. Наприклад, починаючи з відомої роботи Кодда [5], застосування алгебраїчних методів поширилось на моделювання баз даних. Об'єктами дослідження у його роботах стали алгебраїчні системи з операціями над відношеннями реляційної алгебри. Суть реляційного підходу до моделювання даних полягає у тому, що будь-який стан можна подати у вигляді відношення (таблиць), а опрацювання даних зводиться до операцій над цими відношеннями.

Алгебра – це математична структура, що містить множини операторів та операндів. Це можна записати у вигляді кортежу:

$$A = (\{Operands\}, \{Operators\}). \quad (1)$$

Взагалі алгебра постає ефективно-репрезентативною мовою опису систем, оскільки оператори можна переписати на основі формальних виразів, що ґрунтуються на таких формальних властивостях, як асоціативність чи комутативність. Операнди ж можуть кодувати інформаційний зміст алгебраїчних виразів, використовуючи аббревіатури та константи.

У роботах [6–8] розглянуто фреймворк “Алгебра систем” (Algebra of Systems), що дає змогу подати складні системи та автоматизувати розв’язання задач. Цей фреймворк ґрунтується на багатосортній алгебрі [9], оскільки домен складається з підмножини різних алгебраїчних доменів.

Розглядаючи поняття системи абстракції, необхідно взяти до уваги два аспекти: що є системою і що ця система виконує. Алгебра систем (AoS) використовує абстракції даних для параметризації властивостей систем.

Алгебру систем формально можна записати як:

$$AoS = \langle \{P, B, C\}, \{encd, enum, eval\} \rangle, \quad (2)$$

де P є властивістю домена; B – булеве значення і C – вміст домена. $Encd$, $enum$, $eval$ – операції кодування, підрахунку та оцінки. Елементи області композиції відображають процес перетворення об'єкта під час моделювання (рис.3) у вигляді дводольного графу.



Рис. 3. Відображення перетворення об'єкта у вигляді графу

Алгебра систем як алгебраїчний фреймворк моделювання дає змогу подати як декларативні дані, так і функції абстракції. Алгебра систем є метамовою, що підтримує базову множину абстракцій і може бути підлаштована для різних областей застосування.

На рис. 4 показано процес моделювання складних систем з використаннями фреймворка “Алгебра систем”.

Цей процес можна подати у вигляді послідовності перетворень доступних (наявних) знань. На рис. 4 артефакти подано у вигляді прямокутників, а операції – у вигляді овалів. Операція кодування спрямована на створення алгебраїчної моделі з доступних знань. В результаті створюються моделі підрахунку. Операція обчислення створює нові моделі та дає змогу отримати сформовані підмоделі, повторно їх оцінюючи за допомогою операції оцінки. Відповідні та ефективні моделі використовують для створення нових та оновлення уже наявних знань.

В основу процесу оцінювання та ідентифікації ситуації покладемо онтологічне моделювання предметної області з використанням апарату алгебри систем.

Нехай у предметній області існує n множин об'єктів:

$$A_1, A_2, \dots, A_n. \quad (3)$$

Об'єкти, що належать кожній множині, класифікують як екземпляри конкретного поняття. Ці множини є множинами-носіями для n багатосортних алгебр. Окремі екземпляри, що належать цим

множинам, позначатимемо a_1, \dots, a_n . Екземпляр об'єкта може належати декільком множинам одночасно. Екземпляри об'єктів формують базу фактів.

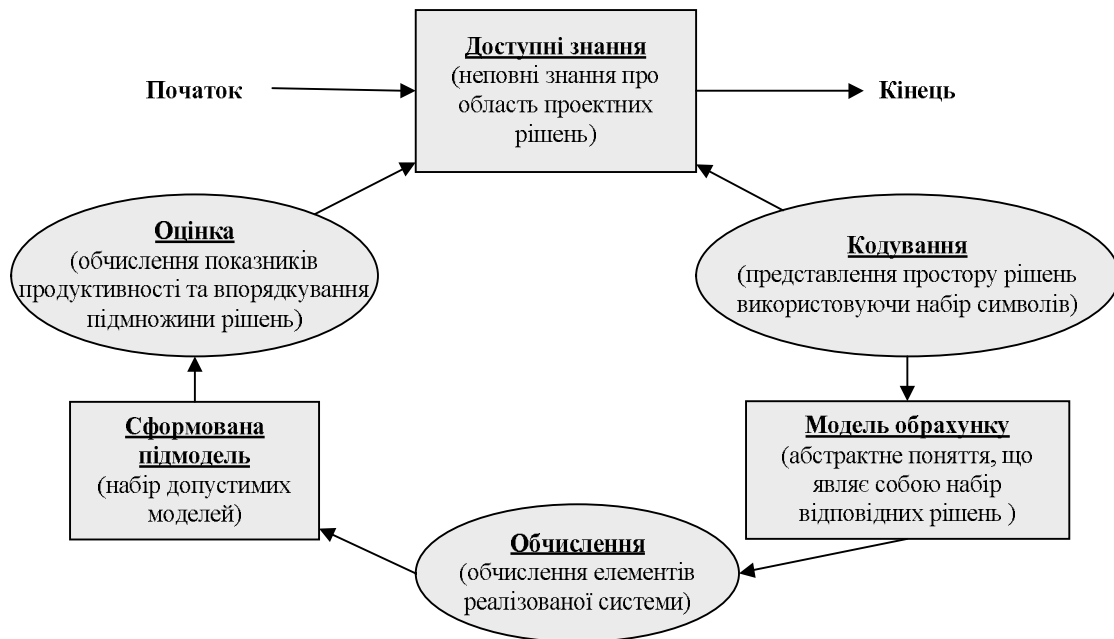


Рис. 4. Процес моделювання системи з використанням знань

Домен концептів. На основі кожної множини A_i визначимо абстрактний тип даних E_i :

$$E_i = (Name, A_i, EOP), \quad (4)$$

де $Name$ – назва типу; A_i – множина екземплярів; EOP – множина операцій. До цієї множини зарахуємо операції теорії множин.

Домен атрибутів At . Визначимо алгебраїчний домен атрибутів At на списку значень атрибутів у вигляді пар ($key, value$). Елемент пари key визначає ідентифікатор атрибуту, а $value$ – його значення. Для цього домена визначено операції об'єднання, підстановки, видалення, інтерпретації $\{merge, substitute, delete, interp\}$.

Булевий домен Cs . До булевого домена належать вирази, результат підрахунку яких належить булевій множині $\{true, false\}$. Операндами виразів є константи та змінні, що належать іншим алгебраїчним доменам. Будемо інтерпретувати елементи булевого домена як аксіоми Cs . Операціями для булевого домена є булеві операції $\{and, or, negation\}$, а також операція інтерпретації $interp$, яка відображає спрощення та підрахунок булевих виразів. Екземпляром булевого домена є конкретна аксіома. Аксіома повинна бути істиною для всіх фактів концепту, для якого вона задана.

Домен концептів з атрибутами T . Визначено на множині кортежів $\langle E_i, At_j, Cs_i^* \rangle$. Для кожного i існує тільки один j , що входить в елемент цього домена:

$$\forall i \exists! j : \langle E_i, At_j \rangle \quad (5)$$

Кожна аксіома $Cs_j \in Cs_j^*$ є виразом з операндами, що належать домену At_j . Операціями над елементами цього домена є об'єднання та поділ сутностей $\{merge, split\}$. Операція об'єднання сутностей інтерпретується як формування нової сутності як спільного набору властивостей та обмежень сутностей-операндів. Операція поділу сутності зворотна до об'єднання.

Домен відношень Rl . Множиною-носієм цього домена є структури виду:

$$\{(T_{1,i} \times T_{2,i} \times \dots \times T_{k,i}, At_i^r, Cs_i^*)\} \quad (6)$$

Кожна структура є кортежем, що містить декартовий добуток алгебраїчних типів даних з домена T , тип з домена атрибутів At (визначає атрибути відношення) та множина аксіом Cs^* . Кожна аксіома $Cs_i \in Cs_i^*$ є булевим виразом з операндами, що належать доменам $At_{1,i}, At_{2,i}, \dots, At_{k,i}, At_i^r$.

Над відношеннями визначено операції об'єднання відношень, відокремлення, підстановки $\{merge, split, substitute\}$. Операції об'єднання та відокремлення відношень трактують подібно до аналогічних операцій з домена T . В операції підстановки замість однієї сутності домена T підставляємо відношення, яке трактується як сутність з атрибутами (реіфікація).

Онтологія є кортежем, поданим доменами понять (концептів) з атрибутами, відношень та аксіомами з булевого домена, що стосуються доменів понять та відношень:

$$On = \langle T, Rl, Cs^* \rangle \quad (7)$$

Запропонована алгебраїчна модель представлення знань про предметну область має певні переваги порівняно із дескриптивною логікою та інтерпретованими системами:

- за допомогою відображень можна побудувати перехід із однієї моделі в іншу, а також поєднувати різні методи такого переходу, такі як, наприклад, моделювання та логічне виведення;
- може бути використана із різними методами моделювання, накопичуючи результати моделювання для прийняття рішення;
- ця модель відображає представлення предметної області, що використовується у концептуальному моделюванні;
- запропонована алгебраїчна модель також відповідає представленню опрацювання знань у системах із ситуаційною обізнаністю JDL моделлю (модель синтезу даних).

Подання ситуації на базі алгебраїчної моделі

Ситуаційна обізнаність ґрунтується на поданні та аналізі ситуацій. Поняття ситуації проаналізували та формалізували Джон Бервайс та Джон Перрі у класичних роботах [10–12].

Згідно з теорією Дж. Бервайса, ситуації доцільно розглядати як фізичні або абстрактні об'єкти, допустиме також визначення типів ситуацій.

Для математичного подання ситуацій Дж. Бервайс запропонував інфони. Інфон (infn) є теоретико-ситуаційним об'єктом, що містить відношення, відповідну кількість аргументів і позитивну або негативну полярності:

$$(Rl_i, t_{1i}, \dots, t_{ni}, p), \dots, (Rl_j, t_{1j}, \dots, t_{kj}, p), \quad (8)$$

де Rl_i, Rl_j – відношення, значення p – полярність і t_1, \dots, t_n – відповідні концепти з домена T для Rl . Отже, ситуація визначена як набір відношень. Полярність відображає істинність чи хибність інфону. Для відображення відношення між реальною ситуацією s та її поданням у вигляді інфону σ використовується відношення *Підтримує* (*Supports*). Наприклад, якщо s підтримує σ :

$$s \text{ a } \sigma. \quad (9)$$

У межах алгебраїчної моделі ситуації є нащадками загального концепту *Ситуація* (домен T). Кожна ситуація у списку атрибутів має атрибут *Визначення*, що подано як список посилань на об'єкти з домена відношень Rl , кожне з яких є описом інфону. Розвиток ситуації простежують, перевіряючи аксіоми, закріплені за кожним інфоном. Аксіоми також використовують і для генерування дій.

Використання дескриптивної логіки з алгебраїчною моделлю у системах із СО

Дескриптивна логіка використовується для побудови логічного висновку в системах із використанням онтологічного підходу до подання знань. В її основу покладено логіку предикатів першого порядку. Дескриптивна логіка популярна, оскільки широко використовується для побудови онтологій та є основою для стандартних мов опису (OWL).

Дескриптивна логіка використовується у системах, основаних на знаннях для:

- перевірки належності певного об'єкта;

- наслідування властивостей від інших об'єктів;
- перевірки відношення узгодженості;
- побудови нових концептів за допомогою конструктора аксіом.

Коли побудова логічних висновків за допомогою дескриптивної логіки завершена, інформаційна база змінюється та з'являються нові факти, значення атрибутів, факт відношень чи змінюються наявні факти.

Дескриптивна логіка розглядає окремі об'єкти предметної області як єдине ціле. Концепт називається групою об'єктів. Будь-який вказаний об'єкт може належати декільком концептам одночасно. Бінарні відношення (ролі) між об'єктами характеризують залежності між цими об'єктами. Аксіоми дескриптивної логіки використовуються для побудови нових концептів та представляють залежності між ними. В основу логічних висновків також покладено аксіоми. Дескриптивна логіка використовує логіку першого порядку та теорію множин у всіх сферах застосування.

Використання логічного міркування є важливою частиною будь-якої системи бази знань, оскільки дає змогу підтримувати логічну стійкість моделі предметної області та коректність даних. Незважаючи на популярність дескриптивної логіки, у неї є низка недоліків. Наприклад, спрощена модель дескриптивної логіки, що ґрунтується на логіці першого порядку, не дає змоги здійснити інші типи міркувань (індуктивні, статистичні, міркування, основані на аналогії), корисні для когнітивної діяльності людини.

Запропонована алгебраїчна модель, як і інші моделі, що використовуються у системах із ситуаційною обізнаністю, спирається на поняття концепту як типу об'єкта, де концепт визначається його атрибутами та відношеннями між іншими концептами.

Для представлення алгебраїчної моделі предметної області за допомогою дескриптивної логіки використовується маркування між цими двома представленнями.

Проте не всі об'єкти в алгебраїчній моделі (АМ модель) повинні бути відображені у модель дескриптивної логіки (DL модель), а лише ті, що використовуються у аксіомах та відношеннях успадкування, оскільки інші об'єкти не підлягають логічному міркуванню. Відображення повинне бути ін'єктивним, щоб після початкового відображення АМ моделі у DL модель та після побудови всіх інших необхідних відображень результат можна було відобразити у зворотному напрямку, тобто з DL моделі у АМ модель.

Формально дескриптивна логіка – це кортеж вигляду (N_C, N_R, N_O) , де N_C – набір імен концептів, N_R – набір імен відношень та N_O – набір імен об'єктів.

База знань (БЗ) – це впорядкована пара (T, A) , де T – це набір ТВох аксіом та A – це набір АВох аксіом.

Процедура відображення алгебраїчної моделі у дескриптивну логіку містить такі кроки:

1. Визначення аксіом $Csl \hat{I} Cs^*$, пов'язаних із концептами $Tl \hat{I} T$, та відношеннями $Rll \hat{I} Rl$, що використовує алгебраїчна модель. Як результат, створена онтологія: $(Tl, Rll, Csl) \subseteq On$.

2. Відображення Onl елементів у DL об'єкти, концепти, ролі та аксіоми. Як результат, DL модель побудовано:

• передусім відображаються об'єкти. Множина імен об'єктів визначається на основі об'єктів у АМ моделі, що належать концептам у Tl :

$$N_o = Al = \bigcup_{i=1}^n Al_i \quad (10)$$

Та $\forall al_i \in Al_i : Type(Al_i) = El_i$ і El_i є частиною концепту $Tl_i, Tl_i \in Tl$;

• наступними відображаються концепти. Множина концептів N_c відповідає множині Tl . Кожному об'єкту al_i ставиться у відповідність аксіома-твердження, що стосується його концепту Tl_i . Множина цих аксіом є множиною АВох аксіом у базі знань;

• для кожного атрибута $At_j \hat{I} Tl_i$ створюється відповідне відношення (роль) rla_i .

$$Rla = \bigcup_{i=1}^n rla_i ; \quad (11)$$

Ї кожне відношення rl_i ; $\hat{I}Rll$ розділяють на бінарні відношення. Множина відношень, отриманих у результаті, $i \in Rlr$. Створено множину ролей у дескриптивній логіці:

$$N_R = Rla \cup Rlr ; \quad (12)$$

Ї на додаток, аксіоми відображають відношення успадкування, а також інші аксіоми із АМ моделі, які можуть бути подані як логічні вирази у DL моделі.

3. Виконати процедури міркування, застосовуючи DL модель.

4. Оновити АМ модель, використовуючи відображення і результати міркування.

Розширене відображення моделей виконується тільки один раз на початковому етапі під час побудови моделі. Надалі, працюючи, системи із ситуаційною обізнаністю тільки відображають оновлені компоненти між моделями.

Використання апарату інтерпретованих систем з алгебраїчною моделлю у системах із ситуаційною обізнаністю

Підхід інтерпретованих систем (ІС) широко застосовується для подання часової динаміки та розумування у мультиагентних системах із ситуаційною обізнаністю [3]. Аналіз ситуації у таких системах відбувається на основі перевірки часозалежних тверджень із бази знань. Це дає змогу не тільки враховувати динаміку змін стану предметної області, але й моделювати процеси прийняття рішень агентами в нечітких умовах, використовуючи числові міри ймовірності, можливості, нечіткості.

Модель інтерпретованих систем визначає стани та переходи між ними у дискретному часі. Для підтримки відображення між алгебраїчною моделлю та інтерпретованою системою факти АМ повинні також містити часовий параметр як один з атрибутів. Формалізм інтерпретованих систем оснований на таких групах понять:

- стани, історії, системи;
- відрізки, дії, протоколи;
- інтерпретована система.

Розглянемо їх, а також відображення у АМ окремо.

Стани, історії та система

Мультиагентну систему визначено як множину агентів $Ag = \{ag_1, \dots, ag_n\}$ та середовище $En = \{en\}$, яке розглядається як окремий різновид агента. Глобальний стан системи визначено як:

$$(St_{en}, St_{ag1}, \dots, St_{agn}), \quad (13)$$

де St_{en} – це стан середовища, а St_{agi} – стан агента. Відповідно, множина усіх глобальних станів визначена як:

$$G : L_{en} \times L_1 \times \dots \times L_n, \quad (14)$$

де L_{en} – це множина усіх можливих станів середовища, а L_i – множина усіх можливих станів агента i .

Для подання змін станів системи у часі використовується поняття історії (run) R як функції-відображення між часом та множиною глобальних станів G , $R: T \rightarrow G$. Множина моментів часу є дискретною та подається як множина цілих додатних чисел. У системі може бути багато різних історій.

Стан St (point) – це стан з історії r у певний момент часу m . Він подається парою (r, m) .

Система Sy визначається як множина усіх можливих історій у G .

Для відображення станів та історій в алгебраїчній моделі визначено концепт *Агент* (домен T), що відповідає агентам ІС. Між іншими концептами АМ, які можуть відігравати роль агента, та концептом *Агент* встановлюється відношення *БутиАгентом*. Середовище подається усіма іншими концептами АМ.

Стани агентів та середовища визначаються фактичними станами екземплярів агентів та середовища у визначені моменти часу. Аналогічно, історія $r(m)$ подається як послідовність станів системи до визначеного моменту часу m .

Система визначається як множина можливих історій.

Відрізки, дії та протоколи

Відрізок історії $rn(m)$ – це частина історії між двома сусідніми моментами часу $(m-1)$ та m .

Агенти та середовище змінюють глобальний стан у результаті виконання дій протягом відрізків історії. Нехай Act_i – множина дій агента i та Act_e – множина дій середовища.

Сумісна дія $GAct$ – це кортеж з діями агентів та середовища $(ac_e, ac_1, \dots, ac_n)$, де $ac_e \in Act_e$ та $ac_i \in Act_i$.

У результаті виконання сумісної дії змінюється глобальний стан системи, що відображено функцією переходів:

$$Tr : (G, GAct) \rightarrow G \quad (15)$$

Агенти визначають, які дії виконувати, з використанням протоколу – набору правил вибору дій:

$$Pr_i : St_{agi} \rightarrow Act_i \quad (16)$$

Протоколи різних агентів узагальнюються у понятті сумісного протоколу $Pr = (Pr_1, \dots, Pr_n)$.

В інтелектуальній системі із ситуаційною обізнаністю доцільно пов'язати дії агентів із ситуаціями, в яких ці агенти перебувають. Отже, виконуючи відображення між АМ та ІС, для кожного агента визначають ситуацію, за його фактичним станом, історією та глобальним станом системи. Для кожної ситуації визначено набір дій агента. Для мультиагентної системи визначення ситуації [8] здійснюється в контексті кожного агента. Правила дій, відображені протоколом в ІС для АМ, замінюються правилами ідентифікації ситуацій та правилами дій у кожній ситуації.

Інтерпретована система

Нехай Φ – множина логічних тверджень на G , що описує факти та залежності у системі. Тоді інтерпретована система I – це пара:

$$(Sy, p), \quad (17)$$

де Sy – це система на множині глобальних станів G , а p – це інтерпретація Φ на G . Для кожного $p \in \Phi$ та $g \in G$ інтерпретація визначає істинність або хибність твердження:

$$p(g)(p) \in \{true, false\} \quad (18)$$

Для формування часозалежних логічних висновків використовують часові оператори \square (завжди), \diamond (з часом, колись), O (у наступний момент), U (поки).

В АМ логічні твердження належать булевому домену Cs . Між Φ та булевими виразами домена Cs існує взаємно однозначне відображення, що дає змогу застосувати методи дослідження та розумування ІС для систем, поданих алгебраїчною моделлю у вигляді онтології.

Отже, використання алгебраїчної моделі для подання знань в системі із ситуаційною обізнаністю дає змогу однозначно відобразити модель системи, подану у вигляді інтерпретованої системи, та застосувати цей апарат для моделювання динаміки станів системи.

Висновки

Незважаючи на популярність наукового напрямку ситуаційної обізнаності та наявність великої кількості робіт, дослідження СО залишаються актуальними. Це пояснюється тенденціями у системах підтримки прийняття рішень, розвитком автономних інтелектуальних систем прийняття рішень, розширенням потреб у ситуаційній обізнаності для таких галузей, як бізнес-аналітика тощо.

Напрямок ситуаційної обізнаності ретельно досліджено, в результаті чого запропоновано декілька моделей представлення знань та методів моделювання ситуаційної обізнаності.

Моделювання та побудова інтелектуальних систем із ситуаційною обізнаністю потребують розроблення формальних методів для подання та опрацювання знань. Проте все ще не вирішеною проблемою залишається побудова єдиного фреймворка, що дає змогу поєднувати декілька різних методів моделювання у межах єдиної системи із ситуаційною обізнаністю.

У роботі здійснено такі дослідження та отримано такі результати:

- розглянуто та проаналізовано формальні моделі подання знань у системах із ситуаційною обізнаністю. Розгляд саме формальних моделей і побудова на їх основі відображень дадуть змогу побудувати єдиний фреймворк для моделювання різних методів у межах єдиної СО системи;
- проаналізовано подання ситуації на основі алгебраїчної моделі. Запропонований алгебраїчний підхід на базі алгебри систем відповідає загальній моделі JDL, що має достатню гнучкість і дає змогу використовувати для моделювання та розв'язання задач інші математичні методи за допомогою побудови взаємно однозначних відображень;
- розроблено методи відображення алгебраїчної моделі у модель дескриптивної логіки та інтерпретованих систем. Ці методи дають змогу використовувати методи логічного виведення та моделювання, розроблені у цих підходах, для дослідження систем із ситуаційною обізнаністю, поданих у вигляді онтології.

Подальші дослідження будуть спрямовані на розроблення методів ідентифікації наближених ситуацій, що, на відміну від вже відомих, дають змогу комбінувати різні механізми ідентифікації проблемної ситуації, зокрема причинно-наслідковий та параметричний.

1. Endsley M. *Theoretical underpinnings of situation awareness: a critical review* *Process More Data ≠ More Information* / M. Endsley, R. Mica // Edited by Array. Most, Vol. 301, 2000. – P. 3–32.
2. BeAware!-Situation awareness, the ontology-driven way / N. Baumgartner [and other] // *Data and Knowledge Engineering*, Vol. 69, 2010. – P. 1181–1193.
3. Joussetme A. *Interpreted Systems for Situation Analysis* / Joussetme Anne Laure, Patrick Maupin // *10th International Conference on Information Fusion*. – IEEE, 2007. – P. 1–11.
4. Steinberg A. N. *Revisions to the JDL Model* / Steinber A. N., Bowman C. L., White F. E // *Joint NATO/IRIS Conference Proceedings*. – Quebec, 1998 and in *Sensor Fusion: Architectures, Algorithms, and Applications, Proceedings of the SPIE*, Vol. 3719, 1999. – P. 430–441.
5. Codd E. *Relational Completeness of Data Base Sublanguages* / E. F. Codd. – IBM corporation, Vol. 54, 1972. – P. 65 – 98.
6. Koo B. *Algebra of Systems: A metalanguage for Model Synthesis and Evaluation* / B. Koo, W.L. Simmons, E.F. Crawley // *IEEE Transactions On* 39.3, *Systems, Man and Cybernetics, Part A: Systems and Humans*, 2009. – P. 501–513.
7. Koo B. *Meta-language for Systems Architecting* / B. A. Koo // *In Engineering Systems and Management Massachusetts Institute of Technology*, 2005. – P. 168.
8. Koo B. *A valuation technology for product development option using an executable meta-modeling language in Complex Systems Concurrent Engineering* / B. Koo, W. L. Simmons, E. F. Crawley // *Collaboration, Technology Innovation and Sustainability*. New York: Springer-Verlag, 2007. – P. 107–115.
9. Burris S. *A Course in Universal Algebra* / S. Burris, H. P. Sankappanavar. – New York: Springer-Verlag, 1981. – P. 129–154.
10. Barwise J. *Situation Theory and Its Applications* / Jon Barwise // *Center for the Study of Language (CSLI)*, Vol. 26, 1991. – P. 631.
11. Devlin K. *Situation theory and situation semantics* / K. Delvin. – *Handbook of the History of Logic*, vol. 7, 2006. – P. 601–664.
12. White F. E. *A Model for Data Fusion* / F. E. White // *Proc. 1st National Symposium on Sensor Fusion*. – 1988. – P. 153–158.