

ВИКОРИСТАННЯ АДАПТИВНИХ ОНТОЛОГІЙ В ІНТЕЛЕКТУАЛЬНИХ СИСТЕМАХ ПРИЙНЯТТЯ РІШЕНЬ

В. В. Литвин

Кандидат технічних наук, доцент
Кафедра інформаційних систем та мереж*
E-mail: vasyll@ukr.net

В. Я. Крайовський

Проректор з господарської роботи*
Контактний тел. 8 (032) 258-24-04

Н. Б. Шаховська

Кандидат технічних наук, доцент, заступник декана*
Контактний тел.: 8 (032) 258-24-04
E-mail: natalya233@gmail.com
*Національний університет «Львівська політехніка»
вул. С.Бандери, 28, Львів-13, 79013

Стаття присвячена функціонуванню інтелектуального агента. Для моделювання поведінки інтелектуального агента використано мережі Петрі і метод онтології. Показано, що модель поведінки, що розвивається, може бути перетворена в асинхронному завданні динамічного програмування.

Ключові слова: онтологія, інтелектуальний агент, система прийняття рішень

Статья посвящена функционированию интеллектуального агента. Для моделирования поведения интеллектуального агента использована сеть Петри и метод онтологии. Показано, что модель поведения, которое развивается, может быть преобразована в асинхронную задачу динамического программирования

Ключевые слова: онтология, интеллектуальный агент, система принятия решений

Article considers intelligent agent functioning plan development. For intelligent agent behavior modeling Petri nets and ontology method were used. It was shown that developed behavior model could be transformed into asynchronous dynamic programming task

Keywords: ontology, intelligent agent, system of making decision

1. Вступ

На даному етапі розвитку інформаційних технологій, а саме систем підтримки прийняття рішень, розрізняють два напрями розвитку інтелектуальних систем прийняття рішень, які базуються на знаннях [1]:

- інтелектуальні системи прийняття рішень, засновані на прецедентах;
- інтелектуальні системи прийняття рішень, засновані на плануванні.

Згідно із сучасною термінологією, процес прийняття рішення (логічного виведення) у таких системах здійснює інтелектуальний агент (ІА).

Практично всі ранні експертні системи моделювали хід ухвалення рішення експертом як дедуктивний процес з використанням логічного виведення, заснованого на правилах. Це означало, що в систему закладалися сукупність правил вигляду “якщо...то...”, згідно з якими на підставі вхідних даних генерувався той або інший розв’язок вирішення задачі, що цікавив користувача (такий підхід є частковим випадком планування діяльності ІА). Однак навколишній світ складний. Існує ряд задач, для яких ніколи не буде знайдено формального розв’язку (судочинство, меди-

цина). Актуальність проблеми обумовлена і чисельністю таких задач, і практичною потребою знайти хоча б один який-небудь відповідний розв’язок там, де через відсутність строгого формалізованого методу не можна знайти всі або найоптимальніші розв’язки.

Щоб вирішувати кожен задачу, виходячи з первинних принципів, експерт часто аналізує ситуацію в цілому і згадує, які рішення ухвалювалися раніше в подібних ситуаціях. Потім він або безпосередньо використовує ці рішення, або, при необхідності, адаптує їх до обставин, що змінилися для конкретної проблеми. Моделювання такого підходу до вирішення проблем, заснованого на досвіді минулих ситуацій, привело до появи технології логічного виведення, заснованого на прецедентах (англійською – Case-Based Reasoning, або CBR), і надалі – до створення програмних продуктів, що реалізують цю технологію. У ряді ситуацій метод виведення за прецедентами має серйозні переваги в порівнянні з виведенням, що ґрунтується на плануванні, зокрема правилах, і особливо ефективний, коли:

- основним джерелом знань про завдання є досвід, а не теорія;
- рішення не унікальні для конкретної ситуації і можуть бути використані в інших випадках;

- метою є не гарантоване вірне рішення, а краще з можливих.

Таким чином, виведення, засноване на прецедентах, є методом побудови інтелектуальних систем, які приймають рішення щодо даної проблеми або ситуації за наслідками пошуку аналогій, що зберігаються в базі прецедентів. На наш погляд, пошук таких аналогій доцільно здійснювати на основі онтологій предметних областей (ПрО) [2] в яких функціонує ІА.

З іншого боку ми часто зустрічаємось із задачами, де необхідно знайти послідовність дій, яка дозволяє досягнути кінцевого стану, який називають цільовим станом. Такий процес пошуку називається плануванням [3]. Задача побудови плану поведінки ІА є актуальною задачею, оскільки саме від правильно побудованого плану залежать кількісні показники, отримані під час досягнення цілі (затрачені ресурси, витрачений час для досягнення цілі, оцінка досягнутого стану, його корисність тощо). З літератури відомо, що для моделювання такого плану знаходження оптимального рішення використовуються стохастичні або детерміновані мережі з вершинами типу І/АБО і методи пошуку вглиб, вишир або інформативний пошук з використанням евристичних функцій [4]. Однак ці моделі можна використовувати для задач, в яких стани чітко задаються множиною фактів, які реалізуються однією з формальних логік або набором продукційних правил та не враховують витрату ресурсів.

Очевидно, що у складних прикладних областях опис станів та вибір альтернатив для відповідних переходів між станами вимагає зовсім інших підходів, а отже і моделей, які базуються на онтологіях задач та онтології ПрО. З огляду на постановку задачі – досягнення цільового стану ІА – для розв’язування цієї задачі запропоновано використовувати мережі Петрі [5] для моделювання шляхів (процесів) досягнення цільового стану, байєсівські мережі [3] для моделювання імовірнісних оцінок переходів між станами, онтологію ПрО для опису станів та обчислення необхідно затрачених ресурсів для переходів між станами.

Мета роботи: розробити моделі інтелектуальних систем прийняття рішень, які використовують онтології та методи функціонування таких систем.

2. Використання онтологій в інтелектуальних системах прийняття рішень, заснованих на прецедентах

Системи виведення за прецедентами показують дуже добрі результати в найрізноманітніших задачах, однак володіють рядом істотних недоліків. По-перше, вони взагалі не створюють яких-небудь моделей або правил, що узагальнюють попередній досвід, – під час вибору розв’язку вони ґрунтуються на всьому масиві доступних історичних даних, тому неможливо сказати, на основі якої конкретно інформації, системи виведення за прецедентами видають свої рішення.

Можна виділити дві основні проблеми, з якими зустрічаються подібні системи: пошук найбільш відповідних прецедентів і подальша адаптація знайденого розв’язку. У основі всіх підходів до відбору прецедентів лежить той або інший спосіб вимірювання ступеня близькості прецеденту і поточної ситуації. При таких вимірюваннях обчислюється чисельне значення де-

якої міри, що визначає склад множини прецедентів, які необхідно опрацювати. Основним недоліком таких систем є відсутність єдиної методики вибору міри близькості. Ще один недолік методу пов’язаний з конструкцією прецедентів і призначення ваг їх атрибутам, що зменшує застосовність таких систем у різних ПрО. Однак процес роботи таких систем вже добре відомий: шукаються відстані між поточною ситуацією та наявними прецедентами; вибирається той прецедент, для якого ця відстань є найменшою; виконується чи пропонується до виконання те рішення, яке відповідає цьому вибраному прецеденту.

Щоб формально це описати введемо такі позначення:

$Pr = \{Pr_1, Pr_2, \dots, Pr_N\}$ – множина прецедентів;

R_i – рішення, яке приймається згідно до прецедента Pr_i ;

S – поточна ситуація;

$d_i(Pr_i, S)$ – відстань між i -им прецедентом та поточною ситуацією;

o – оцінка прийнятого рішення.

У більшості випадків методи пошуку прецедентів зводяться до індукції дерев рішень або до алгоритму “найближчого сусіда” [6], доповненого, можливо, використанням знань про ПрО. Задачі адаптації і використання знайденого рішення до сих пір залишаються недостатньо формалізованими і сильно залежними від ПрО. Обидві проблеми – пошук прецедента і його адаптація – вирішуються (повністю або частково) із залученням фонових знань, іншими словами, використовуючи знання про ПрО (domain knowledge). На наш погляд, джерелом таких знань виступає онтологія ПрО.

Тому нами пропонується проектувати прецеденти та поточну ситуацію на онтологію ПрО; ввести в межах онтології ПрО метрику за допомогою якої шукати необхідну відстань (рис. 1). Крім того сама онтологія буде адаптуватися до ПрО за рахунок методів інтелектуального аналізу даних (ІАД). Щоб ці методи використовувались, нам необхідно вести історію функціонування інтелектуальної системи, тобто набутий нею досвід розв’язування подібних задач, який буде зберігатися у архіві даних.

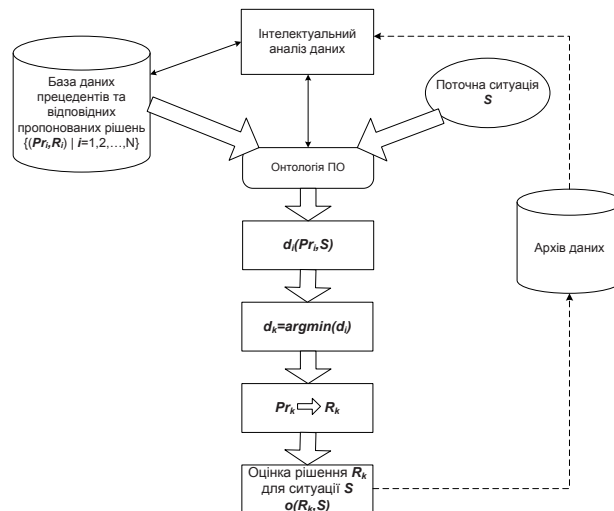


Рис. 1. Процес прийняття рішення на основі прецедентів з використанням онтологій

А саме у цьому архіві даних зберігаються прийняті системою рішення згідно до ситуацій та оцінка цих рішень. Для процедур ІАД використовуються тільки ті дані, для яких оцінка прийнятих рішень є високою, тобто є вищою за деякий поріг. Проблема оцінювання прийнятих рішень в цій роботі не розглядається. Для шкалювання оцінок прийнятих рішень використовуються математичні апарати, такі як теорія нечітких множин та відношень, лінгвістичні змінні тощо [7].

Ефективність адаптації онтології бази знань до особливостей предметної області визначають закладені в її структуру елементи та механізми її оптимізації (точніше адаптації) шляхом самонавчання під час експлуатації. Одним з підходів до реалізації таких механізмів є автоматичне зважування понять бази знань (БЗ) та семантичних зв'язків між ними під час самонавчання. Цю роль беруть на себе коефіцієнти важливості понять та зв'язків. Їх розподіл у БЗ має відповідати таким основним вимогам:

- відображати семантичну вагу понять ПрО, в якій ця інтелектуальна система реально застосовуватиметься;
- формуватися під час наповнення БЗ та коректуватися у відповідності з визначеними правилами;
- забезпечувати контроль цілісності БЗ;
- відповідати вимогам метрики при їх використанні для порівняння семантичної близькості понять.

Стоїть задача сформулювати відповідний набір правил присвоєння вагових коефіцієнтів (інформаційної ваги) поняттям та твердженням в моделі БЗ, що забезпечить оцінку актуальної цінності її інформаційного наповнення та досліджуваних поточних ситуацій (наприклад віднесення текстових документів до класів згідно УДК, визначення захворювання пацієнта).

Покажемо можливість вирішення сформульованої задачі шляхом введення деяких спрощень і припущень. Подамо БЗ у вигляді іменованого графа, числові семантичні характеристики вершин і ребер якого визначаються за певними правилами. Він є орієнтованим зв'язним мультиграфом з наступними властивостями:

- 1) у кожний елемент (вершину) може бути довільна кількість входів дуг графа;
- 2) кожний елемент може мати зв'язок з будь-якою кількістю інших елементів;
- 3) кожному зв'язку (ребру) у моделі відповідає певний напрям і коефіцієнт важливості зв'язку та достовірності відповідного твердження, кожному поняттю (вершині) – коефіцієнти важливості поняття.

Коефіцієнт важливості поняття (зв'язку) – це чисельна міра котра характеризує значимість даного поняття (зв'язку) у конкретній предметній області, і динамічно змінюється за певними правилами в процесі експлуатації системи [8].

Наш підхід до представлення знань у формі зваженої семантичної мережі (концептуальних графів) полягає у тому, що будь-яке можливе узагальнення, тобто комплексне, складене поняття завжди явним чином артикульоване, назване і як окремий концепт фігурує в базі знань. Тому якщо деяке узагальнення має спільні властивості чи способи функціонування, вони фізично можуть бути реалізовані через властивості та обробники подій відповідного узагальнюючого концепта.

Отже таку онтологію ми будемо визначати як п'ятірку:

$$O = \langle X, R, F, W, L \rangle,$$

де X – скінченна множина концептів (понять, термінів) предметної області, яку задає онтологія O ; R – скінченна множина відношення між концептами (поняттями, термінами) заданої предметної області; F – скінченна множина функцій інтерпретації (аксіоматизація), заданих на концептах або відношеннях онтології O ; W – важливість понять X , L – важливість відношень R .

Визначену таким чином онтологію будемо називати адаптивною, тобто такою що адаптується до ПО за рахунок модифікації понять та коефіцієнтів важливості цих понять та зв'язків між ними [9].

Один із способів визначення цих коефіцієнтів важливості та їх зміна будуть описані нижче. Тут тільки відзначимо, що зміна цих коефіцієнтів відбувається згідно модифікації знань методами ІАД. Метою технології видобування даних є виробництво нового знання, яке користувач може надалі застосувати для поліпшення результатів своєї діяльності. Можна виділити, принаймні, сім методів виявлення і аналізу знань: 1) класифікація; 2) регресія; 3) кластеризація; 4) аналіз асоціацій; 5) прогнозування тимчасових послідовностей (рядів); 6) агрегація (узагальнення); 7) виявлення відхилень.

Методи 1, 2, 4 і 5 використовуються, головним чином, для прогнозування, тоді як останні зручні для опису існуючих закономірностей в даних.

Очевидно, що кожний прецедент однозначно задається онтологією, яка є частиною загальної онтології і навпаки – онтологією однозначно визначається прецедент. Тобто між ними є однозначна відповідність, яку запишемо у вигляді:

$$Pr_i \Leftrightarrow O_i \subseteq O$$

Інтелектуальний агент ІАД модифікує онтологію на основі старої онтології та архіву даних, який зберігає відповідний ІА. Тобто:

$$IAD : \langle O, AD \rangle \rightarrow O$$

Розглянемо методи задання початкових коефіцієнтів онтології ІА та їх модифікацію згідно з рішеннями отриманими методами ІАД, а саме методом побудови дерева рішень [6]. Нехай онтологія містить множину понять $X = \{x_1, x_2, \dots, x_n\}$ та множину зв'язків $R = \{r_1, r_2, \dots, r_m\}$. $W = \{w_1, w_2, \dots, w_n\}$ – коефіцієнти важливості відповідних понять, $L = \{l_1, l_2, \dots, l_m\}$ – коефіцієнти важливості відповідних зв'язків. Спочатку вважаємо, що всі значення $w_i = 1, i = 1, 2, \dots, n; l_j = 0, j = 1, 2, \dots, m$.

Нехай у результаті побудови дерева рішень ми отримали k різних правил $Rule_1, Rule_2, \dots, Rule_k$. Кожне правило $Rule_s$ складається із підмножини понять, тобто $Rule_s = \{x_{s1}, x_{s2}, \dots, x_{st}\}$. Додаємо до значення w_i одиницю, якщо поняття x_i входить у будь-яке правило $Rule_s, s = 1, \dots, k$. Аналогічно, якщо між поняттями x_i та x_j існує зв'язок l_p , то значення l_p збільшуємо на одиницю, якщо обидва поняття x_i та x_j входять у одне правило. Тим самим ми збільшуємо важливість понять, які входять у правила та відповідних їм зв'язків. Це відповідає уявленню, що поняття (ознаки), які присутні у правилах є важливішими, ніж інші значення.

Нами пропонується визначати відстань між прецедентом і ситуацією, як суму відстаней між найважливішими поняттями прецедента та поточного випадку. Таких понять може бути одне, два; однак якщо їх є три або більше, то нами пропонується вибирати перші три. Ця кількість визначена на основі опитувань експертів

різних ПрО і вважається ними оптимальною. У такому випадку ми маємо 3 центри ваг прецедента і 3 центри ваг поточної ситуації, тобто 9 усіх можливих відстаней між ними. Вибираємо 3 найменші з них та їх сумуємо. Отримана таким чином сума й буде відстанню між прецедентом та поточною ситуацією. Очевидно, що визначена таким чином відстань залежатиме від того як ми визначимо відстань між двома суміжними вершинами. Для цього нами пропонується визначати відстані між вершинами, що з'єднані зв'язком як

$$d_{ij} = \frac{Q}{L_{ij}(W_i + W_j)}$$

де W_i та W_j – коефіцієнти важливості вершин C_i та C_j відповідно; L_{ij} – коефіцієнт важливості зв'язку між вершинами; Q – константа, яка залежить від конкретної онтології. Приймемо, що $L_{ii} = \infty$, тоді $d_{ii} = 0$.

Далі знаходимо центри ваг концептуального графа. Це перші три вершини для яких середня відстань \bar{d}_i є найменшою:

$$\bar{d}_i = \min_i \bar{d}_i$$

Середня відстань \bar{d}_i для вершини C_i обчислюється згідно формули:

$$\bar{d}_i = \frac{\sum_{j=1, j \neq i}^n d_{ij}^*}{n-1}$$

де n – кількість вершин графа, d_{ij}^* – найкоротший шлях між вершинами C_i та C_j , який обчислюється за допомогою відомих алгоритмів, наприклад Форда, Дейкстри, Флойда-Уоршалла [10].

Далі згідно концептуального графа, що задає онтологію прецедента, шукаємо відстань від даного прецедента до поточної ситуації. Якщо поняття поточної ситуації не входять в концептуальний граф, то онтологію даного прецедента доповнюємо онтологією всього ІА до якого входить цей прецедент. Якщо ж необхідне поняття далі не входить в онтологію ІА, то його відсутність зумовлює ріст відстані до безмежності, що означає не близькість прецедента із поточною ситуацією. Зазначимо, що запропонована таким чином відстань задовольняє трьом аксіомам метрики. Дійсно, згідно визначення відстані, автоматично виконуються дві перші аксіоми: $d(C_i, C_i) = 0$; $d(C_i, C_j) = d(C_j, C_i)$.

Нехай R_{ij}^* – шлях між вершинами C_i та C_j , який відповідає відстані між ними. Тоді $d_{ij} = d_{ik} + d_{kj}$, якщо вершина C_k лежить на шляху R_{ij}^* і $d_{ij} < d_{ik} + d_{kj}$, якщо вершина C_k не лежить на шляху R_{ij}^* . А це означає, що виконується третя аксіома метрики.

Відкритим залишається питання як часто використовувати роботу агента ІАД для модифікації онтології ІА. Тобто агент ІАД виступає ще в ролі координатора, а питання моментів втручання є задачею координації [11]. У цій роботі дане питання не розглядається.

3. Використання онтологій в інтелектуальних системах прийняття рішень, заснованих на плануванні

Для досягнення цільового стану інтелектуальний агент (ІА) насамперед повинен промодельовати план

досягнення цього стану зі всіма можливими альтернативами. Процес планування ґрунтується на декомпозиції. Задача планування ZP містить 3 складові: множину станів St, множину дій F, множину цілей Z; тобто $ZP = \langle St, F, Z \rangle$.

У свою чергу дія складається із 3-ох частин: ім'я дії і список параметрів, передумова та результат, а сам план визначається як кортеж з 4-ох елементів – \langle множина дій, множина обмежень впорядкування, множина причинних зв'язків, множина відкритих передумов \rangle [3]. Для врахування декомпозиції, і/або залежностей між станами та переходами, відображення альтернатив досягнення цільових станів пропонуємо використовувати мережі Петрі. Приклад такої мережі з цільовим станом $St(z)$ наведено на рис. 2.

Нехай біжучий стан ІА у просторі станів, змодельованому мережею Петрі, відображається деякою фішкою, яка в певний момент часу може знаходитись на одному з вузлів мережі. На початку фішка мережі знаходиться у стані $St(0)$.

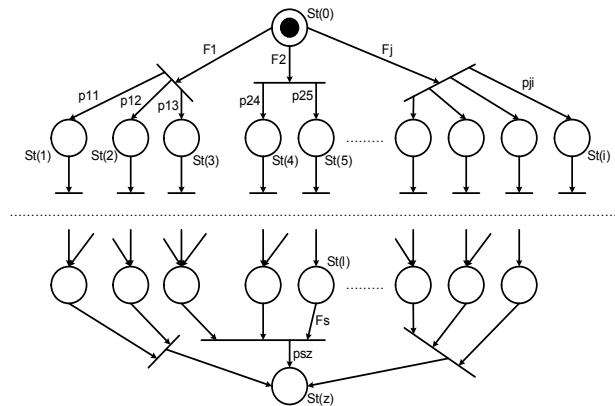


Рис. 2. Приклад мережі Петрі побудови плану діяльності ІА

Тоді очевидно, що суть функціонування ІА полягає у переміщенні фішки в кінцевий стан $St(z)$ з мінімальною витратою ресурсів. Щоб почати розв'язувати цю задачу слід довести, що мережа Петрі побудована так, що кінцевий стан $St(z)$ можна досягнути. Відомо, що для такого доведення використовуються планувальники – програми, які шукають розв'язок або доводять неіснування розв'язку. Ця робота оминає розгляд такого доведення. Априорно вважаємо, що такий перехід існує і не один, в іншому випадку задача немає змісту.

Стан $St(i)$ характеризується базою даних та базою знань (сховищем даних), поданих у вигляді множини фактів з відповідними імовірнісними оцінками. Дія F_{ij} подається у вигляді відображення зі стану $St(i)$ в стан $St(j_i)$ з відповідною ймовірністю p_{ji} , тобто: $St(i) \rightarrow St(j_i)$ з ймовірністю p_{ji} .

Для вибору необхідних дій користувач повинен вміти оцінювати стани: як ті, в яких він вже перебував, так і бажані. Нехай $q(St(i))$ – оцінка стану $St(i)$. Для оцінки станів в яких вже перебував раціональний агент, використовуватимемо онтологію предметної області О. Вважатимемо, що онтологія однозначно визначає стан $O(St(i))$, який за певними значеннями відрізняється від реального опису стану. Тоді оцінку стану визначимо як обернену пропорційну величину

до відстані між описом згідно онтології та реальним описом стану:

$$q(\text{St}(i)) = \text{const}/d(\text{O}(\text{St}(i)), \text{St}(i)),$$

де const – коефіцієнт пропорційності, $d(\text{O}(\text{St}(i)), \text{St}(i))$

– відстань між ідеальним описом та реальним описом стану $\text{St}(i)$. Для обчислення цієї відстані пропонується спосіб, аналогічний запропонованому вище у цій статті – для обчислення відстані між прецедентами.

Для оцінки стану, в який ІА ще не потрапляв, використовуватимемо евристичні функції. Як відомо, загального визначення евристик не існує, а таке визначення строго прив'язується до предметної області. Тому оцінка майбутніх станів є складною задачею. У наших дослідженнях для вибору дій ІА ми спиратимось на раціональність агента, як прагнення мінімізувати витрати ресурсів для досягнення кінцевого стану. Тому вважатимемо, що дія F_j однозначно визначається витратами ресурсів R_{ijk} (ціна переходу зі стану в стан), де $k=1,2,\dots,n_i$, n_i – кількість альтернатив a_k для здійснення переходу F_{ij} .

Розглянемо функціонування ІА на основі оцінки витрат ресурсів. Інформація про альтернативи та необхідний для цього ресурс знаходиться в літературних джерелах Т.

Кортеж $\langle d(\text{O}(\text{St}(j_i)), \text{St}(j_i)), \text{St}(j_i), R_{ijk}, o(\text{St}(j_i)) \rangle$ визначає чи робити нам перехід F_{ij} , використовуючи альтернативу a_k .

Для обчислення необхідних ресурсів R_{ijk} використовуємо онтологію літературних джерел $\text{O}(T)$ [12] та онтологію станів $\text{O}(\text{St}(i))$. А саме:

$$\langle \text{O}(T), \text{O}(\text{St}(i)) \rangle \rightarrow R_{ijk},$$

$$\langle \text{O}(F_j(\text{St}(i))), \text{O}(\text{St}(j_i)) \rangle \rightarrow d(\text{O}(\text{St}_k(j_i)), \text{St}(j_i)).$$

Мета досягнута, якщо $d(\text{O}(\text{St}(z)), \text{St}(z)) < v$, де v порогова величина, яка залежить від предметної області та ІА. Враховуючи вигляд мережі переходів станів, а саме кон'юнкції дій, можемо стверджувати, що стан $\text{St}(j)$ досягнений, якщо $\text{St}(j) = F_{j_1} \wedge F_{j_2} \wedge \dots \wedge F_{j_{m_i}}$.

Затрати ресурсів для досягнення стану $\text{St}(j)$ становлять $R(j) = \sum_{i=1}^m R_{ijk}$, якщо була використана альтернатива a_k .

№	Випадок	Мережа Петрі	Граф
1.	Один вихід – один вхід		
2.	Один вихід – більше одного входу (для прикладу 2 входи)		
3.	Більше одного виходу (для прикладу 2 виходи)		

Позначимо g - функцію для визначення часових параметрів переходу зі стану в стан. Час для переходу зі стану i в стан j запишемо як $t_{ij} = g(R_{ijk}, R'_{ij})$, де R'_{ij} додаткові ресурси, які необхідні затратити для зменшення часу переходу.

Якщо, завдяки онтології ПО та літературним джерелам, ми оцінили затрати ресурсів R_{ij} для переходу із стану i в стан j , то мережу Петрі можна трансформувати у граф і задачу вибору шляху, розглядати як задачу асинхронного динамічного програмування [13]. Для

такої трансформації розглянемо три можливих випадки переходів у мережі Петрі. Тобто, якщо є більше, ніж один вихід, то дуги, що збираються у вершині з'єднуємо, щоб вказати, що ми маємо справу з кон'юнкцією. Використовуючи методи, придатні для розв'язування задач асинхронного динамічного програмування, знаходимо розв'язок у вигляді шляху переходу з початкового у кінцевий стан з мінімальними затратами [13].

4. Висновки

Запропонований метод використання адаптивних онтологій в інтелектуальних системах, заснованих на прецедентах та плануванні, дозволяє видавати ефективніші розв'язки, ніж традиційними методами, які використовуються інтелектуальними системами. Підвищення ефективності досягається шляхом введення в модель онтології коефіцієнтів важливості понять та зв'язків, які безпосередньо використовуються для обчислення відстаней між прецедентами та біжучою ситуацією, а також для оцінки станів ПрО. Для правильного функціонування інтелектуальної системи коефіцієнти важливості понять та зв'язків модифікуються на основі досвіду набутого системою, використовуючи методи інтелектуального аналізу даних. Показано, що визначена таким чином відстань задовольняє трьом аксіомам метрики.

Ця стаття теоретично обґрунтовує запропонований метод. У наступних роботах нами буде розглянуто практична доцільність використання такого методу з відповідними аплікаціями у предметній області фізико-механічних властивостей матеріалів. На цей момент онтологія цієї ПрО нами розробляється.

Література

1. Каменнова М.С. Каменнова. "Корпоративные информационные системы: технологии и решения". Системы Управления Базами Данных № 3/1995 стр. 88-99.
2. Гаврилова Т. А. Базы знаний интеллектуальных систем / Т.А.Гаврилова, В.Ф.Хорошевский. –СПб: Питер, 2001. – 384с.
3. Рассел С., Норвиг П. Искусственный интеллект. - М.,С.-П.,К.: Вильямс, 2006. - 1408с.
4. Логика рассуждений и ее моделирование / Под ред. Д.А.Поспелова. – М.: Науч. совет по комплекс. пробл. "Кибернетика" АН СССР, 1983. – 180 с.
5. Питерсон Дж. Теория сетей Петри и моделирование систем. – М.: Мир, 1984. – 264с.
6. Цветков 93 Цветков А. М. "Разработка алгоритмов индуктивного вывода с использованием деревьев решений". Кибернетика и системный анализ. - 1993. - № 1. - С. 174-178.
7. Аверкин А. Н., Батыршин И. З., Блишун А. Ф., Силов В. Б., Тарасов В. Б. Нечеткие множества в моделях управления и искусственного интеллекта // Под ред. Д. А. Поспелова. М.: Наука, 1986.
8. Даревич Р. Р., Досин Д. Г., В. В. Литвин. Метод автоматического назначения информационной ваги понятий в онтологий бази знань // Відбір та обробка інформації. – 2005. – Вип. 22(98). – С. 105-111.

9. Литвин В.В., Голощук Р.О. Інтелектуальні системи підтримки прийняття рішень на основі адаптивних онтологій. – тези VI міжнародної конференції „Математичне та програмне забезпечення інтелектуальних систем. – Дніпропетровськ. – 12-14 листопада 2008р. – С.208-209.
10. Свами М., Тхуласираман К. Графы, сети и алгоритмы. - М., 1984.
11. Месарович М., Мако Д., Такахага И. Теория иерархических многоуровневых систем. – М.: Мир, 1973. – 344 с.
12. Даревич Р.Р., Досин Д.Г., Литвин В.В., Назарчук З.Т. Оцінка подібності текстових документів на основі визначення інформаційної ваги елементів бази знань // Штучний інтелект. - №3. – 2006. – С.500-509.
13. Bertsekas, D. P. (2000). Dynamic Programming and Optimal Control, Vols. 1 & 2, 2nd ed. Athena Scientific.

Розглянута проблема інтеграції даних та знань, інтеграція інформаційних систем, що побудовані на основі онтологічного підходу. Розроблена архітектура об'єднання декількох інформаційних систем на основі застосування принципів мультиагентних систем та менеджменту знань

Ключові слова: архітектура інтеграції, інтелектуальний агент, інтеграція даних та знань

Рассмотрена проблема интеграции данных и знаний, интеграция информационных систем, спроектированных на основе онтологического подхода. Разработана архитектура объединения нескольких информационных систем на основании применения принципов проектирования мультиагентных систем и менеджмента знаний

Ключевые слова: архитектура интеграции, интеллектуальный агент, интеграция данных и знаний

In this paper considered the problem knowledge and data integration, integration ontology-based information systems. Proposed the integration architecture based on principles multi-agent system and knowledge management

Key words: integration architecture, intelligent agent, knowledge and data integration

УДК 519.7:004.8

РАЗРАБОТКА АРХИТЕКТУРЫ ИНТЕГРАЦИИ НЕСКОЛЬКИХ ИНФОРМАЦИОННЫХ СИСТЕМ

А.А. Воскобойникова

Аспирант
Харьковский национальный университет
радиоэлектроники
пр. Ленина 14, г. Харьков, 61166
Контактный тел.: 8 (057) 702-13-37
e-mail: voskobjnikova@gmail.com

1. Введение

На сегодня основным достижением развития IT-индустрии в рамках обеспечения и поддержки любой информации и знаний на корпоративном уровне стало внедрение систем управления документооборотом и систем управления знаниями. Крупные предприятия начинают повсеместно использовать достижения и технологии менеджмента знаний для повышения эффективности своей работы. Поэтому, активно разрабатываются и внедряются в жизнь интеллектуальные информационные системы, которые аккумулируют в себе все знания предприятия, представленные в виде онтологии.

На фоне того, что информационные технологии очень быстро развиваются и внедряются во все сферы

жизни человека, нельзя упускать из виду того, что они очень разнообразны. Каждое предприятие применяет те программные продукты и архитектуры построения систем, которые считает наиболее оптимальными, опираясь при этом на различного рода критерии: быстродействие, безопасность, стоимость решений и прочее. Поэтому, представление и описание одной и той же информации и знаний в двух разных информационных системах может существенно отличаться.

Эта проблема становится существенной, когда происходит объединение нескольких предприятий, включение одного в состав другого, объединение различных организаций в сообщества и т.д. При этом каждое предприятие (организация) предоставляет другому всю или часть имеющихся в нем информации и знаний, и взаимодействие между ними происходит