

КОМП'ЮТЕРНІ ЗАСОБИ, МЕРЕЖІ ТА СИСТЕМИ

V.Yu. Velychko

AN ALGORITHM OF GROWING PYRAMIDAL NETWORKS BUILDING IN PARALLEL COMPUTING ENVIRONMENT

Models and methods of the logical-linguistic analysis of complex objects are considered. Models are constructed on the basis of growing pyramidal networks (GPN) and expand opportunities of the latter due to improvement of GPN construction procedures.

Key words: growing pyramidal networks, complex objects.

Рассмотрены модели и методы логико-лингвистического анализа сложных объектов, в которых расширяются возможности растущих пирамидальных сетей за счет использования параллельного алгоритма построения сети.

Ключевые слова: растущие пирамидальные сети, сложные объекты.

Розглянуто моделі та методи логико-лінгвістичного аналізу складних об'єктів, в яких розширюються можливості зростаючих пірамідальних мереж за рахунок використання паралельного алгоритму побудови мережі.

Ключові слова: зростаючі пірамідальні мережі, складні об'єкти.

© В.Ю. Величко, 2011

УДК 681.3

В.Ю. ВЕЛИЧКО

АЛГОРИТМ ПОБУДОВИ ЗРОСТАЮЧИХ ПІРАМІДАЛЬНИХ МЕРЕЖ У ПАРАЛЕЛЬНОМУ ОБЧИСЛЮВАЛЬНОМУ СЕРЕДОВИЩІ

Вступ. Найбільш прийнятним типом моделей, що використовуються в інформаційно-аналітичних системах для обробки складних структур різнотипних даних і знань є логико-лінгвістичні моделі, тобто такі моделі, у яких основними елементами є не числа і обчислювальні операції, а імена та логічні зв'язки. Характерним прикладом логико-лінгвістичних моделей є поняття, які відображають закономірності, притаманні класам об'єктів. Під поняттям будемо розуміти узагальнений логіковий опис ознакової моделі класу об'єктів. Однією з добре апробованих можливих реалізацій логико-лінгвістичних моделей є організація пам'яті інформаційно-аналітичних систем у вигляді зростаючих пірамідальних мереж (ЗПМ) [1].

Відмінною рисою ЗПМ є структурування інформації одночасно з її введенням. Зміни у структурі пам'яті відбуваються при взаємодії збереженої інформації з новими даними. В результаті здійснення процесів структурування інформації встановлюються асоціативні зв'язки між об'єктами, які відображаються і закріплюються у структурних змінах пам'яті. На основі сформованої інформаційної моделі множини об'єктів здійснюються операції виділення понять, які надалі використовуються для вирішення задач класифікації, діагностики та прогнозування властивостей нових об'єктів, шляхом порівняння їх ознакових описів з поняттями виділеними у мережі. При індуктивному формуванні деякого поняття на основі атрибутивних мо-

делей метою аналізу є пошук сукупності кон'юнкцій значень атрибутів, що визначають всі об'єкти цього поняття та не визначають жодного об'єкта інших понять.

Побудова зростаючих пірамідальних мереж. У роботі [2] наведено модифіковані правила побудови ЗПМ, використання яких дозволяє позбутись негативного впливу порядку перегляду об'єктів з вибірки для навчання на результат побудови мережі. При обробці великих обсягів вхідних даних актуальним є питання розробки алгоритму паралельної побудови ЗПМ. Розглянемо більш детально такий алгоритм. Наведемо визначення основних термінів [3], які будуть використовуватись в описі алгоритму. Пірамідальною мережею Q називається ациклічний орієнтований граф $Q=(U,E)$ (U – множина вершин, E – множина дуг), в якому відсутні вершини, які мають одну дугу, що заходить. Вершини, які не мають дуг, що заходять, називаються *рецепторами*, інші вершини – *концепторами*. Рецептори відповідають окремим значенням ознак з описів об'єктів. Концептори відповідають комбінаціям значень ознак, що ідентифікують об'єкт у цілому, або відповідним спільним частинам описів декількох об'єктів. Множина вершин пірамідальної мережі – це множина $U=R_Q \cup C_Q$, де R_Q – множина рецепторів пірамідальної мережі, C_Q – множина концепторів пірамідальної мережі. Підграф пірамідальної мережі, що включає вершину a та всі вершини, з яких існує шлях до a , називається пірамідой вершини a . Вершини, що входять до піраміди вершини a утворюють її субмножину. Множина вершин, до яких існують шляхи з вершини a , називається її супермножиною. Множину вершин з субмножини вершини a , що безпосередньо зв'язані дугами з вершиною a , будемо називати $\mathbf{0}$ -субмножиною a та позначати F_a .

На початковій стадії у мережі існують лише рецептори з множини R_Q , кожен з яких відповідає значенню ознаки та концептори з множини C_A , які відповідають відомим об'єктам. Останні складають множину $C_A=A= \{a_1, a_2, \dots, a_i, \dots, a_n\}$. $C_{A'}$ – деяка підмножина множини A відомих об'єктів. Для кожного з об'єктів a_i відома відповідна множина рецепторів R_{a_i} . Побудова мережі починається з виконання правила **A1** [2].

Правило A1. Для кожного нового об'єкта a_i , який додається до мережі, та об'єкта $a_j, j=1, i-1$, який вже існує у мережі, визначається множина рецепторів $R_{c_k^l} = R_{a_j} \cap R_{a_i}$, яка відповідає деякому концептору c_k^l . Якщо концептор c_k^l , який відповідає перетину множин рецепторів об'єктів, існує у мережі, то він додається до $\mathbf{0}$ -субмножини об'єкта a_i . В іншому випадку в мережу вводиться новий концептор c_k^l , який з'єднується дугами, що виходять з

вершин множини $R_{c_k}^l$ та дугами, що заходять до вершин a_i та a_j . Дуги від рецепторів з $\mathbf{0}$ -субмножини нового концептора, які безпосередньо йшли до вершин a_i та a_j розриваються. Для кожного концептора c_k^l визначається його рівень l у мережі, який дорівнює кількості рецепторів з його $\mathbf{0}$ -субмножини $l = \text{Card}(R_{c_k}^l)$.

На рис. 1 показано застосування правила A1 при побудові ЗПМ. Початковий стан ЗПМ показано на рис. 1, а. Після додавання до мережі нового об'єкта a_i в мережі з'явиться новий концептор з номером 11 (рис. 1, б).

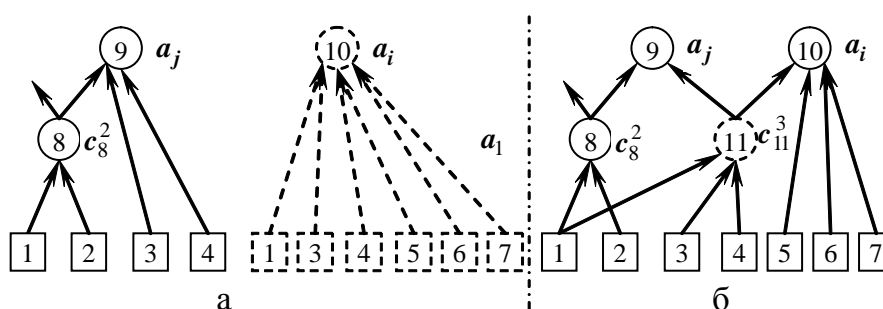


РИС. 1. Правило A1 побудови пірамідальної мережі

На другому етапі структуривання мережі виконується перевірка пірамід концепторів з множини $C_Q \setminus C_A$ на вкладеність за множиною рецепторів [4]. При обробці кожного концептора c_k^l починаючи з рівня $l = (|R_a| - 2)$ до рівня $l = 2$ його рецептори $R_{c_k}^l$ переводяться в стан збудження – $R_{c_k}^+$. Збудження розповсюджується по мережі та для кожного частково збудженого концептора $\forall c_m^n | l < n < (|R_a| - 1), 0 < m < |C_Q|$, який має шляхи до збуджених рецепторів, підраховується ступінь збудження за формулою $|R_{c_m}^+|$. Концептори, які входять до супермножини концептора c_k^l не розглядаються. Якщо ступінь збудження частково збудженого концептора c_m^n дорівнює l , концептор c_k^l додається до $\mathbf{0}$ -субмножини концептора c_m^n . На цьому етапі використання асоціативних властивостей ЗПМ дозволяє зменшити кількість операцій порівнянь концепторів.

Представимо вищенаведений алгоритм у вигляді блок-схеми з урахуванням паралельного виконання процедур порівняння об'єктів у зростаючій пірамідальній мережі (рис. 2).

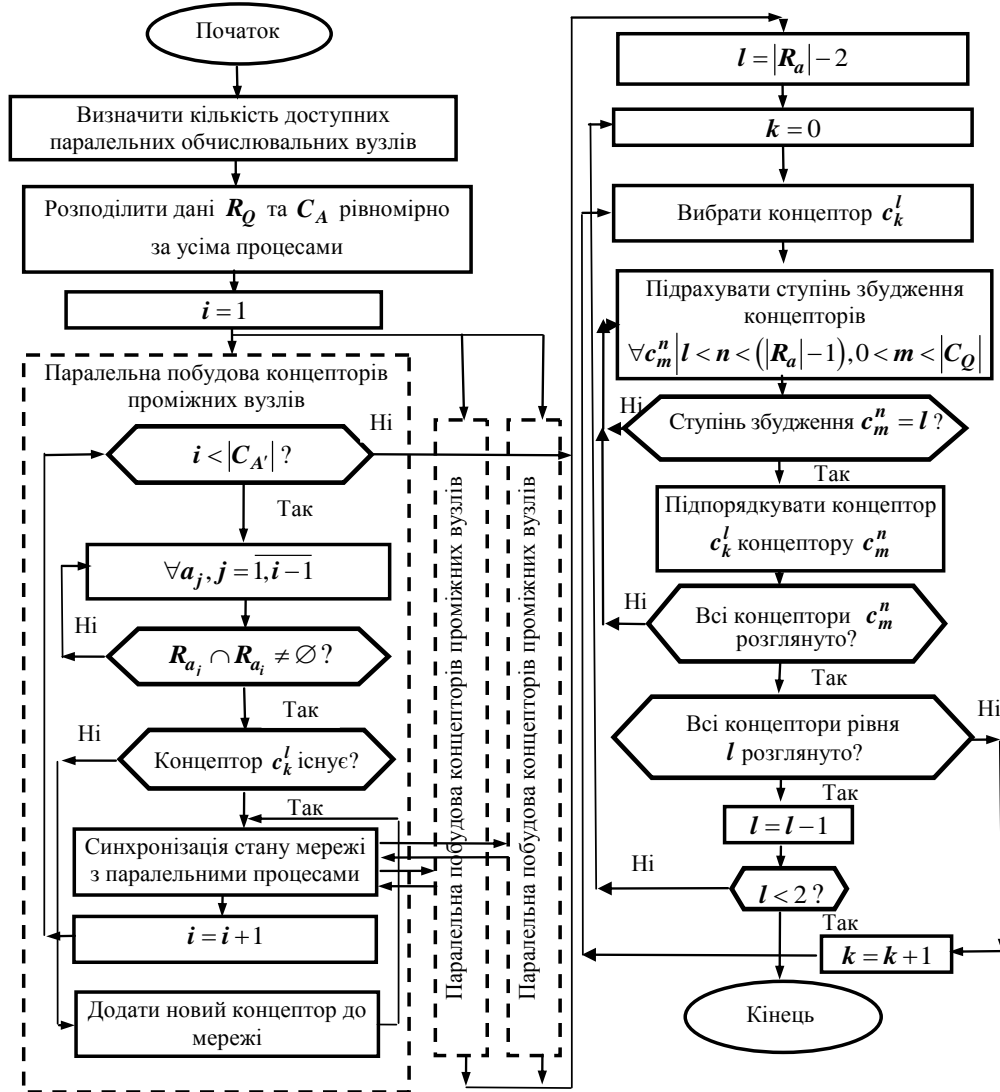


РИС. 2. Блок-схема алгоритму побудови ЗПМ

Оцінимо максимальну обчислювальну складність алгоритму побудови ЗПМ. Відповідно до правила **A1** загальна кількість порівнянь об'єктів розраховується за формулою $O_1 = |A| \lfloor (|A| + 1) / 2 \rfloor$. При виконанні порівнянь до мережі може бути додано не більше O_1 концепторів. Таке можливо, якщо $R_{a_j} \cap R_{a_i} \neq \emptyset$ для $\forall a_j, j = \overline{1, i-1}$ та множини рецепторів жодного з нових концепторів c_k^l не збігаються. Максимальна кількість операцій порівнянь на другому етапі

структурування мережі розраховується за формулою $O_2 = O_1 \lfloor (O_1 + 1) / 2 \rfloor$. Загальна кількість операцій порівнянь у мережі дорівнює $O = O_1 + O_2 = |A| \lfloor (|A| + 1) / 2 \rfloor + \left(\left(|A| \lfloor (|A| + 1) / 2 \rfloor \right) \lfloor (|A| \lfloor (|A| + 1) / 2 \rfloor + 1) / 2 \rfloor \right) / 2 \approx$

$\approx |A|^2 + |A|^3 / 4 + |A|^4 / 8$. Тобто обчислювальна складність алгоритму зростає пропорційно четвертого степеня загальної кількості об'єктів мережі. У реальних задачах обчислювальна складність буде меншою за рахунок використання асоціативних властивостей ЗПМ та меншої кількості концепторів, які будуються за правилом **A1**.

Пірамідальні мережі зручні для виконання різних операцій асоціативного пошуку [5, 6]. Наприклад, можна вибрати всі об'єкти, що включають задане поєднання значень ознак, простежуючи шляхи, що виходять з вершини мережі, яка відповідає цьому поєднанню. Для вибірки всіх об'єктів, описи яких перетинаються з описом заданого об'єкта, досить прослідкувати шляхи, що виходять з вершин піраміди цього об'єкта. Алгоритм побудови мережі забезпечує автоматичне встановлення асоціативної близькості між об'єктами за спільними елементами їх описів. Перехід від конвергованих представлень об'єктів (концепторів) до розгорнутих (наборів рецепторів) здійснюється шляхом перегляду пірамід у різних напрямках (зверху вниз і у зворотному напрямку).

Важливою властивістю пірамідальних мереж є їх ієрархічність, що дозволяє природним чином відображати структуру складених об'єктів і зв'язки типів рідвид та об'єкт-властивість. Концептори мережі відповідають поєднанням значень ознак, що визначають окремі об'єкти і кон'юнктивні класи об'єктів. При включенні збуджених вершин у піраміду об'єкта здійснюється прив'язка об'єкта до класів, визначення яких представлено цими вершинами. Таким чином, при побудові мережі формуються кон'юнктивні класи об'єктів, тобто здійснюється класифікація без вчителя.

Навчання ЗПМ полягає у формуванні в них структур, що представляють поняття [5]. Після побудови пірамідальної мережі утворюється структура, яка є представленням описів об'єктів вибірки для навчання. Кожна вершина мережі визначає кон'юнктивний клас об'єктів. Таким чином, об'єкт належить до всіх класів об'єктів, які визначаються вершинами, що входять до його піраміди, і також кожний об'єкт, піраміда якого містить в собі деяку вершину, належить класу об'єктів, який вона представляє. На основі аналізу мережі можна представити поняття у формі логікового виразу з використанням логікових операцій диз'юнкції, кон'юнкції та заперечення [5].

Подання поняття у вигляді логікового виразу є наочним, добре інтерпретується й може бути використано фахівцем з метою більш глибокого розуміння закономірностей, які притаманні предметній області.

Алгоритм побудови поняття у вигляді логікового виразу. Блок-схема алгоритму побудови логікових виразів показана на рис. 3. Формування логікових виразів виконується послідовно для кожного класу, об'єкти якого представлені в вибірці для навчання. Сформовані логікові вирази записуються в текстовий файл формату rtf.

Після настроювання на черговий клас аналізуються всі контрольні вузли [1] даного класу. Контрольні вузли поточного класу будемо називати «позитивними» контрольними вузлами, всі контрольні вузли інших класів вважаються «негативними» щодо поточного класу. Побудова логікового виразу починається з упорядкування позитивних контрольних вузлів у порядку убутання m – кількості об'єктів класу до пірамід яких входить контрольний вузол. Кожен «позитивний» контрольний вузол є основою для формування відповідного диз'юнктивного члена логікового виразу, що представляє поняття поточного класу.

Формування кожного диз'юнктивного члена починається із нерозглянутого контрольного вузла з найбільшим m . Насамперед у текстовий файл записується число m , що відповідає числу об'єктів вибірки для навчання, які належать до поточного класу й містять у своєму описі рецептори з субмножини обраного контрольного вузла. Потім у текстовий файл вписуються рецептори з піраміди обраного вузла, для чого вона проглядається в напрямку «зверху вниз». У текстовому файлі рецептори зв'язуються знаком кон'юнкції (&). Таку кон'юнкцію будемо називати базовою.

Далі здійснюється формування так званих кон'юнкцій-виключень, для чого в супермножині позитивного контрольного вузла здійснюється пошук найближчих негативних контрольних вузлів, тобто таких, на шляху до яких від позитивного контрольного вузла не зустрічаються інші контрольні вузли. Як негативні контрольні вузли розглядаються вузли, що належать будь-якому іншому класу, крім поточного. Якщо таких у мережі немає, у текстовий файл записується знак диз'юнкції (V) і починається формування чергового диз'юнктивного члена. Якщо в супермножині позитивного контрольного вузла є негативні контрольні вузли, триває формування кон'юнкцій-виключення шляхом вписування рецепторів, що входять у піраміду негативного контрольного вузла, але без урахування рецепторів, які належать до субмножини позитивного контрольного вузла. Рецептори, які вписуються, поєднуються знаком кон'юнкції (&), беруться в дужки й приєднуються до раніше сформованої частини логікового виразу через знаки кон'юнкції (&) і заперечення (¬). Аналіз негативних контрольних вузлів і формування відповідних кон'юнкцій-виключень також здійснюється в порядку убутання m , яке характеризує клас, що відповідає негативному контрольному вузлу. Формування диз'юнктивного члена закінчується після аналізу всіх контрольних вузлів, які є негативними щодо даного позитивного контрольного вузла.

Після того, як всі позитивні контрольні вузли поточного класу проаналізовано вищенаведеним чином, здійснюється перехід до формування логікового виразу для наступного класу.

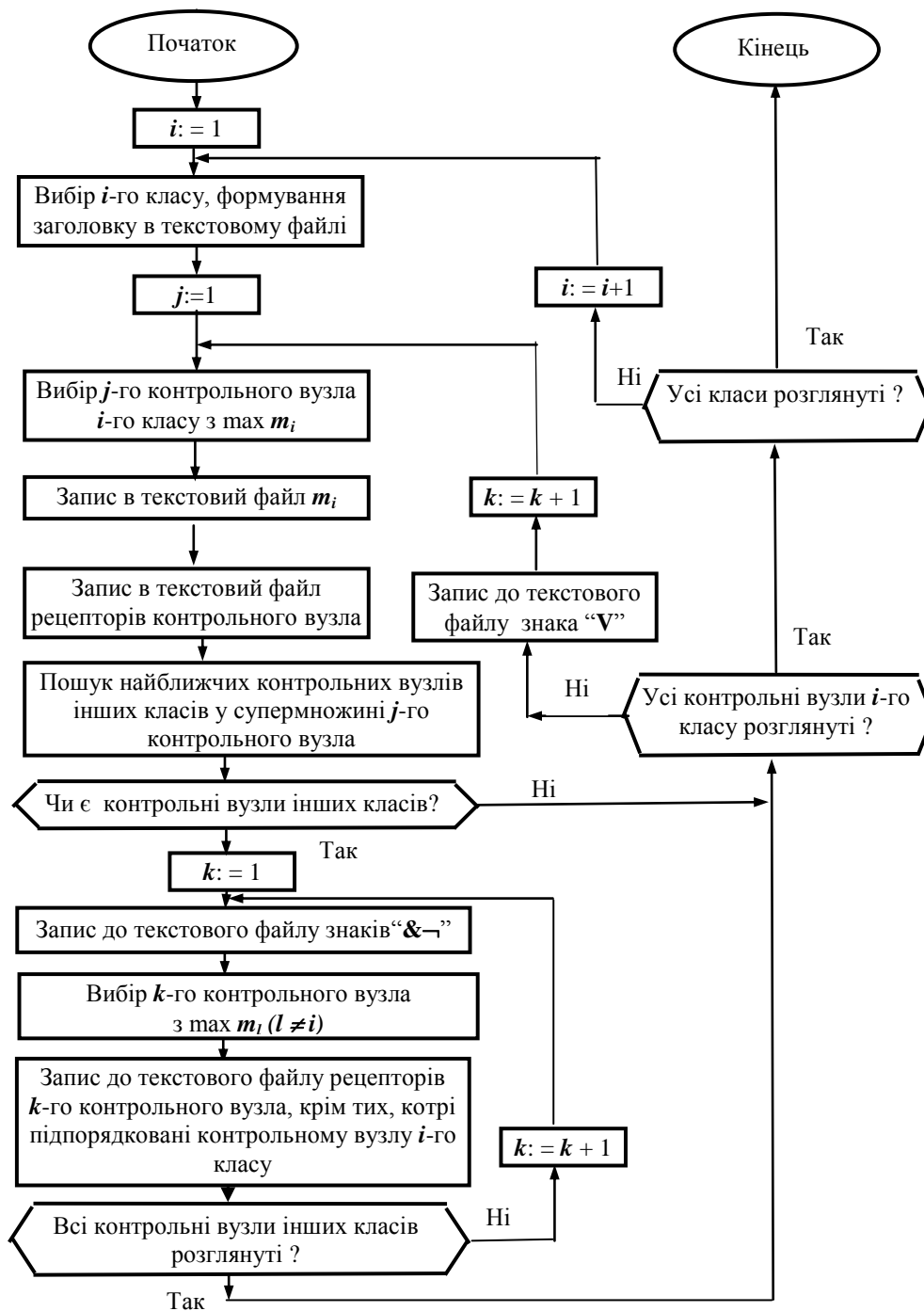


РИС. 3. Блок-схема алгоритму побудови логікових виразів

Базова кон'юнкція кожного диз'юнктивного члена відповідає сполученню значень ознак, характерних для об'єктів поточного класу, кон'юнкція-виключення відповідає сполученню значень ознак, яким об'єкти інших класів відрізняються від схожих на них об'єктів поточного класу.

Важливою особливістю методу формування понять у пірамідальних мережах є можливість введення в поняття кон'юнкцій-виключень, які не належать об'єктам досліджуваного класу. В результаті сформовані поняття мають більш компактну логічну структуру, що в принципі дає можливість підвищувати точність прогнозування.

Висновки. У роботі розглянуто методи підвищення ефективності логіко-лінгвістичного аналізу складних об'єктів, які розширюють діапазон застосування зростаючих пірамідальних мереж за рахунок використання паралельного алгоритму побудови ЗПМ. Подальші дослідження орієнтовані на проведення експериментів, вдосконалення паралельних алгоритмів та процедур побудови ЗПМ.

1. *Гладун В.П.* Планирование решений. – Киев: Наукова думка, 1987. – 168 с.
2. *Величко В.Ю., Палагін О.В.* Методи підвищення ефективності застосування зростаючих пірамідальних мереж // Комп'ютерні засоби, мережі та системи. – 2010. – № 9. – С. 106–114.
3. *Victor Gladun, Vitaly Velichko, Yuri Ivaskiv.* Selfstructured Systems // International J. "Information Theories & Applications". FOI ITHEA, Sofia. – 2008. – **15**. – N 1. – P. 5 – 13.
4. *Москалькова Н.М.* Методи правдоподібного виведення на основі представлення атрибутивних моделей знань в семантичних мережах // Автореф. дис. ... канд. техн. наук. – К.: Інститут кібернетики імені В.М. Глушкова НАН України, 2000. – 19 с.
5. *Гладун В.П.* Процессы формирования новых знаний. – София: СД «Педагог б», 1994. – 189 с.
6. *Гладун В.П.* Партнерство с компьютером. Человеко-машинные целеустремленные системы. – Киев: Port-Royal, 2000. – 118 с.

Отримано 20.10.2011