

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ТАРАСА ШЕВЧЕНКА

МАЖАРА ОЛЬГА ОЛЕКСАНДРІВНА



УДК 004.832

**ПРОГРАМНИЙ ІНСТРУМЕНТАРІЙ РОЗРОБКИ ПРОДУКЦІЙНИХ
СИСТЕМ З УРАХУВАННЯМ ОБМЕЖЕНИХ РЕСУРСІВ**

01.05.03 – математичне та програмне забезпечення
обчислювальних машин і систем

Автореферат дисертації на здобуття наукового ступеня
кандидата технічних наук

Київ – 2016

Дисертацією є рукопис.

Робота виконана на кафедрі автоматизації проектування енергетичних процесів і систем Національного технічного університету України «Київський політехнічний інститут» Міністерства освіти і науки України

Науковий керівник – кандидат технічних наук, доцент
Шаповалова Світлана Ігорівна,
Національний технічний університет України
«Київський політехнічний інститут» МОН України,
доцент кафедри автоматизації проектування
енергетичних процесів і систем

Офіційні опоненти: доктор технічних наук
Кургаєв Олександр Пилипович,
Інститут кібернетики ім. В. М. Глушкова НАН України,
м. Київ,
провідний науковий співробітник
відділу мікропроцесорної техніки

кандидат технічних наук
Поліновський В'ячеслав Васильович,
Інститут комп'ютерних технологій університету
«Україна», м. Київ, директор

Захист відбудеться 9 червня 2016 р. о 14:00 годині на засіданні спеціалізованої вченої ради Д 26.001.09 Київського національного університету імені Тараса Шевченка за адресою: 03680, м. Київ, пр. Академіка Глушкова, 4д, факультет кібернетики, ауд. 40.

З дисертацією можна ознайомитися в науковій бібліотеці імені М. Максимовича Київського національного університету імені Тараса Шевченка за адресою: 01601 МСП, м. Київ, вул. Володимирська, 58.

Автореферат розіслано « 27 » квітня 2016 р.

Вчений секретар
спеціалізованої вченої ради



В. П. Шевченко

ЗАГАЛЬНА ХАРАКТЕРИСТИКА РОБОТИ

Актуальність теми. Подальша інтелектуалізація сучасного програмного забезпечення визначає широке використання систем, які базуються на знаннях. Продукційні системи (ПС) є одним з найрозповсюдженіших різновидів таких систем. Сучасною тенденцією в розробці ПС є забезпечення їх застосування на портативних пристроях, зокрема для експрес-діагностування за результатами «польових обстежень». Наприклад, для дослідження віддалених об'єктів на сільськогосподарських угіддях, територіях заповідників та заказників, при проведенні розвідувальних робіт на важкодоступній місцевості або закритих територіях з підвищеними вимогами до інформаційної безпеки. На сьогодні масове застосування ПС на портативних пристроях часто обмежується вимогами щодо використання ресурсів оперативної пам'яті. Інший критерій ефективності – швидкодія – не є критичним для автономних продукційних систем. В той же час, він є визначальним для гібридних програмних комплексів, в яких інтелектуальна задача є підзадачею та вирішується окремою підсистемою з власним механізмом обробки інформації. Проте зараз відсутній програмний інструментарій розробки прикладних продукційних систем, який дозволить мінімізувати затрати на логічне виведення заключення за критеріями швидкодії та/або затрат оперативної пам'яті.

Концепція продукційної моделі представлення знань була закладена E.Post, в подальшому розвинена A.Newell та H.Simon. Базовими механізмами логічного виведення в ПС є розв'язання конфлікту та співставлення зі зразком (СЗЗ). С.Forgy доведено, що саме співставлення займає до 90 % ресурсів при логічному виведенні. Для реалізації СЗЗ існує два базових підходи, запропоновані С.Forgy та D.Miranker. Оптимізовані методи на їх основі створили E. Hanson, I. Wright, P. Rosenbloom. Однак всі методи СЗЗ, які використовуються в програмному інструментарії розробки продукційних систем, є модифікаціями базових способів: Rete та Treat. Тому саме за цими способами D.Miranker, I.Wright, P.Nayak, Y.Wang, A. Gupta проводили дослідження з порівняння характеристик логічного виведення. Однак їх результати є суперечливими тому, що: по-перше, кожне дослідження проводилося лише для окремих випадків прикладних задач, по-друге – ефективність логічного виведення для однієї й тієї ж задачі залежить від способу представлення умов продукцій в базі знань (БЗ).

Наразі не розроблено узагальнених рекомендацій для вибору одного з цих підходів за критеріями швидкодії та ресурсів пам'яті. За умови відсутності теоретичних засад, ефективність логічного виведення поточної задачі можна було б визначити експериментально. Однак в усіх сучасних середовищах розробки продукційних систем реалізовано лише один з механізмів співставлення зі зразком. Таким чином, з одного боку, необхідно вдосконалити підходи до формального порівняння базових способів співставлення, незалежно від засобів реалізації, а з іншого – створити середовище логічного виведення з реалізацією двох підходів до СЗЗ для експериментального вибору оптимального для поточної прикладної задачі.

Таким чином, проблема створення теоретичних і практичних методів розробки продукційних систем з визначення оптимального для поточної задачі способу

співставлення зі зразком з урахуванням обмежень щодо швидкодії та/або ресурсоемності є актуальною та має практичне значення.

Зв'язок роботи з науковими програмами, планами, темами. Дисертаційна робота виконувалася у Національному технічному університеті України «Київський політехнічний інститут» згідно з планом науково-дослідних робіт кафедри автоматизації проектування енергетичних процесів та систем, а також в рамках науково-дослідних робіт «Автоматизація моніторингу геологічного середовища в зоні впливу споруд АЕС» (номер державної реєстрації 00113U01696) та «Автоматизація моніторингу гідрохімічного стану підземних вод АЕС» (номер державної реєстрації 0115U000329).

Мета і завдання дослідження. Створити теоретичні та алгоритмічні основи розробки продукційних систем, які потребують мінімальних ресурсів пам'яті та часу. Для досягнення поставленої мети визначено такі завдання:

- провести аналіз підходів до співставлення зі зразком в продукційних системах для визначення оптимального за ресурсоемністю;
- дослідити підходи до формалізації опису продукційних систем для представлення в єдиному форматі формального опису співставлення зі зразком за двома базовими способами: Rete та Treat;
- дослідити сучасні програмні засоби розробки продукційних систем та обрати інструментарій, який задовольняє вимогам застосування на портативних пристроях;
- визначити в термінах логіки першого порядку спосіб співставлення зі зразком за Treat алгоритмом;
- створити програмний інструментарій розробки прикладних продукційних систем, які потребують мінімальних ресурсів при розв'язанні поточної прикладної задачі;
- провести обчислювальні експерименти на тестових прикладах на запропонованому програмному інструментарії;
- розробити загальне рішення продукційної системи експрес-діагностування на портативних пристроях в умовах обмеженого доступу до мереж мобільного зв'язку та Internet;
- експериментально перевірити запропоновані рішення при розробці програмного забезпечення розв'язання прикладних задач діагностування.

Об'єктом дослідження є логічне виведення в продукційних системах.

Предметом дослідження є співставлення зі зразком в продукційних системах.

Методи дослідження. Розв'язання поставлених у роботі завдань виконувалось на базі теорії логічного виведення за продукційною моделлю, методів розрахунку витрат пам'яті при співставленні антецедентів продукцій, методів тестування оболонок продукційних систем, методів вимірювання характеристик ефективності логічного виведення оболонки продукційних систем CLIPS для проведення обчислювальних експериментів з доведення достовірності отриманих результатів.

Наукова новизна отриманих результатів полягає в наступному:

1. Вперше формалізовано співставлення зі зразком за Treat алгоритмом в термінах логіки першого порядку для порівняння Rete та Treat способів за критеріями ресурсоемності.

2. Вперше запропоновано програмний інструментарій розробки прикладних продукційних систем на основі двох базових способів співставлення зі зразком для мінімізації ресурсів при розв'язанні поточної прикладної задачі.

3. Удосконалено програмний інструментарій розробки продукційних систем на базі CLIPS за рахунок реалізації Treat алгоритму для вибору алгоритму співставлення зі зразком, який забезпечує мінімальне використання обчислювальних ресурсів.

4. Удосконалено програмне забезпечення розробки продукційних систем експрес-діагностування за рахунок інтеграції механізму логічного виведення з мінімальними вимогами щодо оперативної пам'яті для використання на портативних пристроях в умовах автономної експлуатації.

Практичне значення отриманих результатів полягає, по-перше, у створенні формалізації, яка дозволяє адаптувати існуючий математичний апарат Rete алгоритму для обчислення характеристик співставлення за Treat алгоритмом, по-друге, в розробці програмного середовища моделювання логічного виведення за базовими підходами до СЗЗ для підвищення ефективності прикладної продукційної системи.

Впровадження одержаних результатів виконані в: 1) ТОВ «Енерджи енд Дейта Інтелідженс Консалтінг Юкрейн» (акт про впровадження № 0104-95 від 19.09.2013 р., м. Київ) – для розробки інформаційної системи діагностики будівель підприємств енергетики; 2) ДП «Квазар-4» (акт про впровадження № 45–14-14 від 14.02.2014 р., м. Київ) для програмного забезпечення інформаційних систем підприємства; 3) агрохолдингу «Астарта-Київ» (акт про впровадження № 21-4/403 від 19.10.2015 р., м. Київ) для інформаційної системи фітосанітарного моніторингу рослин; 4) НТУУ «КПІ» (акти №1063-01-16 та №1064-01-16 від 13.01.2016 м. Київ) для виконання науково-дослідних робіт; 5) НТУУ «КПІ» (довідка про впровадження № 1097-01-16 від 15.01.2016 р., м. Київ) в навчальному процесі на кафедрі автоматизації проектування енергетичних процесів і систем при викладанні дисципліни «Методи та системи штучного інтелекту» підготовки бакалаврів напряму «Комп'ютерні науки».

Особистий внесок здобувача. Результати виконаних досліджень, які виносяться на захист і складають наукову новизну роботи, отримано автором самостійно та викладено у 21 праці. Роботи [4] та [6] підготовлені самостійно. В інших роботах, опублікованих в співавторстві, автору належать: визначення характеристик логічного виведення в продукційних системах [1], формалізація компіляції та правил переходу Treat мережі потоку даних [2, 17, 21], архітектурне рішення та реалізація Treat алгоритму для середовища CLIPS [19, 20], загальне рішення задачі діагностування для ПС на портативних пристроях [5, 9], представлення задач пошуку в оболонці CLIPS [8, 11, 16], формалізація прикладних задач для оболонки CLIPS [3, 7, 8, 10, 12–15].

Апробація результатів дисертації. Основні результати досліджень та окремі їх положення пройшли апробацію на 12 конференціях, з яких 2 – за кордоном: XII та XIII міжнародних конференціях «Інтелектуальний аналіз інформації» (Київ, 2012, 2013 рр.), II та III міжнародних науково-технічних конференціях «Обчислювальний інтелект (результати, проблеми, перспективи)» (Черкаси, 2013, 2015 рр.),

I Международной Интернет-конференции «Энергетика в современном мире» (Казань, Россия, 2013 г.), II Міжнародній науково-практичній конференції «Інформаційні технології в освіті, науці і техніці» (Черкаси, 2014 р.), VI науково-практичному семінарі з міжнародною участю «Економічна безпека держави і науково-технологічні аспекти її забезпечення» (м. Київ, 2014 р.), XIV міжнародній науково-технічній конференції «Системний аналіз та інформаційні технології» (м. Київ, 2014 р.), Sustainability Innovation Cross border Collaboration Fellowship PhD workshop GUC (Gjøvik, 15–17 October 2014), II науково-практичній конференції молодих вчених, аспірантів та студентів «Сучасні аспекти розробки програмного забезпечення» (Київ, 2015 р.), XV та XVI міжнародних науково-практичних конференціях «Сучасні інформаційні та електронні технології» (Одеса, 2014, 2015 рр.).

Публікації. Результати виконаних дисертаційних досліджень опубліковані в 21 праці, з яких 2 – у виданнях, що входять до наукометричних баз (Index Copernicus); 4 – у виданнях, затверджених МОН України, як фахові, 4 – у монографіях, 2 статті у збірниках наукових праць за матеріалами конференцій. Отримано авторське свідоцтво «Вирішення задачі пошуку в просторі станів у середовищі CLIPS» (а. с. № 43282, Україна).

Структура та обсяг роботи. Дисертація складається зі вступу, чотирьох розділів, висновку, списку використаних джерел із 106 найменувань та 4 додатки. Робота містить 112 сторінок основного тексту, 19 рисунків, 16 таблиць.

ОСНОВНИЙ ЗМІСТ РОБОТИ

У **вступі** розкрито стан наукової проблеми, її теоретичну та прикладну значущість; обґрунтовано актуальність теми, сформульовано мету та задачі роботи; відзначено її наукову новизну та практичне значення результатів; наведено дані про зв'язок дисертації з науковими програмами та темами, а також інформацію про апробацію результатів роботи та її публікацію у провідних фахових виданнях; відображено особистий внесок здобувача.

У **першому розділі** розглянуто концепцію ПС. Продукційна система – це набір продукційних правил, кожне з яких представляє послідовність дій, які повинні виконуватися за певних умов, та механізм для співставлення та виконання цих правил. У розділі проведено аналіз базових механізмів продукційних систем: співставлення зі зразком та розв'язання конфліктів. Визначено, що мінімізувати ресурсоємність логічного виведення в прикладних задачах можна за рахунок обрання оптимального алгоритму СЗЗ.

Сучасні способи СЗЗ реалізуються у вигляді інкрементних алгоритмів (рис. 1). Найбільш поширеними серед них є алгоритми нетерплячої оцінки, які передбачають формування повної конфліктної множини на кожному циклі логічного виведення. Проведено аналіз досліджень з порівняння співставлення за цими двома базовими способами нетерплячої оцінки. Визначено необхідність створення теоретичних засад та програмних засобів порівняння характеристик ресурсоємності для загального випадку, а також для поточних прикладних задач. Розглянуто способи формалізації опису продукційних систем. Обґрунтовано використання найбільш універсальної формалізації ПС S4 в термінах логіки першого порядку.

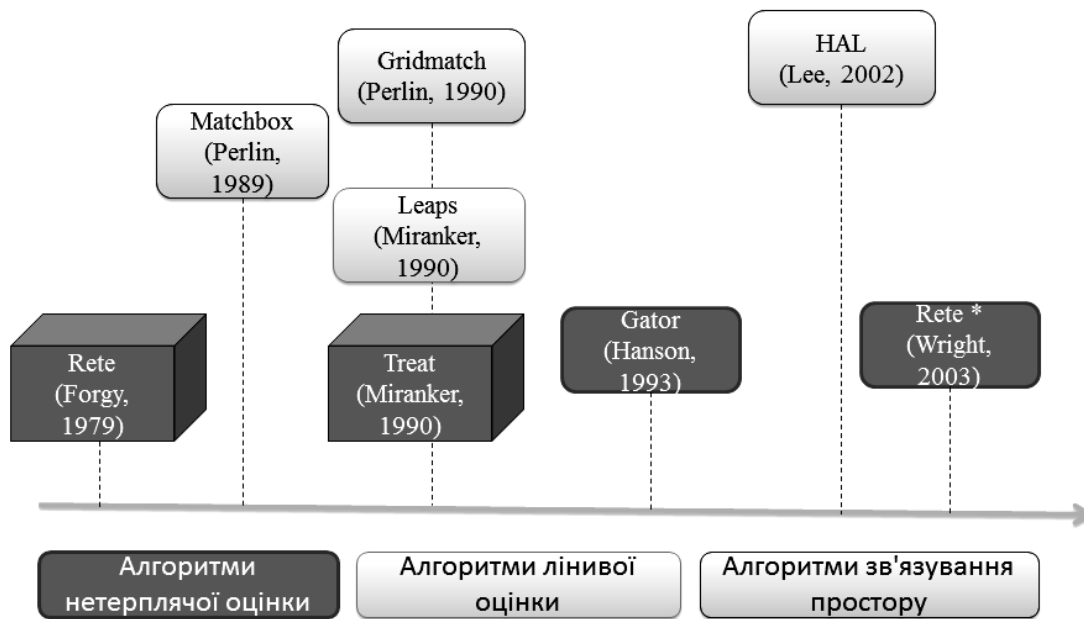


Рис. 1. Еволюція інкрементних алгоритмів співставлення

Формалізація S4 має розширення для Rete алгоритму СЗЗ. Однак наразі не представлено формалізацію другого базового способу СЗЗ – Treat алгоритму в загальному вигляді, який дозволив би їх формальне порівняння незалежно від мови та засобів реалізації. Визначено необхідність створення формального опису базових механізмів співставлення зі зразком в ПСв загальному вигляді.

Проведено дослідження програмного інструментарію розробки ПС, яке довело, що в усіх сучасних спеціалізованих середовищах реалізовано лише один з механізмів СЗЗ. Сформульовано необхідність створення програмного інструментарію розробки прикладних ПС на основі двох базових способів СЗЗ для мінімізації ресурсів при розв'язанні поточної прикладної задачі. Обґрунтовано використання програмної оболонки CLIPS.

В **другому розділі** представлено базові способи СЗЗ – Rete та Treat – в єдиному форматі і за описом процесу співставлення, і за його формалізацією. Rete та Treat відносяться до інкрементних алгоритмів, які ґрунтуються на запам'ятовуванні кортежів фактів, що узгоджуються з частиною антецеденту.

Співставлення відбувається у два етапи. На першому етапі – прекомпіляції – для кожної продукції з БЗ на основі умовної частини будується єдина мережа потоку даних. Компіляція виконується при запуску системи та повторюється після кожної модифікації БЗ.

Другий етап – виконання – здійснюється на кожному циклі логічного виведення за поставленою прикладною задачею. При цьому відбувається обробка мережі потоку даних на основі змін робочої пам'яті (РП) ПС, виконаних на попередньому циклі. Результатом цього етапу є конфліктна множина з продукцій БЗ та кортеж фактів, які призвели до їх активації. Таким чином, співставлення здійснюється не безпосередньо за представленням логічного виразу умовної частини продукції, а за заздалегідь створеною на етапі прекомпіляції мережею розбору антецеденту (рис. 2).

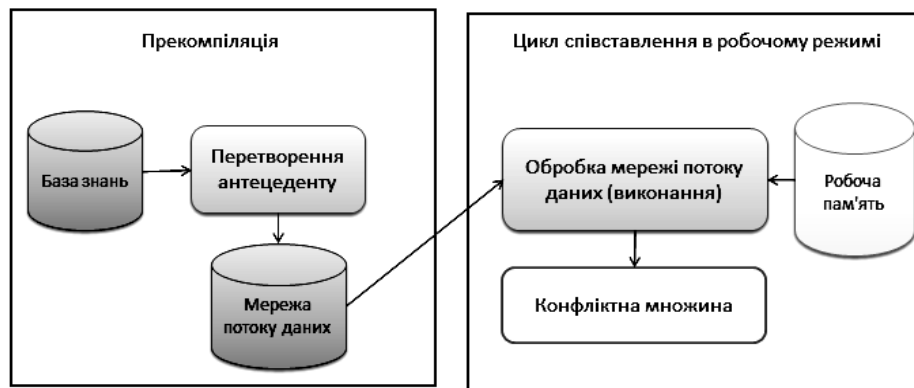


Рис. 2. Співставлення зі зразком за алгоритмами нетерплячої оцінки

Мережа потоку даних Rete алгоритму складається з двох частин, які відповідають етапам обробки антецеденту продукції: α - та β -мережі (рис. 3-а). Узгодження в рамках одного шаблону продукції називаються внутрішніми тестами. На етапі виконання у вузлах α - мережі зберігаються відомості про факти, які призвели до активації вузла. Множина цих фактів складає α -пам'ять. У вузлах β - мережі відбувається узгодження шаблонів в рамках окремої продукції – зовнішні тести.

В Treat алгоритмі α -мережа потоку даних використовується аналогічно до Rete, але β -мережа відсутня (рис. 3-б). Натомість узгодження змінних здійснюється в режимі логічного виведення за динамічно створеною термінальною мережею.

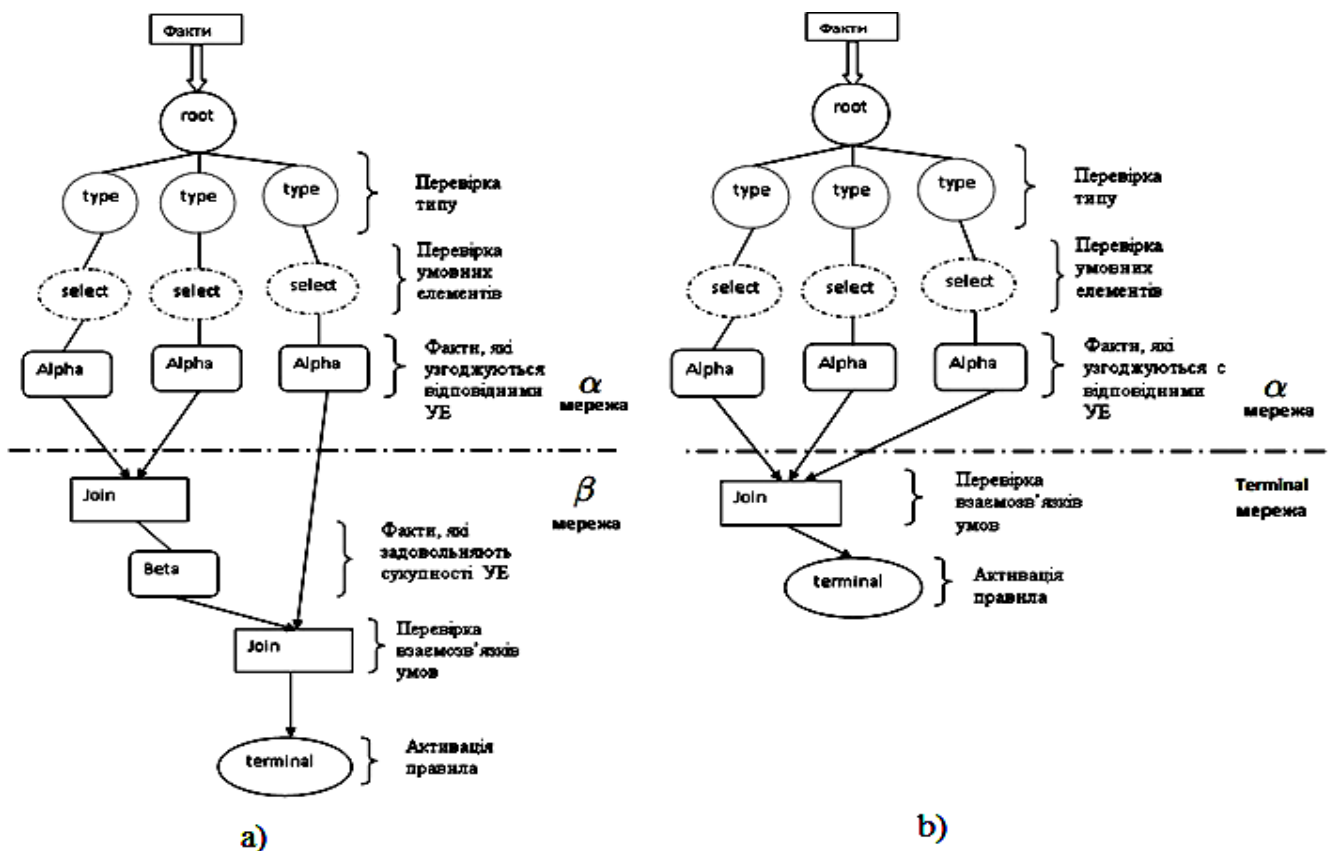


Рис. 3. Мережа потоку даних: а) – Rete алгоритму, б) – Treat алгоритму

В розділі представлено формалізації базових алгоритмів нетерплячої оцінки. Для цього наведено формалізацію S4 продукційної системи (табл. 1), визначення та позначення в термінах логіки першого порядку тих її складових, які використовуються для формалізації алгоритмів співставлення.

Таблиця 1

Формалізація продукційної моделі

Визначення продукційної систем	Визначення продукції	Вміст РП
$PS = (F, P, X, L, WM_0, R, S)$, F – множина функціональних символів, P – множина предикативних символів, X – множина змінних, L – множина міток, WM_0 – початковий стан РП, обмежений до термів висотою 1, R – множина продукційних правил, де правила мають шаблони висотою 1, S – стратегія розв’язання конфліктів.	$[l]p, c \rightarrow r, a$ l – ім’я з множини міток; $p = p^+ \cup p^-$ – множина позитивних p^+ та негативних p^- шаблонів, представлених у вигляді базових термів; c – пропозиція, множина вільних змінних якої є підмножиною змінних шаблону p ; r – множина фактів, які мають бути видалені з РП; a – множина фактів, які мають бути додані до РП	Множина фактів, представлених у вигляді базових термів $f \in T(F)$, де $T(F)$ – множина базових термів

Нехай дано: FS – множину фактів та P^+ – множину позитивних шаблонів. Шаблон P^+ узгоджується з фактами FS у відповідності до формальної теорії FT та підстановки σ , записується $P^+ \ll_{FT}^{\sigma} FS$, якщо

$$\forall p \in P^+ \exists t \in FS \mid \sigma(p) =_{FT} t \quad (1)$$

Негативний шаблон P^- не узгоджується з множиною фактів FS , записується $P^- \ll_{FT}^{\sigma} FS$, якщо і лише якщо

$$\forall -p \in P^- \forall t \in FS \forall \sigma \mid \sigma(p) \neq_{FT} t \quad (2)$$

Позиція w підтерма s в термі t визначається наступним чином:

$$\begin{aligned} w(t) \mid t &= 0, \\ w(s_i) \mid t(s_1, \dots, s_n) &= i, \\ w_1 = w(s) \mid t \wedge w_2 = w(t) \mid r &\Rightarrow w(s) \mid r = w_1.w_2, \end{aligned}$$

де n – кількість підтермів у термі, s_i – i -й підтерм, t, r – терми визначені на множині термів $T(X, P)$ – множина термів, P – множина предикативних символів, X – множина змінних.

В розділі наведено існуючу формалізацію Rete алгоритму. На її основі розроблено повну формалізацію Treat алгоритму.

Компіляція $comp$ продукції $[l]p, c \rightarrow r, a$ для α -мережі Treat алгоритму аналогічна до відповідного інкрементного визначення компіляції Rete:

$$comp(t) := comp(t, 0) \text{ для } t \in T(X, P), \quad (3)$$

$$\begin{aligned} \text{comp}(t(s_1, \dots, s_n)) &:= \text{input}.w = t, \\ &\quad \bigwedge_{i \in I} \text{comp}(s_i, w.i), \end{aligned} \quad (4)$$

$$\bigwedge_{\forall i < j \in I} \bigwedge_{\exists w_1, w_2 \mid s_i | w_1 = s_j | w_2} \text{input}.w.w_1 = \text{input}.w.w_2, \quad (5)$$

$$\text{comp}(\neg t(s_1, \dots, s_n)) := \neg(\exists \text{input} \in WM) \mid \text{comp}(t(s_1, \dots, s_n)), \quad (6)$$

де input – факт, який узгоджується з поточним шаблоном, відповідно input_i для шаблону p_i ; $i \in I$ – номер шаблону в правилі; n – кількість шаблонів в правилі.

Кон'юнкція всіх виразів вигляду (4) – (6) позначається як q_i^k для шаблону p_i правила l_k бази знань, $k \in K$ – ідентифікатор правила, K – множина ідентифікаторів правил в БЗ

Прекомпіляція зовнішніх тестів Treat на базі відповідного формального представлення β -мережі Rete алгоритму:

$$\text{comp}(p, c) := \bigwedge_{i \in I} \text{comp}(p_i), \quad (7)$$

$$\bigwedge_{\forall i < j \in I(p^+) \vee I(p^-)} \bigwedge_{\forall w_1 w_2 \mid p_i | w_1 = p_j | w_2 \in X} \text{input}_i.w_1 = \text{input}_j.w_2, \quad (8)$$

$$\bigwedge_{\forall i \in I(p^+) \wedge j \in I(p^-)} \bigwedge_{\forall w_1 w_2 \mid p_i | w_1 = p_j | w_2 \in X \wedge \neg(\exists f \in WM)} \text{input}_i.w_1 = f.w_2, \quad (9)$$

$$\bigwedge_{\forall i < j \in I} \bigwedge_{\exists w \mid p_i | w = x \in \text{Var}(c)} \text{input}_i.w = x =: \sigma_i \wedge \sigma_j(c), \quad (10)$$

c – пропозиція, множина вільних змінних якої є підмножиною змінних шаблону p , p^+ – позитивний шаблон з множини шаблонів, p^- – негативний шаблон; Var – множина змінних поточного антецеденту, $\text{Var} \in X$; σ – підстановка для даного шаблону, f – факт РП.

Кон'юнкцію всіх виразів вигляду (4) – (6) для продукції l_k бази знань R позначають як r^k .

Для подальшого коректного порівняння алгоритмів співставлення правила переходу представлено за форматом та синтаксисом, запропонованим для Rete алгоритму.

В загальному вигляді правила переходу для вершин Treat мережі мають наступний вигляд:

$$?m \xrightarrow{\text{act}} ms! \quad (11)$$

де $?m$ – вхідний маркер у вигляді $\langle + | -, f \rangle$ для факту f , що додається чи видаляється з РП; $ms!$ – список вихідних маркерів, act – опціональні дії на поточному кроці.

Для антецеденту правила π ліва пам'ять, що містить результати внутрішніх тестів, lm , визначаються таким чином:

$$lm_i = \{(f_1, \dots, f_{j_i}) \in WM \mid \text{comp}(\pi)(f_1, \dots, f_{j_i})\} \quad (12)$$

Термінальна пам'ять tm , яка використовується в процесі узгодження змінних та пам'ять конфліктної множини cm , яка зберігається на кожному кроці співставлення визначаються наступним чином:

$$tm_i = \{f \in WM \mid q_{i+1}(f)\} \quad (13)$$

$$cm^k = \{(f_1, \dots, f_{j_i}) \in WM \mid comp(l_k)\}. \quad (14)$$

Термінальна пам'ять формується в процесі узгодження змінних в межах антецеденту.

Для α -вершин правила переходу визначаються наступним чином:

$$\langle +, f \rangle ? \xrightarrow{lm_j \leftarrow lm_j \cup \{f\}} \{+, f\}! \text{ if } q(f) \quad (15)$$

$$\langle -, f \rangle ? \xrightarrow{lm_j \leftarrow lm_j - \{f\}} \{-, f\}! \text{ if } q(f) \quad (16)$$

$$\langle + \mid -, f \rangle ? \rightarrow 0! \text{ if } \neg q(f). \quad (17)$$

Надамо формальне представлення переходів в межах динамічної мережі узгодження при обробці термінальної вершини. Формалізуємо додавання факту до РП для різних випадків. Додавання факту, який узгоджується з позитивним шаблоном:

$$\langle +, f \rangle ? \xrightarrow{tm_j \leftarrow tm_j \cup \{f\} \wedge \sigma \leftarrow \sigma \cup \sigma \setminus \exists p_i^k \in p^+(l^k) \mid \sigma \setminus p_i^k = f} \{\langle +, f \rangle\}! \text{ if } r_k(f) \quad (18)$$

$$\langle +, f \rangle ? \xrightarrow{tm_j \leftarrow tm_j \cup \{f\} \wedge \sigma \leftarrow \sigma \cup \sigma \setminus \exists p_i^k \in p^+(l^k) \mid \sigma \setminus p_i^k = f} \{\langle +, (l^k, tm_j) \rangle\}! \text{ if } r_k(tm_j). \quad (19)$$

Додавання факту до термінальної вершини спричиняє її активацію у тому випадку, якщо виконалися умови узгодження змінних в межах факту та поточної термінальної пам'яті. Перехід до вершини конфліктного набору здійснюється, коли узгоджено всі змінні правила.

Додавання факту, який узгоджується з негативним шаблоном:

$$\langle +, f \rangle ? \xrightarrow{tm = \emptyset} \{\langle -, (l^k, f) \rangle\}! \text{ if } r_k(f) \wedge \exists p_i^k \in p^-(l^k) \mid \sigma \setminus p_i^k = f. \quad (20)$$

Видалення факту, який узгоджувався з позитивним шаблоном, призводить до деактивації відповідної продукції. При цьому з екземпляру конфліктного набору видаляються факти, які приводили до активації виключно цього правила. Цей процес потребує додаткової перевірки на узгодженість фактів поточного правила з іншими правилами конфліктної множини CS .

Видалення факту, який узгоджувався з позитивним шаблоном правила формалізується наступним чином:

$$\langle -, f \rangle ? \xrightarrow{tm = \emptyset} \{\langle -, (l^k, f) \rangle\}! \text{ if } (l^k \in CS) \wedge (\sigma p_i^k = f \mid p_i^k \in p^+). \quad (21)$$

Видалення факту, який призводив до узгодження з негативним шаблоном правила:

$$\langle -, f \rangle ? \xrightarrow{tm = \emptyset} \{\langle +, l^k, f \rangle\}! \quad (22)$$

$$\text{if } (yf \in lm_i^k \mid \sigma p = f \wedge r(l, f)) \wedge \sigma p_i^k = f \mid p_i^k \in p^-).$$

Тоді для конфліктної множини CS виконуються наступні зміни у відповідності до вхідних даних з термінальної вершини:

$$\langle -, (l^k, f) \rangle ? \xrightarrow{cm \leftarrow cm \cup \left(\begin{array}{c} cm^k - \bigcup \\ \forall j \neq k \mid cm^j \in cm \end{array} \right)} \{\langle -, l^k \rangle\}!, \quad (23)$$

$$\langle +, (l^k, tm^k) \rangle ? \xrightarrow{cm \leftarrow cm \cup tm^k} \{\langle +, (l^k, tm^k) \rangle\}!, \quad (24)$$

$$\langle +, l^k \rangle ? \rightarrow \{ \langle +, l^k \rangle \} ! \quad (25)$$

Таким чином, запропоновано формалізацію Treat алгоритму (5) – (23), в якій формули (5) – (10) та (15) – (17) аналогічні існуючим формулам Rete алгоритму, (18) – (25) – запропоновано вперше.

На основі представлених в єдиному форматі Rete та Treat способів СЗЗ запропоновано формальний опис порівняння Rete та Treat, обґрунтування існуючих методів оптимізації Rete алгоритму, наведено приклад розширення існуючої моделі розрахунку витрат пам'яті для відповідних алгоритмів.

В **третьому розділі** на основі оболонки ПС CLIPS (ver. 6.3) за рахунок реалізації додаткового способу СЗЗ запропоновано реалізацію середовища розробки прикладних ПС, оптимальних за ресурсами пам'яті та/або швидкістю.

Поставлено вимоги до реалізації Treat алгоритму: 1) збереження загальної архітектури, форматів представлення даних та існуючих методів оптимізації СЗЗ, 2) забезпечення компіляції ПС лише з одним з механізмів співставлення.

Розглянуто способи оптимізації Rete алгоритму та визначено відмінності між: класичною версією Rete алгоритму (С. Forgy); версією *Rete^{cl}*, реалізованою в середовищі CLIPS (G. Riley) та класичною реалізацією Treat алгоритму (D. Miranker) (табл. 2). Визначено, що в CLIPS використовується реалізація Rete алгоритму з хешуванням α - і β -пам'яті в мережі потоку даних та підходу до асиметричний підхід до видалення фактів.

Таблиця 2

Порівняння основних реалізацій співставлення

Способи СЗЗ	Rete (Forgy)	<i>Rete^{cl}</i> (Riley)	Treat (Miranker)
Компоненти реалізації			
Мережа потоку даних	α -мережа внутрішніх тестів	α -мережа внутрішніх тестів	α -мережа внутрішніх тестів
	β -мережа зовнішніх тестів	β -мережа зовнішніх тестів	Динамічна мережа узгодження
Збереження даних	α -пам'ять узгодження фактів та β -пам'ять часткового узгодження з шаблонами	α -пам'ять узгодження фактів та β -пам'ять часткового узгодження з шаблонами, конфліктна множина	α -пам'ять узгодження фактів, конфліктна множина
Обробка додавання фактів	Поширення від кореневого вузла в статичній мережі потоку даних	Поширення від кореневого вузла в статичній мережі потоку даних	Поширення по статичній α -мережі та побудова динамічної мережі узгодження змінних у разі необхідності
Обробка видалення фактів	Симетричне видалення	Асиметричне видалення	Асиметричне видалення

Для реалізації Treat алгоритму: представлено узагальнену схему роботи (рис. 4); проведено аналіз існуючого функціоналу СЗЗ оболонки CLIPS, поточної внутрішньої архітектури та особливостей представлення даних; виокремлено

фрагменти архітектури CLIPS, які містять модулі прекомпіляції та обробки Rete мережі потоку даних (рис. 5).

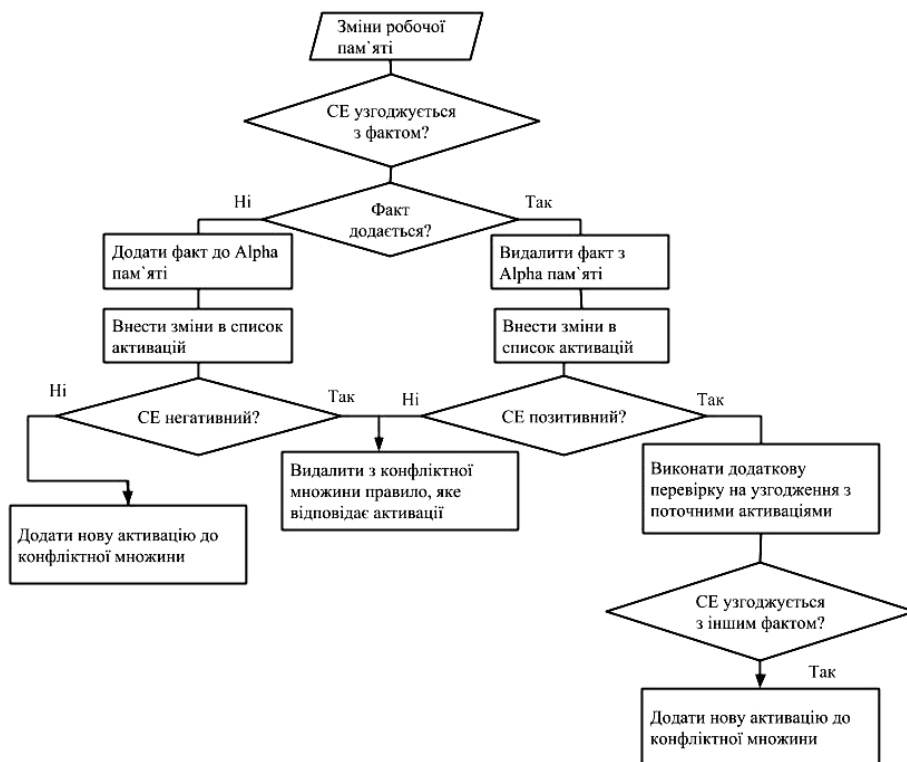


Рис. 4. Схема роботи Treat алгоритму

Компоненти, необхідні для реалізації Treat алгоритму, розподілено за категоріями: незмінні та ті, які потребують модифікації. Штриховими лініями позначені модулі, які не потребують змін.

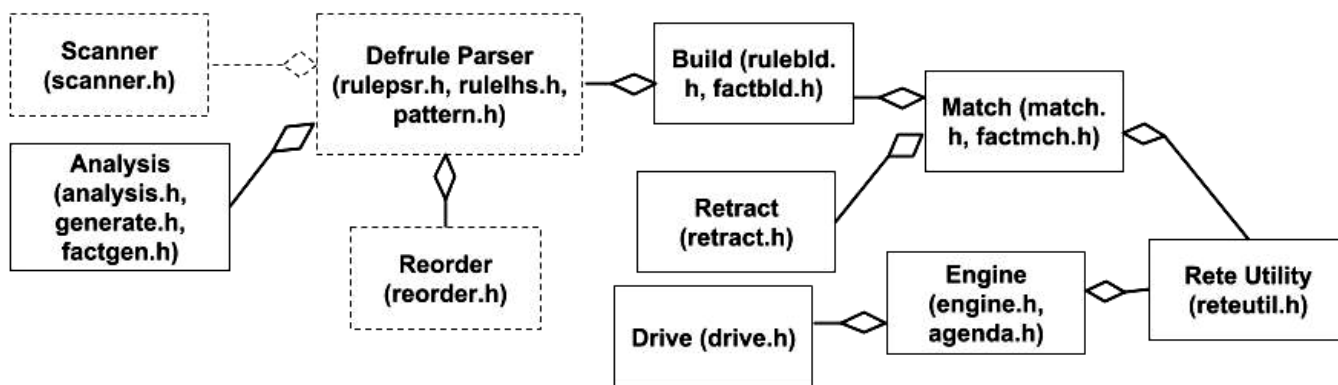


Рис. 5. Модулі співставлення зі зразком в CLIPS

Модулі Scanner та Defrule Parser відповідають за зчитування продукцій з вхідного потоку даних та їх представлення у внутрішньому форматі системи. Модуль Reorder забезпечує додаткову оптимізацію представлення правил в системі. Модуль Analysis відповідає за збереження положення змінних в антецеденті та формування виразів, які будуть використовуватися для оцінки узгодження в β -мережі. Модуль Build забезпечує інтеграцію правила в мережу потоку даних.

В модулі Match представлено основні структури даних для відображення мережі узгодження алгоритму. Модуль Retract відповідає за видалення фактів РП, які узгоджувалися з шаблонами мережі. В поточній версії CLIPS для оптимізації обробки видалення фактів кожен вузол мережі містить посилання на своїх предків та всіх своїх нащадків. Таким чином, для видалення фактів не відбувається додаткових перевірок узгоджень в вузлах мережі за рахунок швидкого пошуку вузлів за посиланнями. Модуль Drive відповідає за обробку мережі потоку даних у випадку додавання нових фактів до РП. Модуль Engine відповідає за обробку поточного списку активацій.

Для проведення експериментів з визначення оптимального за критеріями користувача механізму СЗЗ запропоновано відображення нових характеристик щодо статистичної інформації логічного виведення. Для цього модифіковано:

- команду відображення статистики (watch statistics) для отримання даних щодо загальної кількості узгоджень з умовними елементами,
- команду відстеження процесу співставлення для заданої продукції (match <production_name>) для визначення відповідності умовних елементів і фактів, які з ними узгодилися.

Апробацію проведено на класичних задачах (benchmark) тестування оболонок ПС. Наприклад, в таблиці 3 наведено характеристики логічного виведення в залежності від розміру вхідних даних для задачі Manners (комбінаторної задачі розміщення людей за столом) за стратегією розв'язання конфлікту depth. Програма складається з 8 простих правил. Лише одне з них містить 7 умовних елементів. Для даної задачі характерна низька тимчасова надмірність, тому на кожному кроці проводиться значна кількість обчислень для визначення змін в конфліктному наборі. Результати експерименту доводять, що Treat алгоритм є більш ефективним з точки зору ресурсоемності. В той же час, з точки зору швидкодії Treat не завжди переважає Rete. Це співпадає зі спостереженнями Wright та Marshall і пояснюється тим, що, по-перше, зі збільшенням розміру вхідних даних зростають додаткові витрати на формування динамічної мережі та обробку видалення проміжних результатів узгодження, по-друге, реалізація Rete алгоритму в CLIPS має оптимізацію за рахунок хешування β -пам'яті, ефективність якої проявляється зі збільшенням розмірів РП.

Таблиця 3

Результати роботи алгоритмів СЗЗ для benchmark Manners

Вхідні дані		Затрати пам'яті, Kb		Швидкодія, sec		Кількість узгоджень з елементами	
Характеристика	Розмір (n)	Rete ^{cl}	Treat ^{cl}	Rete ^{cl}	Treat ^{cl}	Rete ^{cl}	Treat ^{cl}
Кількість гостей	16	1045	762	0.016	0.015	6146	4827
	64	3132	2442	0.2539	0.2368	401670	310034
	128	9830	8452	2.531	2.893	3371483	2640989

В загальному випадку швидкодія алгоритмів СЗЗ залежить від кількості правил в БЗ та їх характеристик: кількості умовних елементів антецеденту, порядку їх розташування, кількості негативних умовних елементів та їх положення в антецеденті продукції, кількості повторюваних та унікальних змінних, а також наявності тимчасової надмірності.

Розширена оболонка CLIPS дозволяє обрати механізм співставлення шляхом діалогу налаштувань для поточного логічного виведення (рис 6), передбачає два варіанти застосування: для стаціонарних комп'ютерів у вигляді середовища розробки з графічним редактором та для портативних пристроїв у вигляді бібліотеки.

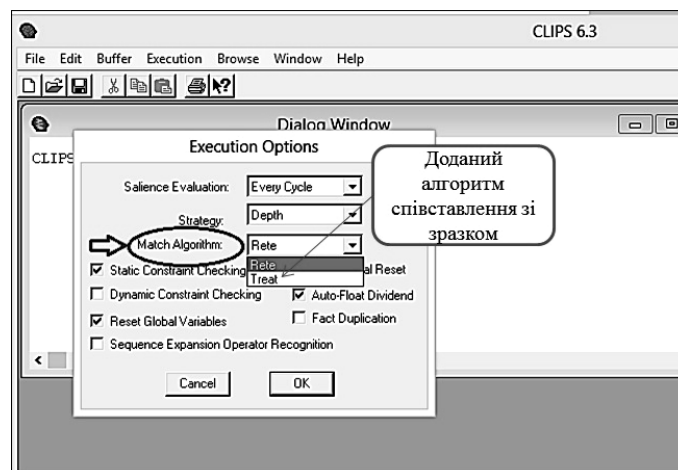


Рис. 6. Вибір способу співставлення в розширеному середовищі

Запропонований спосіб розширення продукційної оболонки CLIPS надає можливість в подальшому доповнювати систему додатковими алгоритмами співставлення зі збереженням єдиного формату представлення даних.

Четвертий розділ присвячено розв'язанню задачі діагностування як однієї з найпоширеніших в застосуванні ПС. Запропоновано у загальному вигляді формалізацію задачі діагностування стану об'єкту за продукційною моделлю представлення знань.

Представлено розроблені прикладні ПС, для яких існують обмеження щодо швидкодії (у складі гібридного програмного комплексу) та використання оперативної пам'яті (на портативних пристроях).

В програмному комплексі моніторингу геологічного середовища в зоні впливу споруд АЕС реалізовано продукційну підсистему, яка здійснює інформаційну підтримку прийняття експертного рішення щодо стану об'єктів проммайданчику.

Об'єктом діагностування може виступати споруда в цілому або її компонента. На вхід підсистеми надходять відомості про об'єкт діагностування, типи дослідження і характеристики додаткових обставин (позаштатна ситуація). На виході отримуються дані у вигляді зведених таблиць, креслень, результатів розрахунків, графіків, витягів з нормативної документації. Також можливий запит додаткової інформації та пояснень проміжних висновків, які стосуються формування сценарію обстеження.

Розвиток мобільних пристроїв сприяв поширенню застосування спеціалізованих експертних систем для працівників різних галузей господарства, наприклад: прикордонників та провідників залізничного транспорту, агрономів, працівників заповідників, рятувальних служб. Подібні експертні системи зазвичай створюються для вирішення задачі експрес-діагностування стану об'єкту за

результатами поточного обстеження в «польових умовах». Тому окремою актуальною задачею є створення методології розробки ПС експрес-діагностування на портативних пристроях.

В архітектурному рішенні (рис. 7) враховано такі вимоги: забезпечення роботи в режимі відсутності доступу до мобільної мережі та Internet; здійснення логічного виведення за даними, отриманими в процесі обстеження; можливість оновлення локальних баз знань та даних при появі зв'язку з мережею Internet без необхідності повторної інсталяції застосунку.

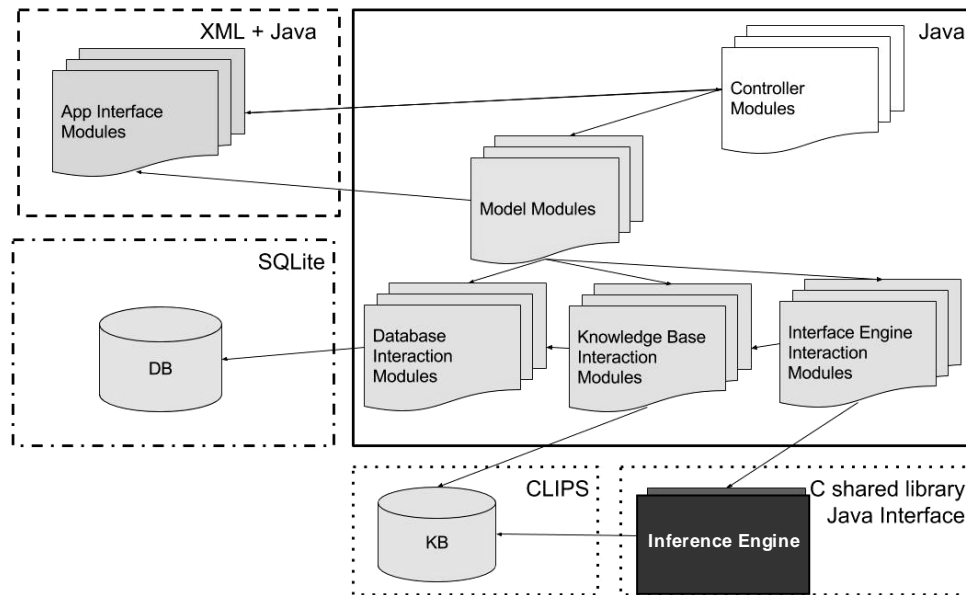


Рис. 7. Архітектура автономного клієнтського застосунку

Бібліотека CLIPS з відповідним інтерфейсом взаємодії з мовою програмування Java інтегрується безпосередньо в застосунок. В роботі запропоновано інтеграцію CLIPS та SQLite в єдину бібліотеку. Для перетворення інформації з БД в факти РП використано структурну подібність представлення фактів ПС та таблиць БД. Для реалізації задачі: 1) створено додатковий модуль роботи з БД в оболонці CLIPS, 2) модифіковано модулі обробки продукцій та конструкцій шаблонів; 3) в модуль команд завантаження даних додано команди роботи з БД. Таким чином, користувачеві надається програмне рішення, яке дозволяє повторне використання реалізації зв'язків між модулями системи. Для створення ПС експрес-діагностування необхідно заповнити БД та БЗ, а також на основі реалізованих базових функцій модифікувати інтерфейс взаємодії з користувачем.

На основі запропонованого загального рішення ПС для портативних пристроїв розроблено ПС діагностування шкідників сільськогосподарських рослин в польових умовах. На вхід система отримує результати обстеження у вигляді інформації щодо стану рослини та умов її зростання, опису шкідника. Додаткові дані запитуються в режимі діалогу з користувачем в процесі виведення логічного заключення (рис. 8 – а). Система надає висновок у вигляді назви, зображення та опису шкідника (рис. 8 – б). За бажанням користувач може зберегти отримані результати в локальну БД.

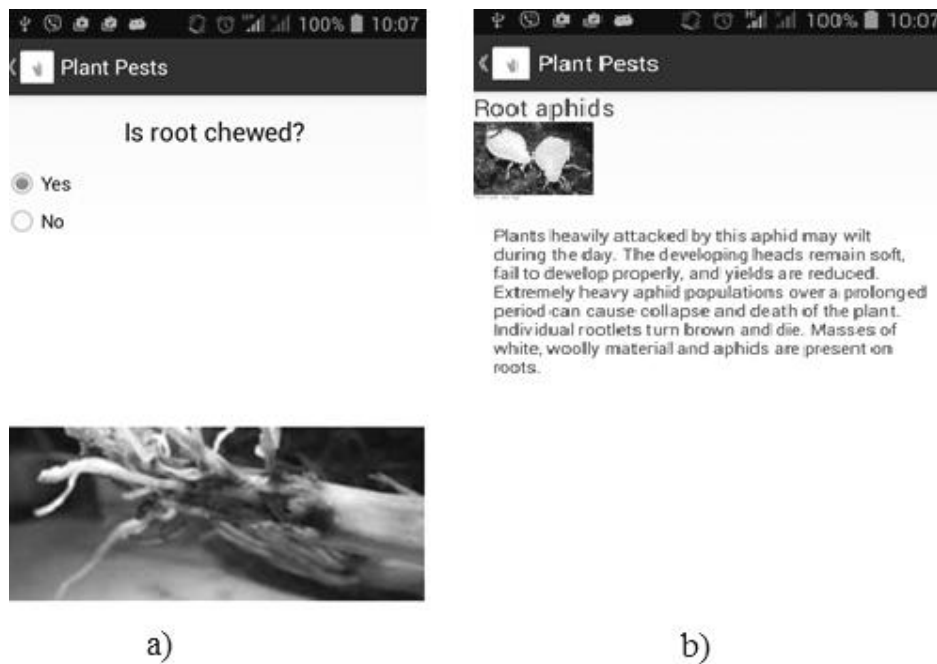


Рис. 8. Фрагменти інтерфейсу: а) – діалог з користувачем, б) – висновок щодо шкідника

ПС визначає шкідників з наступних родин: Aleyrodidae, Mycetophilidae, Thripidae, Lumbricidae, Tomoceridae, Aphididae, Pseudococcidae, Diaspididae, Heteroderidae.

Запропонований програмний інструментарій дозволив вбудувати в прикладні ПС механізм співставлення, який забезпечує максимальну швидкість при діагностуванні стану споруд АЕС та мінімальні витрати оперативної пам'яті для визначення шкідників сільськогосподарських рослин.

ВИСНОВКИ

Основним результатом дисертаційного дослідження є створення теоретичних і практичних методів розробки продукційних систем з урахуванням обмежень щодо швидкодії та/або ресурсоемності, що вдосконалює процес логічного виведення заключення в ПС та має істотне значення для розвитку галузі програмної інженерії.

Наукове значення полягає в створенні формалізації Treat алгоритму, яка дозволить адаптувати існуючий для Rete алгоритму математичний апарат оцінювання ресурсоемності.

Значення для практики полягає в розробці програмного середовища моделювання логічного виведення за базовими підходами до СЗЗ, яке дозволяє підвищити ефективність прикладної ПС за рахунок вибору оптимального Rete або Treat алгоритму.

Це включає в себе такі результати, що мають наукову новизну та практичну цінність:

1. Проведено аналіз підходів до СЗЗ в ПС, а також досліджень з порівняння двох базових способів: Rete та Treat. Визначено, що наразі не представлено

загальних методик, які дозволяють обрати алгоритм СЗЗ, оптимальний з точки зору застосування ресурсів.

2. В результаті дослідження способів формалізації опису продукційних систем обґрунтовано використання формалізації S4, яка описує ПС в термінах логіки першого порядку.

3. Дослідження спеціалізованого програмного інструментарію розробки ПС встановило, що в усіх сучасних середовищах розробки продукційних систем реалізовано лише один з механізмів СЗЗ. Обґрунтовано використання для подальшого дослідження оболонки продукційних систем CLIPS.

4. На базі існуючих формалізацій механізмів ПС в термінах логіки першого порядку розроблено формалізацію співставлення зі зразком за Treat алгоритмом, яка надає можливість порівняння характеристик ресурсоемності за базовими способами СЗЗ.

5. Створено програмний інструментарій розробки оптимальних за критеріями ресурсоемності прикладних ПС на основі двох базових алгоритмів СЗЗ для мінімізації ресурсів при розв'язанні поточної прикладної задачі за рахунок розширення оболонки CLIPS Treat алгоритмом та додатковими засобами визначення ресурсів пам'яті та часу.

6. Проведено обчислювальні експерименти на запропонованому програмному інструментарії на базових тестах випробовування обгортки продукційних систем. Підтверджено достовірність експериментального визначення характеристик ресурсоемності для вибору оптимального способу СЗЗ для поточної задачі.

7. На базі запропонованого програмного інструментарію розроблено загальну методику створення прикладних ПС експрес-діагностування, оптимізованих за швидкістю та/або ресурсами пам'яті, для портативних пристроїв.

8. Результати дисертаційної роботи впроваджено в:

- оптимальній за швидкістю підсистемі інформаційного супроводу діагностування будівель АЕС в програмному комплексі моніторингу стану навколишнього середовища в зоні впливу споруд АЕС;
- оптимальній за ресурсами оперативної пам'яті системі діагностування шкідників сільськогосподарських рослин для використання на портативних пристроях.

СПИСОК ОПУБЛІКОВАНИХ ПРАЦЬ ЗА ТЕМОЮ ДИСЕРТАЦІЇ

1. Шаповалова С. І. Вибір оптимального алгоритму співставлення зі зразком при проектуванні продукційної системи [Текст] / С. І. Шаповалова, О. О. Мажара // Східно-Європейський журнал передових технологій. – 2014. – Вип. 2/2 (68). – С. 43–49. (Входить до Index Copernicus). *Особисто здобувачем запропоновано схему визначення оптимального алгоритму співставлення зі зразком за характеристиками продукційної системи.*

2. Шаповалова С. І. Формалізація базових алгоритмів співставлення зі зразком в продукційних системах / С. І. Шаповалова, О. О. Мажара // Східно-Європейський журнал передових технологій. – 2015. – Вип. 4/3 (76). – С. 22–27.

(Входить до Index Copernicus). *Особисто здобувачем запропоновано формалізацію компіляції та правил переходу Treat мережі.*

3. Шаповалова С. І. Програмний комплекс діагностики економічного стану підприємства [Текст] / С. І. Шаповалова, О. О. Мажара // Математичне та комп'ютерне моделювання. Сер.: Технічні науки : зб. наук. пр. – Кам'янець-Подільський : Кам'янець-Подільськ. нац. ун-т, 2011. – № 5. – С. 209–216. *Особисто здобувачем запропоновано концепцію та структуру програмного комплексу на основі продукційної моделі.*

4. Мажара О. О. Порівняння Rete та Treat алгоритмів співставлення зі зразком / О. О. Мажара // Адаптивні Системи Автоматичного Управління : міжвідомчий наук.-техн. зб. – К. : НТУУ «КПІ», 2014. – Вип. 1 (24) – С. 53–61.

5. Sharovalova S. Production system shell for mobile devices [Text] / S. Sharovalova, O. Mazhara // Математичне та комп'ютерне моделювання. Сер.: Технічні науки : зб. наук. пр. – Кам'янець-Подільський : Кам'янець-Подільськ. нац. ун-т, 2015. – С. 131–141. *Особисто здобувачем запропоновано розширення продукційної оболонки CLIPS для розробки оптимальних за ресурсоемністю прикладних продукційних систем.*

6. Мажара О. А. Реализация Treat алгоритма на основе сопоставления с образцом в программной оболочке CLIPS / О. А. Мажара // Электронное моделирование. – 2015. – Вип. 37 (5). – С. 61–75.

7. Шаповалова С. І. Вирішення задач моніторингу стану будівель та споруд на енергопідприємствах за допомогою продукційних систем [Текст] / С. І. Шаповалова, О. О. Мажара // Інформаційне забезпечення вирішення еколого-енергетичних проблем сталого розвитку суспільства : моногр. / за заг. ред.: Лук'яненко С. О., Караєвої Н. В. ; Національний технічний університет України «Київський політехнічний інститут». – К. : Тамподек ХХІ, 2012. – Вип. 1. – С. 158–162. *Особисто здобувачем запропоновано формалізацію знань на основі продукційної моделі.*

8. Шаповалова С. І. Інформаційне забезпечення виведення заключень щодо стану споруд АЕС [Текст] / С. І. Шаповалова, О. О. Мажара // Автоматизація моніторингу стану довкілля у зоні впливу атомних електростанцій : моногр. / за заг. ред.: Лук'яненко С. О., Шаповалової С. І. – К. : Вид.-полігр. п-во «ТЕКСТ», 2013. – С. 182–189. *Особисто здобувачем розроблено структуру представлення інформації у прикладній продукційній системі як систему взаємопов'язаних баз даних і знань.*

9. Шаповалова С. І. Середовище CLIPS розробки експертних систем малого бізнесу для планшетів [Текст] / С. І. Шаповалова, О. О. Мажара // Економічна безпека держави: стратегія, енергетика, інформаційні технології : моногр. / за наук. ред.: Лук'яненко С. О., Караєвої Н. В. – К. : Вид-во ООО «Юрка Любченка», 2014. – С. 424–432. *Особисто здобувачем запропоновано модифіковане середовище розробки продукційних систем CLIPS.*

10. Мажара О. О. Обчислювальний механізм розв'язання задачі діагностування стану споруд і ведення супровідної документації / О. О. Мажара // Інформаційне і програмне забезпечення системи моніторингу стану довкілля у зоні

впливу атомних електростанцій : моногр. / за заг. ред. Лук'яненка С. О. – К. : Вид.-полігр. п-во «Текст», 2015. – С. 164–174.

11. Авторське свідоцтво № 43282. Україна. Комп'ютерна програма «Вирішення задачі пошуку в просторі станів у середовищі CLIPS» / С. І. Шаповалова, О. О. Мажара (Україна). – № 43359 ; заявл. 14.02.12 ; опубл. 02.07.12, Бюл. № 27. *Особисто здобувачем запропоновано розв'язання тестових задач пошуку на мові CLIPS.*

12. Мажара О. А. Автоматизация сбора информации для обеспечения технического осмотра зданий и сооружений АЭС [Текст] / О. А. Мажара, С. И. Шаповалова // Энергетика в современном мире : сб. тр. I междунар. Интернет-конф., (Казань, 8 февр. 2013 г.). – Казань : Изд-во «Казанский университет», 2013. – С. 24–26. *Особисто здобувачем обґрунтовано використання продукційної моделі для автоматизації підбору інформації щодо технічних оглядів споруд АЕС.*

13. Шаповалова С. І. Програмно-методичний комплекс вивчення дисципліни «Методи та системи штучного інтелекту» [Текст] / С. І. Шаповалова, О. О. Мажара // Інформаційні технології в освіті, науці і техніці : тези доп. II Міжнар. наук.-практ. конф., (Черкаси, 24–26 квіт. 2014 р.) : у 2-х т. – Черкаси : ЧДТУ, 2014. – Т. 2. – С. 99–100. *Особисто здобувачем запропоновано реалізацію тестових задач на мовах Prolog і CLIPS.*

14. Шаповалова С. І. Формалізація представлення продукційної моделі [Текст] / С. І. Шаповалова, О. О. Мажара // Системний аналіз та інформаційні технології : матеріали 16-ї Міжнар. наук.-техн. конф. SAIT 2014, (Київ, 26–30 трав. 2014 р. / ННК «ІПСА» НГУУ «КП»). – К. : ННК «ІПСА» НГУУ «КП», 2014. – С. 277–278. *Особисто здобувачем представлено основні підходи до формалізації представлення продукційної моделі.*

15. Шаповалова С. І. Вибір стратегій розв'язання конфліктів в продукційних системах [Текст] / С. І. Шаповалова, О. О. Мажара // Сучасні інформаційні та електронні технології : труди XV Міжнар. наук.-практ. конф., (Одеса, 26–30 трав. 2014 р.) : у 2-х т. – Одеса : Политехперіодика, 2014. – Т. 1. – С. 34–35. *Особисто здобувачем формалізоване представлення і реалізацію тестових задач на мові CLIPS.*

16. Шаповалова С. І. Вибір та обґрунтування програмних засобів розробки продукційних систем [Текст] / С. І. Шаповалова, О. О. Мажара // Обчислювальний інтелект (результати, проблеми, перспективи) : матеріали III Міжнар. наук.-техн. конф., (Черкаси, 12–15 трав. 2015 р.). – Черкаси : Вид. Чабаненко Ю., 2015. – С. 406. *Особисто здобувачем запропоновано характеристики поточної БЗ, які впливають на необхідну функціональність продукційних оболонок.*

17. Мажара О. О. Розширення засобів налагодження середовища CLIPS / О. О. Мажара // Сучасні аспекти розробки програмного забезпечення : зб. тез II наук.-практ. конф. молодих вчених і спеціалістів в області розробки програмного забезпечення, (Київ, 22 трав. 2015 р.). – Черкаси : Вид. Чабаненко Ю. А., 2015. – С. 44–46.

18. Шаповалова С. І. Формалізація співставлення зі зразком за Treat-алгоритмом / С. І. Шаповалова, О. О. Мажара // Сучасні інформаційні та електронні технології : труди XVI Міжнар. наук.-практ. конф., (Одеса, 25–29 трав. 2015 р.). – Одеса : Политехперіодика, 2015. – С. 44–45. *Особисто здобувачем формалізовано в термінах логіки першого порядку правила побудови та обробки мережі даних Treat-алгоритму.*

АНОТАЦІЯ

Мажара О. О. Програмний інструментарій розробки продукційних систем з урахуванням обмежених ресурсів. – Рукопис.

Дисертація на здобуття наукового ступеня кандидата технічних наук за спеціальністю 01.05.03 – Математичне та програмне забезпечення обчислювальних машин і систем. – Київський національний університет імені Тараса Шевченка, Київ, 2016.

Дисертація присвячена вирішенню проблеми створення теоретичних та алгоритмічних основ розробки продукційних систем з мінімальними затратами на логічне виведення заключення за критеріями швидкодії та/або ресурсоємності.

Розроблено формалізацію співставлення зі зразком за Treat способом в термінах логіки першого порядку для створення формальних засад порівняння базових способів співставлення зі зразком.

Запропоновано програмне середовище експериментального визначення оптимального способу співставлення в поточній прикладній задачі. Удосконалено оболонку продукційних систем CLIPS за рахунок реалізації Treat алгоритму співставлення зі зразком для розробки прикладних експертних систем з мінімальним використанням ресурсів пам'яті та часу.

Удосконалено програмне забезпечення розробки продукційних систем експрес-діагностування за рахунок інтеграції механізму логічного виведення з мінімальними вимогами до оперативної пам'яті для використання на портативних пристроях в умовах експлуатації з обмеженим або відсутнім доступом до мережі зв'язку та Internet.

Ключові слова: продукційні системи, співставлення зі зразком, Rete алгоритм, Treat алгоритм, CLIPS.

АНОТАЦІЯ

Мажара О. А. Программный инструментарий разработки продукционных систем с учетом ограниченных ресурсов. – Рукопись.

Диссертация на соискание ученой степени кандидата технических наук по специальности 01.05.03 – Математическое и программное обеспечение вычислительных машин и систем. – Киевский национальный университет имени Тараса Шевченка, Киев, 2016.

Диссертация посвящена решению проблемы создания теоретических и алгоритмических основ разработки продукционных систем (ПС) с минимальными

затратами на логический вывод заключения по критериям быстродействия и/или ресурсоёмкости.

Тенденция к интеллектуализации программного обеспечения способствовала более широкому распространению продукционных систем. Это относится, прежде всего, к их внедрению как подсистем в гибридных программных комплексах, а также использованию как автономных систем на портативных устройствах. В первом случае требуется максимальное быстродействие, во втором – минимальное использование оперативной памяти. Поэтому проблема создания программного инструментария разработки прикладных продукционных систем, оптимальных по заданным пользователем критериям ресурсоемкости, актуальна и имеет практическое значение.

Наиболее ресурсоёмким при осуществлении логического вывода является механизм сопоставления с образцом, который определяет истинность логического выражения условной части продукций.

Целью диссертационной работы является создание теоретических и алгоритмических основ разработки продукционных систем, требующих минимальных затрат ресурсов памяти и времени.

В работе решены следующие задачи: анализ подходов к сопоставлению с образцом в продукционных системах; исследование подходов к формализации продукционных систем для представления в едином формате описания сопоставления с образцом по базовым способам: Rete и Treat; исследование современных оболочек продукционных систем для выбора инструментария реализации ПС для портативных устройств; разработка формализации сопоставления с образцом для Treat алгоритма в терминах логики первого порядка; создание программного инструментария разработки оптимальных по заданным критериям ресурсоёмкости продукционных систем на базе CLIPS; разработка общего представления ПС экспресс-диагностирования для портативных устройств в условиях ограниченного доступа к сетям мобильной связи и Internet; реализация программного обеспечения решения прикладных задач диагностирования в условиях ограниченных ресурсов.

На основе анализа подходов к сопоставлению с образцом в ПС, а также исследований по сравнению двух базовых способов: Rete и Treat определено, что не представлено общих методик, позволяющих выбрать алгоритм сопоставления с образцом, оптимальный с точки зрения использования вычислительных ресурсов. На основании формализации Rete алгоритма в терминах логики первого порядка разработана формализация Treat алгоритма, что позволяет проводить формальное сравнение базовых способов сопоставления с образцом.

В результате исследования специализированного программного инструментария разработки продукционных систем определено, что во всех современных оболочках реализован только один из механизмов сопоставления с образцом. Обосновано усовершенствование оболочки продукционных систем CLIPS за счет реализации Treat алгоритма для экспериментального определения способа сопоставления с образцом для текущей задачи.

Для продукционных систем экспресс-диагностирования на портативных устройствах в условиях ограниченного или отсутствующего доступа к сетям связи и

Internet предложено решение в виде общей архитектуры системы и архитектуры клиентского приложения. В предложенном решении учтены следующие общие требования: возможность сохранения данных на портативном устройстве, логический вывод по данным, полученным в процессе обследования, обновление локальных баз знаний и данных при появлении связи с сетью Internet.

На основании предложенных решений разработана подсистема информационного сопровождения диагностирования зданий АЭС в программном комплексе мониторинга состояния окружающей среды в зоне влияния сооружений АЭС и автономное приложение диагностирования вредителей сельскохозяйственных растений на портативных устройствах.

Ключевые слова: производственные системы, сопоставление с образцом, Rete алгоритм, Treat алгоритм, CLIPS.

ABSTRACT

Mazhara O. O. Software development tools of production systems with limited resources. – Manuscript.

The thesis for the candidate degree in engineering sciences, within the subfield 01.05.03 – Mathematical software of computing machines and systems. – Taras Shevchenko National University of Kyiv, Ministry of education and science of Ukraine, Kyiv, 2016.

The thesis addresses the problem of creating theoretical and algorithmic foundations of the production systems development with the minimal cost of the inference by the measures of performance and / or resource intensity.

The formalisation of the Treat match algorithm was developed in terms of first-order logic to create formal principles for the comparison of basic match methods.

The experimental software environment was proposed to determine the optimal match algorithm for the current application. CLIPS production system shell was improved by the implementation of Treat match algorithm for the development of applied expert systems with minimal usage of memory and time resources.

The software for development of express-diagnostic production systems was improved by integrating the inference mechanism with minimum RAM requirements for the use on portable devices in environments with the limited or no access to the mobile net or Internet.

Based on the proposed solutions, the information support subsystem for the diagnostics of the NPP buildings in the software complex for monitoring environmental conditions in the area of nuclear power plants impact and the standalone application for diagnosing pests of crops on portable devices were developed.

Key words: production systems, match, Rete algorithm, Treat algorithm, CLIPS.

Підписано до друку 26.04.2016 р. Формат 60x90/16.
Ум. друк. арк. 0,9. Обл.-вид. арк. 0,9.
Тираж 100. Зам. 38.

«Видавництво “Науковий світ”»[®]
Свідоцтво ДК № 249 від 16.11.2000 р.
м. Київ, вул. Казимира Малевича (Боженка), 23, оф. 414.
200-87-15, 050-525-88-77
E-mail: nsvit23@ukr.net
Сайт: nsvit.cc.ua