УДК 621.382

Р.П. Базилевич, І.Б. Гудз, Р.А. Ждан
Національний університет "Львівська політехніка",
кафедра програмного забезпечення

ІЄРАРХІЧНА КЛАСТЕРИЗАЦІЯ СКЛАДНИХ СХЕМ

© Базилевич Р.П., Гудз І.Б., Ждан А.Р., 2005

Розглянуто особливості програмної реалізації побудови дерева оптимального згортання схеми. Розкрито основні підходи формування пар елементів для утворення кластерів. На прикладі декількох тестів проаналізовано отримані результати.

The features of programmatic realization construction of Optimal Reduction Circuit Tree. Basic approaches of forming pair of elements are exposed for formation clusters. The analysis of the got results is conducted on the example of a few tests

Вступ. Розбиття складних схем на частини є однією з найважливіших часткових задач у проектуванні сучасних мікропроцесорів, складних електронних схем, НВІС, багатопроцесорних систем. Складність розв'язування задачі зумовлена неполіномінальною обчислювальною складністю та великою розмірністю схем (сотні тисяч базових елементів).

Для розв'язання задачі можна застосовувати метод оптимального згортання схеми, який полягає в паралельно-послідовному формуванні підсхем, що мають однакові або близькі за взаємною зв'язністю елементів характеристики [1]. Ідея методу оптимального згортання схеми полягає в паралельно-послідовному формуванні підсхем, які мають однакові або близькі за взаємною зв'язністю елементів характеристики. Метод виявляє ієрархічну кластерну структуру схеми на основі багатокрокового процесу ідентифікації та об'єднання підсхем, які щільніші (густіші), ніж інші. Бажано виявити реальні згустки схеми, їхнє взаємне входження – менших в більші та зв'язки між ними. Це сприятиме виділенню окремих сильно зв'язних груп, а також буде вказувати на їхнє взаємне входження. Метод виявляє ієрархічну кластерну структуру схеми на основі багатокрокового процесу, виділяє реальні згустки схеми, їхнє взаємне входження – менших в більші та зв'язки між ними. Наявність такої інформації значно полегшить розв'язування наступної задачі – розбиття схеми на частини із забезпеченням бажаних характеристик. Для виявлення природних згустків (кластерів) схеми можуть бути використані певні формальні критерії, які враховують такі показники, як зв'язність між елементами чи їхніми групами, а також якісні, що враховують функційні властивості груп елементів.

Формулювання задачі. Більшість реальних складних схем можна подати як пару:

$$S=(P, E),$$

де $P=\{p_1,...,p_n\}$ – множина елементів; $E=\{e_1,...,e_m\}$ – множина зв'язків між елементами.

Необхідно виявити сильно зв'язані згустки системи та побудувати дерево оптимального згортання T^R, яке відображає ієрархічне входження малих згустків (кластерів) у більші та корінь якого відповідає всій схемі. Якщо система містить кілька незв'язаних підсистем – будують ліс дерев $F^R=\{T^R_1,...,T^R_n\}$.

Алгоритмізація задачі формування кластерів. Метод оптимального згортання схеми базується на виявленні сильно зв'язних згустків схеми – кластерів. На кожному кроці два елементи (кластери) об'єднуються в один, кластери формують дерево оптимального згортання T^R, яке відображає ієрархічне входження малих згустків у більші та корінь якого відповідає всій схемі.

Базові кроки реалізації алгоритму побудови дерева оптимального згортання схеми T^R є такими:

1. Формування пар елементів/кластерів, що зв'язані між собою.
2. Визначення критерію об'єднання для виділених пар.

3. Впорядкування пар за значенням критерію.
4. Вибір пар елементів/кластерів для об'єднання та утворення нових кластерів.
5. Вилучення пар елементів, які об'єднались в кластери із списку пар.
6. Модифікація всіх пар елементів, в яких наявні елементи, що об'єднались.
7. Визначення нового значення критеріїв та впорядкування їх у списку пар.
8. Перехід на крок 4, поки не будуть об'єднані всі кластери.

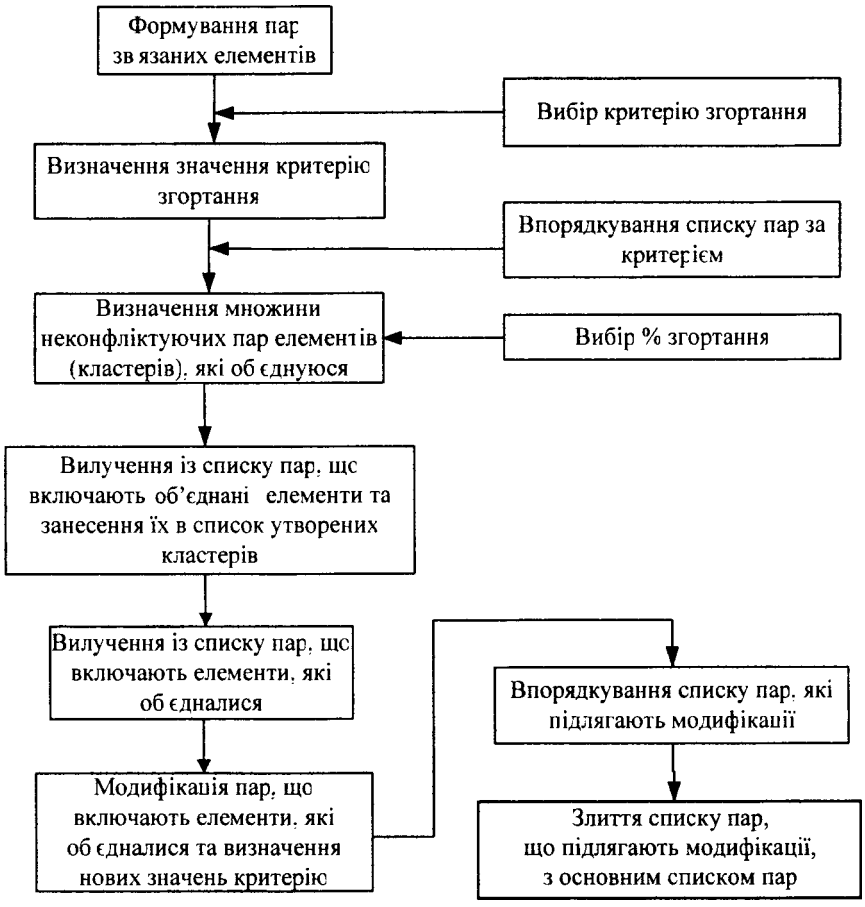

Рис. 1. Алгоритм побудови дерева згортання

Опис структур даних. Структура даних програми ґрунтується на використанні однонапрямлених списків та списків із розгалуженням. Також використовують динамічні масиви списків і динамічні масиви простих типів даних. У методі оптимального згортання схеми застосовано низку процедур роботи з списками, зв'язки кожної пари подають у вигляді списків, які характеризують її внутрішні та зовнішні зв'язки. Робота зі списками зв'язків побудована так, щоб звести до мінімуму затрати пам'яті і повторні обчислення.

Пари елементів будують із списків ListElement і ListNet. ListElement – це спискова структура, яка містить інформацію про елементи, ListNet – спискова структура, яка містить інформацію про зв'язки між елементами. Пара елементів – це структура, яка містить: назву першого елемента; назву другого елемента; список зовнішніх зв'язків; список внутрішніх зв'язків, інформацію про кількість внутрішніх і зовнішніх зв'язків; значення критерію згортання, яке визначається на основі чисел зовнішніх і внутрішніх зв'язків залежно від вибору критерію згортання; два прапорці, які вказують, з чого утворена пара (елементів, кластерів, елемента і кластера); список зв'язків для першого елемента (кластера); список зв'язків для другого елемента (кластера).

На рис. 2. подано інформацію, що зберігається в пам'яті про пару зв'язаних елементів. Введено такі позначення: $p_\alpha p_\beta$ – елементи, або кластери, які утворюють пару $p_\alpha p_\beta$; $e^{in}_{\alpha\beta}$, $e^{ex}_{\alpha\beta}$ –

список внутрішніх і зовнішніх зв'язків між p_α *і* p_β; к1, к2 – критерії згортання; η^{in}, η^{ex} – кількість внутрішніх і зовнішніх зв'язків пари p_α p_β; *f1, f2* – прапорці, які вказують, чи p_α *і* p_β є елементами, чи кластерами; e_α, e_β – списки всіх зв'язків для елементів (кластерів) *α і β; C* – назва утвореного кластера; η_c –кількість кластерів, яку об'єднує кластер С; *Itr* – кількість ітерацій, потрібних для згортання кластера *C*. На рис. 3 подана інформація про модель дерева згортання.

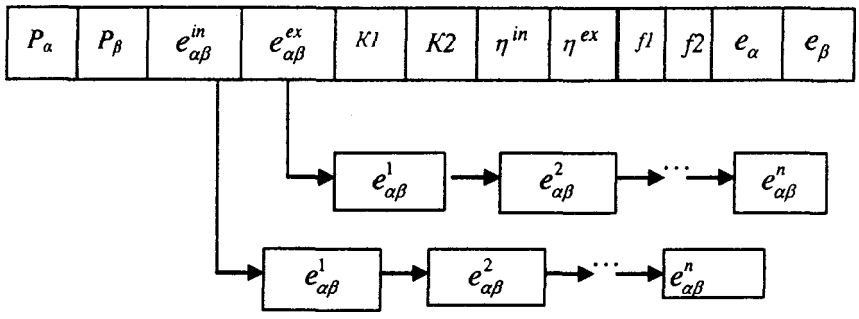

Рис. 2. Представлення даних про пару зв'язаних елементів

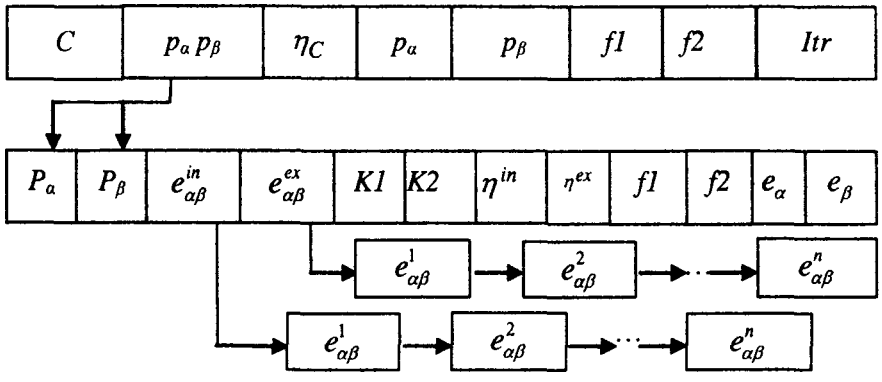

Рис. 3. Представлення даних про модель дерева згортання

Особливості програмної реалізації. Розроблена програмна система (рис. 4) складається з трьох частин (потоків), які взаємодіють між собою, і працює виключно з оперативною пам'яттю для досягнення високої швидкодії. Для розроблення програмної системи використано об'єктно-орієнтований підхід.

Потік побудови дерева оптимального згортання взаємодіє з потоками розбиття і оптимізації, що зменшує затрати часу, оскільки не потрібно зчитувати інформацію про зв'язки і елементи з файла, ці дані зчитуються з оперативної пам'яті і не потрібно будувати список зв'язків і елементів. Окрім того, всі три потоки постійно взаємодіють з модулем роботи з списками, в якому реалізована базові процедури роботи із списками. Розроблена програмна система може використовуватись як допоміжний засіб у САПР, а також для задач пакування.

Під час побудови пари елементів визначають значення її критерію згортання і пару додають до списку пар з аналогічним значенням критерію (рис. 1). Так формуються блоки пар, в яких містяться пари з однаковим значенням критерію. Доступ до цих блоків здійснюється за допомогою індексного масиву. Наприклад, для того, щоб здійснити доступ до блока із значенням критерію пар, який дорівнює 3, потрібно звернутись до третього елемента масиву, який є вказівником на початок блока, що нас цікавить. В процесі згортання це дає можливість вибирати пари, які будуть згортатися, з окремого блока або з декількох блоків залежно від вибору параметрів згортання. Пари, які підходять для згортання з блока, записують у спискову структуру, яка містить інформацію про кількість елементів у списку і вказівник на відповідну пару у списку пар. Пари, які мають спільні елементи із елементами, що згорнулись, записують у аналогічну спискову структуру (список конфліктних пар).

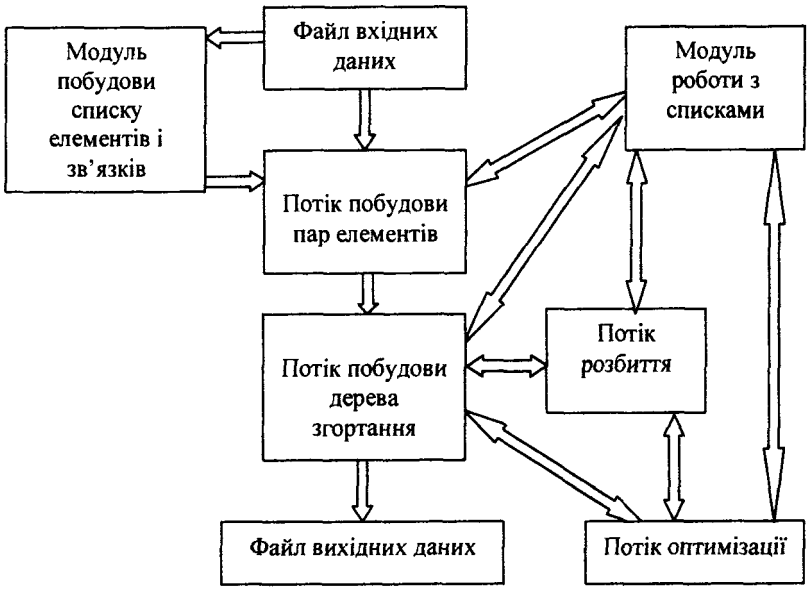

Рис. 4. Програмна система реалізації методу оптимального згортання схеми

Для ефективнішого визначення списку пар для згортання і списку пар, які конфліктують, застосовуємо два індексні масиви (перший індексує елементи, другий – кластери). Індексами в цих масивах є назва елемента і кластера, які є цілочисловими значеннями.

Якщо пару елементів (кластерів) заносять у список пар, які згортаються, то в індексний масив, який індексує елемент або кластер залежно від складу пари, заносять значення, що вказує на пару з відповідними елементами (кластерами), яка уже наявна у списку пар для згортання. Заносячи наступний елемент у список пар для згортання, переглядаємо масив за відповідним індексом, якщо пара з елементом чи кластером уже наявна, то записуємо її у список конфліктуючих пар, в противному разі в список пар для згортання.

При модифікації списку пар використовуємо індексні масиви, індексами в яких виступають елементи (для масиву, який індексує елементи) і кластери для відповідного масиву. Значення цих масивів є вказівниками на пари із списку пар. Наприклад, п'ятий елемент масиву (який індексує елемент) вказує на всі пари, в яких міститься п'ятий елемент, відповідно п'ятий елемент масиву (який індексує кластер) вказує на всі пари, в яких міститься п'ятий кластер. Це дає можливість під час модифікації пар не затрачати додатково час на пошук пар у всьому списку, почергово переглядаючи його, а за відповідними індексами отримувати список пар, що нас цікавлять.

Усі модифіковані пари записують у спискову структуру, яка зберігає список модифікованих пар. У цьому списку пари дублюються, для вилучення їх із цього списку і загального списку пар використовують індексні масиви, які використовуються при модифікації. У такому разі звертаємось за цим індексом до масиву і одержуємо вказівник на список пар. Якщо трапляється ситуація, що в списку є вказівники, які вказують на декілька пар, то виділяються дублюючі пари із цього списку. Після вилучення всіх пар, які повторюються, вставляють пари у список пар, за відповідним значенням критерію згортання. Пари вставляють за аналогічним принципом, як при побудові списку пар з використанням блоків. Залежно від критерію згортання вставляють пари у блок, який відповідає критерію згортання.

Пари, які згорнулись, записують у масив. Елементи цього масиву є структурами і містять в собі таку інформацію: назву кластера; кількість елементів, які об'єднує кластер; назву елементів чи кластерів, які утворили цей кластер; інформацію про кількість зв'язків кластера.

Експериментальні дослідження. Дослідження здійснено на основі реальних схем фірми IBM [4] на пакеті з 11 тестів. Розмірність тестів лежить в діапазоні від 12752 до 70558 базових елементів.

У таблиці наведено характеристики тестів, затрати часу на побудову дерева згортання, загальний час (побудова списку елементів, зв'язків, пар елементів, дерева згортання). За критерій згортання вибрано різницю чисел внутрішніх і зовнішніх зв'язків.

Назва тесту	Кількість елементів	Кількість зв'язків	Кількість пар елементів	Час побудови дерева згортання, с	Загальний час, с
Ibm01	12752	14111	109183	13	83
Ibm02	19601	19584	343409	53	284
Ibm03	23136	27401	206069	38	157
Ibm04	27507	31970	220423	45	229
Ibm05	29347	28446	349676	61	271
Ibm06	32498	34826	321308	65	368
Ibm07	45926	48117	377000	78	596
Ibm08	51309	50513	732550	115	967
Ibm09	53395	60902	478777	92	745
Ibm10	69429	75196	707969	102	997
Ibm11	70558	81454	508442	84	922

Висновки. Розроблено програмну систему для ієрархічної кластеризації складних схем. Експериментальні дослідження на 11 тестах фірми IBM підтвердили високу швидкодію системи, що робить можливим її використовувати для задач великих розмірностей.

1. Базилевич Р.П. Декомпозиционные и топологические методы автоматизированного конструирования электронных устройств. – Львов: Вища шк., 1981. – 168 с. 2. Базилевич Р.П., Подольський І.В. Особливості організації пакета програм для ієрархічної кластеризації схем // Вісн. Нац. ун-ту "Львівська політехніка". – 2002. – № 440. – С. 139–144. 3. Базилевич Р.П., Подольський І.В., Базилевич П.Р. Ієрархічна кластеризація – ефективний засіб розв'язування неполіноміальних комбінаторних задач схемного типу високої розмірності // Штучний інтелект. НАН України. – 2002. – № 3. – С. 474–483. 4. Charles J. Alpert, The ISPD98 Circuit Benchmark Suite. ISPD98 Monterey CA USA, 1998.

УДК 681.3

Л.О. Березко, В.В. Троценко
Національний університет "Львівська політехніка",
кафедра електронних обчислювальних машин

ОПТИМАЛЬНЕ РОЗМІЩЕННЯ МОБІЛЬНИХ АГЕНТІВ У НЕЧІТКОМУ ІНФОРМАЦІЙНОМУ СЕРЕДОВИЩІ

© Березко Л.О., Троценко В.В., 2005

Оптимальність розміщення мобільних вимірювальних агентів розглядається з аспекту критерію оцінки збігу прогнозів значень параметрів нечіткого середовища, які досліджують за певний період та реальних даних, отриманих за наступний період.

The optimality of the mobile agents placement is examined on the view of the evaluation criterion of the fuzzy environment parameters values concurency forecasts, which are examined during some period of time and real data, collected during the next period.

Вступ. Запропонований в [4] підхід до побудови конфігурованих вимірювально-обчислювальних мереж (ВОМ) включає такі складові: