

УДК 004.925.8

М.В. Черкаський, А.О. Саченко, доктори техн. наук

ПАРАМЕТРИЧНІ МОДЕЛІ АЛГОРИТМІВ

***Анотація.** Запропоновано концепцію алгоритму з використанням узагальненої параметричної моделі. Описаний підхід обґрунтовує зростання набору характеристик складності, в тому числі часових, апаратних (об'єкт), ємнісних, програмних та структурних. Запропоновану параметричну модель алгоритму впроваджено в процесі розробки спеціалізованих процесорів.*

***Ключові слова:** концепція алгоритму, параметрична модель, характеристики складності.*

M.V. Cherkassky, ScD., A.O. Sachenko, ScD.

ALGORITHMS PARAMETRICAL MODELS

***Abstract.** A concept of algorithm is proposed using the generalized parametrical model. A described approach creates conditions for a growth of complexity characteristics set including time, hardware (object), capacitive, program and structural ones. The implementation of parametrical models is considered for specialized computer systems.*

***Keywords:** algorithm conception, parametric model, difficulty characteristics.*

Н.В. Черкасский, А.А. Саченко, доктора техн. наук

ПАРАМЕТРИЧЕСКИЕ МОДЕЛИ АЛГОРИТМОВ

***Аннотация.** Предложена концепция алгоритма с использованием обобщенной параметрической модели. Описанный подход обосновывает возрастание набора характеристик сложности, в том числе временных, аппаратных (объект), емкостных, программных и структурных. Предложенная параметрическая модель алгоритма внедрена в процессе разработки специализированных процессоров.*

***Ключевые слова:** концепция алгоритма, параметрическая модель, характеристики сложности.*

Вступ

Процес уточнення поняття “алгоритм” триває декілька сторіч і продовжується разом з розвитком науки і техніки до теперішнього часу. В попередній доповіді авторів був представлений варіант уточнення на основі SH-моделі [1]. Тепер пропонуються результати досліджень поняття “алгоритм” на основі параметричних моделей. Для цього запропонована узагальнена параметрична модель. Вона не суперечить існуючим моделям, дозволяє установити між ними логічні зв'язки, конструювати нові моделі, створювати нові підходи до оптимізації апаратних та апаратно-програмних комп'ютерних засобів.

Процес розроблення алгоритму починається із структурного синтезу. Для синтезу використовуються моделі алгоритмів у вигляді словесного опису, математичної функції, графічної залежності, блок-схеми програми, програми на мові високого рівня та інш. В результаті структурного синтезу отримують алгоритм, який задає порядок виконання обчислювальних операцій в просторі і часі.

На наступному етапі конструювання

© Черкаський М.В., Саченко А.О., 2011

ефективного алгоритму проводиться параметрична оптимізація. Для будь-якого математичного або технічного об'єкта оптимізація складається з трьох операцій: синтезу, аналізу та оптимізації. Початковими даними для параметричної оптимізації є заданий набір характеристик. Для алгоритму це сукупність характеристик складності. Результатом синтезу є набір параметрів об'єкта. Аналіз синтезованого об'єкта показує, наскільки характеристики синтезу відрізняються від заданих. Якщо розбіжність характеристик не припустима, проводять параметричну оптимізацію об'єкта.

Структура алгоритму

Для побудови ефективних алгоритмів будемо використовувати такі вісім параметрів алгоритму: правило початку, правило вводу даних, система вхідних даних, правило безпосереднього перероблення, система проміжних результатів, система кінцевих результатів, правило виводу, правило закінчення. Неповний список перерахованих параметрів відомий [2]. Але в сучасних літературних джерелах конструювання ефективних алгоритмів з використанням параметричної оптимізації не зустрічається. Описуються лише несистематизовані способи отримання

алгоритмів з мінімізованою часовою складністю. Не описується і сама параметрична модель.

Алгоритм – фундаментальне поняття математики, йому не можна дати точне математичне визначення. Відсутність такого визначення не заважає інтуїтивному розумінню його змісту. Але інтуїтивне розуміння не дає змоги використовувати його як інструмент дослідження обчислювального процесу. Для цього потрібні моделі.

Найбільш узагальненою, на думку авторів, є параметрична модель алгоритму (рис.). Зміна параметрів викликає зміну значень характеристик складності. Якщо кількість характеристик складності, які потребують оптимізації, більше однієї, існуючі способи не ефективні. Використання параметричної моделі алгоритму дозволяє позбутись недоліків існуючих способів побудови ефективних алгоритмів.

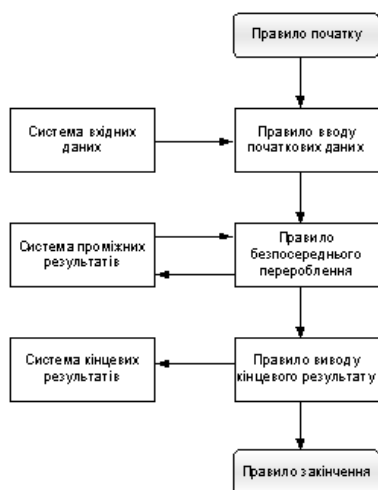


Рис.1. Параметрична модель алгоритму

Вона складається з правил і систем даних. Правила називають стани і операції алгоритму. До станів належать правило початку та правило закінчення, до операцій відносяться правило вводу даних, правило виводу результатів, правило безпосереднього перероблення. До складу системи даних входять система вхідних даних, проміжних та кінцевих результатів. Параметри об'єднані в єдину структуру. Послідовність виконання алгоритму задається лініями зв'язку між параметрами. Для формального опису параметричної моделі використаємо кортеж параметрів:

$$M_{par} : \langle A, Q, q_0, q_f, I, O, W \rangle, \quad (1)$$

де A – кінцева множина символів зовнішнього алфавіту, які позначають елементи систем вхідних даних, проміжних та кінцевих результатів; Q – кінцева множина символів внутрішнього алфавіту; q_0, q_f – початковий і кінцевий стани роботи моделі, $q_0, q_f \in Q$; I, O – правила вводу і виводу; W – правило безпосереднього перероблення.

Розглянута параметрична модель, назвемо її узагальненою, може бути використана як основа для конструювання та дослідження нових і вже існуючих моделей. Ці моделі складають два класи: абстрактні алгоритми і формальні алгоритмічні системи. Вони розрізняються напрямком використання, а також повнотою та деталізацією параметрів алгоритму.

Абстрактні моделі алгоритму

Модель алгоритму належать до класу абстрактних, якщо правило безпосереднього перероблення не містить засобів здійснення обчислень: $G \notin W$, де G – конфігурація “апаратних” засобів.

Поширеним підкласом моделей абстрактних алгоритмів є функціональні залежності $F: y = f(x)$; $F \subset W$. З параметричної моделі тут зафіксовані лише правило безпосереднього перероблення, системи вхідних даних та кінцевих результатів. Інші параметри ігноруються:

$$M_{af} : \langle A, F \rangle, \quad (2)$$

де F – алгоритм у вигляді функціональної залежності.

Метою дослідження і конструювання алгоритмів цього підкласу є оцінка і мінімізація лише однієї характеристики – часової складності [3].

До іншого підкласу абстрактних алгоритмів належать блок-схеми програм і програми на мовах високого рівня. Розробка та практичне використання моделей цього підкласу перетворилося в одну з провідних індустрій сучасності. Модель програми містить всі параметри, задекларовані в параметричній моделі. Є тільки одна особливість: правило безпосереднього перероблення задано лише набором взаємозв'язаних інструкцій – програмою. Правило безпосереднього

перероблення не містить засобів виконання інструкцій. Програма відокремлена від засобів її реалізації: $P \subset W$. Модель визначається наступним кортежем:

$$M_{pr} : \langle A, Q, q_0, q_f, I, O, P \rangle. \quad (3)$$

Основним об'єктом дослідження цієї моделі є часова складність. Програмний продукт – результат інтелектуальної діяльності. Але кількісних характеристик складності програмної моделі алгоритму, які б оцінювали вклад людини в розробку програм, практично немає.

Таку оцінку дозволяє робити псевдо SH-модель [4]. Ця абстрактна модель програми розглядає сукупність інструкцій $\{B\}$ як множину об'єктів, зв'язаних між собою множиною з'єднань $\{U\}$. Правило безпосереднього перероблення псевдо SH-моделі задано цими двома множинами, що не перетинаються. Параметричний опис псевдо SH-моделі такий:

$$M_{ps} : \langle A, Q, q_0, q_f, I, O, G_s \rangle, \quad (4)$$

де $G_s = (B, U)$ – граф, який відображає структуру блок-схеми програми, B – множина об'єктів, U – множина з'єднань між об'єктами.

У визначенні (4) немає параметра P – програми, тому що модель M_{ps} сама є програмою. В перелік характеристик складності псевдо SH-моделі, крім часової, входять також об'єктна і структурна складності. Об'єкт це інструкція програми. Об'єктна складність дорівнює потужності множини об'єктів. Структурна складність оцінює ступінь нерегулярності зв'язків блок-схеми програми. Вона визначається логарифмічною мірою степені нерівномірності матриці інцидентів моделі. Структурна складність є інформаційною характеристикою. Одиницею структурної і об'єктної складності є біт.

Формальні алгоритмічні системи

Клас алгоритмів, в яких правило безпосереднього перероблення задано програмою і засобами виконання інструкцій, отримав назву “Формальні алгоритмічні системи” (ФАС). Моделі ФАС можна поділити на два підкласи. До першого належать моделі ФАС з уявними засобами виконання інструкцій. Ці засоби тільки пояснюють, як може відбуватися обчислювальний процес. Їх наявність,

їх складність не враховується у формальному кількісному описі моделі. Прикладами уявних ФАС є машина Тюрінга, нормальні алгоритми Маркова, система Колмогорова та ін. Формальне визначення моделей алгоритму дається з використанням параметрів. Наприклад, машина Тюрінга визначається шістькою параметрів:

$$M_{mT} : \langle A, Q, a_0, q_f, q_0, P \rangle, \quad (5)$$

де a_0 – позначення порожньої комірки стрічки, $a_0 \in A$; P – програма [5].

В наведеному визначенні немає параметра, який описує наявність апаратних засобів.

За допомогою машини Тюрінга досліджувалися, головним чином, теоретичні проблеми, зокрема, проблеми розв'язності. Машина Тюрінга була першою сходинкою на шляху побудови сучасних комп'ютерів. На основі її досліджень були розроблені принципи побудови універсальної машини, започаткована теорія складності обчислень. В перелік характеристик складності машина Тюрінга входять часова, ємнісна складність та кількість команд програми. Досліджені способи мінімізації часової складності задекларовані у визначенні параметра. Але для розв'язання звичайних задач машина Тюрінга непридатна – через принципово малу продуктивність.

Апаратно-програмна модель алгоритму

До другого класу ФАС належить апаратно-програмна модель алгоритму. У визначенні цієї моделі у явній формі зафіксована наявність апаратних і програмних засобів виконання інструкцій. Правило безпосереднього перероблення задано апаратно-програмними засобами:

$$W = (G, P), \quad (6)$$

де G – конфігурація апаратних засобів,

$$G = (X, U), \quad (7)$$

де X – множина елементарних перетворювачів, U – множина з'єднань елементарних перетворювачів; P – програма.

Модель апаратно-програмної ФАС має назву “SH-модель алгоритму” (S – Software, H – Hardware). Параметричний опис SH-моделі [6]:

$$M_{SH} : \langle A, Q, a_0, q_f, q_0, I, O, G, P \rangle. \quad (8)$$

Інтерпретація правила безпосереднього

перероблення алгоритму, яке об'єднує апаратні та програмні засоби, ґрунтується на аксіомах теорії комп'ютерних алгоритмів [7]:

Аксіома I. *Алгоритми можуть бути реалізовані апаратними засобами.*

Аксіома II. *Алгоритми можуть бути реалізовані апаратно-програмними засобами.*

Аксіома III. *Алгоритми не можуть бути реалізовані лише програмними засобами.*

Описаний підхід до інтерпретації правила безпосереднього перероблення розширює список параметрів алгоритму. Збільшення кількості параметрів SH-моделі створює умови для збільшення списку характеристик складності. До списку входять: часова, апаратна (об'єктна), ємнісна, програмна та структурна характеристики складності. Останні дві характеристики дозволяють оцінити вклад розробника в об'єкт проектування. Апаратно-програмні засоби є інформаційним об'єктом. Одиницею програмної та структурної характеристик складності є біт. Всі перераховані характеристики взаємозалежні. Це дає змогу проводити оптимізацію по декількох характеристиках одночасно, у тому числі оцінювати винахідливість розробника апаратних та програмних засобів.

Припустимо, що потрібно розробити спеціалізований процесор із мінімізованою часовою складністю і мінімізованим часом розроблення. Мінімізована часова складність забезпечує високу продуктивність, мінімізована програмна складність – зменшення втрат часу на проектування. Для даного прикладу процес оптимізації SH-моделі алгоритму досягається зменшенням часової і програмної складності за рахунок збільшення апаратної, ємнісної та структурної складності. Вартісний еквівалент останніх трьох характеристик складності суттєво менший, ніж часової та програмної складності [8].

Використання параметричних моделей

Розглянуті параметричні моделі алгоритмів можуть бути використані в процесі аналізу і розроблення як універсальних, так і спеціалізованих комп'ютерних систем. Для універсальних систем, які будуються з наслідуванням основних архітектурних особливостей попередніх моделей комп'ютерів, корисною є порівняльна оцінка їх складності. За-

вдяки архітектурному принципу “наслідування” оптимізація апаратно-програмних засобів при проектуванні має обмежений характер. Вона торкається лише окремих завдань удосконалення системи, не створює єдиної стратегії розбудови системи з самого початку – заважає необхідність збереження рішень попередніх моделей.

Інша справа із спеціалізованими засобами. Основною метою розроблення потужних спеціалізованих комп'ютерних систем в умовах конкуренції з універсальними системами є впровадження новіших досягнень в області математики, комп'ютерних наук та технологій. Це сприяє досягненню надвисокої продуктивності обчислень на обмеженому колі завдань і мінімальних втрат часу на проектуванні. Специфікою таких завдань, яка сприяє розробці продуктивних спеціалізованих систем, є векторна організація даних, можливість поряд з арифметичними операціями обчислювати фіксовану кількість елементарних та спеціальних функцій. Використання параметричних моделей дозволяє, користуючись особливостями задач, узгоджено вирішити обидві проблеми. Параметричні моделі алгоритму при проектуванні спеціалізованих систем утворюють єдиний ланцюг. Для кожного ієрархічного рівня підбирається відповідна параметрична модель. Параметричні моделі різних ієрархічних рівнів зв'язані між собою змістом правил безпосереднього перероблення і потребують узгодження за характеристиками складності в процесі проектування.

Проектування спеціалізованих систем з використанням параметричних моделей потрібно починати з верхнього ієрархічного рівня. Надвисока продуктивність обчислень і мінімальні втрати часу проектування досягаються оптимізацією характеристик складності SH-моделі. Досягнення високої продуктивності забезпечується мінімізацією часової складності. Способами її мінімізації крім конвеєризації та просторового паралелізму є апаратне виконання операцій, використання розподіленої системи пам'яті, заміна пересилань операціями комутації вузлів пам'яті, мінімізація критичних шляхів розповсюдження сигналів, обчислення кожної елемен-

тарної та спеціальної функції за одною командою.

Мінімізація часу проектування досягається мінімізацією програмної складності, а саме: апаратним виконанням операцій, використанням розподіленої системи пам'яті, заміною пересилань операціями комутації вузлів пам'яті, обчисленням кожної елементарної та спеціальної функції за одною командою.

Відзначимо, що способи мінімізації програмної та часової складності не суперечать один одному. В той же час суттєво збільшуються апаратна, ємнісна і структурна характеристики складності. На сьогодні вартісна вага часової та програмної складності суттєво перевищує вартісну вагу апаратної, ємнісної, структурної характеристик складності.

Практична реалізація побудови спеціалізованої комп'ютерної системи з залученням параметричних моделей алгоритму може бути такою. Проводиться аналіз всієї сукупності процедур розв'язання задач за часовою складністю, обирається процедура з найбільшою часовою складністю. Для неї будується головна структура SH-моделі з мінімальними часовою та програмною складностями. Аналізується можливість суміщення інших процедур оброблення даних з застосуванням вже обраної структури. Якщо структура, отримана в результаті суміщення, не задовольняє допустимим значенням характеристик складності, то суперечливі SH-моделі відокремлюються від головної. Вони утворюють одну чи декілька гілок паралельних SH-моделей. Розділення сукупності SH-моделей на паралельні гілки доцільно проводити за часовою складністю. Утворюються групи за асимптотичною складністю $O(N)$, $O(N \log_2 N)$, $O(N^2)$. Така структура задовольняє вимогам мінімізації часової та програмної складностей, але потребує збільшення, головним чином, апаратної складності.

Наступним об'єктом розроблення є процесорні елементи. На цьому ієрархічному рівні системи слід використовувати H-модель (H-модель реалізує лише апаратні засоби). Мінімізується часова складність, програмна складність дорівнює нулю, що є наслідком апаратного обчислення алгоритму. Крім апаратної, зростає також структур-

на складність обчислювача і операційних пристроїв, що утворюють його.

На останньому етапі проектування обчислювальних засобів проводиться конструювання ефективних абстрактних алгоритмів з метою синтезу операційних пристроїв і корекції структур H-моделей.

Вузол керування спеціалізованої комп'ютерної системи призначений для забезпечення реконфігурації системи. Реконфігурація дає змогу настроювати систему на потрібний граф розв'язання задачі.

Описану методику проектування на основі параметричних моделей алгоритмів впроваджено в процесі розробки спецпроцесорів "СКІФ" [9], а отримані наукові результати апробовані на Міжнародній конференції IDAACS'09 [10].

Висновки

Використання параметричної моделі уможливорює розв'язання проблеми різноманітності тлумачення поняття "алгоритм".

Початкова параметрична модель є основою для конструювання та дослідження нових і вже існуючих моделей алгоритмів.

Параметричний аналіз правила безпосереднього перероблення дає змогу розширити список характеристик складності і підвищити ефективність оптимізації алгоритмів при проектуванні спеціалізованих комп'ютерних систем.

Описаний підхід до аналізу моделей алгоритмів відкриває нові шляхи широкого впровадження математичних методів в побудову перспективних комп'ютерних систем.

Список використаної літератури

1. Cherkaskyy M. Algorithm Conception Clarification Based on the SH-model / Cherkaskyy M., Sachenko A., Osolinskiy O. // The 4th IEEE Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications. 6-8 September 2007, Dortmund, Germany. – 2007. – P. 300-302.
2. Математическая энциклопедия. Гл. ред. И.М. Виноградов. – М.: Советская Энциклопедия, 1977.
3. Кормен Т. Алгоритмы: построение и анализ, 2-е изд. / Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К. Пер.с англ. – М.: Изд. дом "Вильямс", 2005. – 1296 с.

4. Черкаський МПсевдо SH-модель / Черкаський М., Абдалла Саїд Садек //Комп'ютерні системи та мережі. Вісник НУ "Львівська політехніка" – Львів: – 2004. – № 523. – С. 145–150.

5. Успенский В.А. Теория алгоритмов: основные открытия и приложения / Успенский В.А., Семенов А.Л. // Гл. ред. физ.-мат. лит. – М.: Наука, 1987. – 288 с.

6. Черкаський М.В. Взаємозалежність характеристик складності і надійності обчислювальних систем / Черкаський М.В. //Радіоелектронні комп'ютерні системи. Науково-технічний журнал. Харківський авіаційний інститут. – Харків: – 2008. – № 6(33) – С. 46–51.

7. Cherkaskyy M.V. Computer algorithms: axioms and consequences / Cherkaskyy M.V. //Advanced Computer Systems and Networks: Design and Application. Proceedings of the 1st International Conference ACSN – 2005. September 21-23, 2005. – Lviv: – Ukraine. Publishing House of Lviv Polytechnic National University, 2005. – 147 p.

8. Черкаський Н., Саченко А. Технологія проектування на основі теорії складності комп'ютерних алгоритмів / Черкаський Н., Саченко А. // Вестник БрГТУ, серія Фізика, математика, інформатика. – Белоруссия Брест: – № 5. – 2008. – С. 20–23.

9. Черкаський М.В. Перші вітчизняні спецпроцесори / Черкаський М.В // Наук.вісник Чернівецького ун-ту: Збір.наук.праць.: Фізика. Електроніка: Тематич.вип. "Комп'ютерні системи та компоненти". – Чернівці: ЧНУ. – Вип.423. – Ч.1. – 2008. – С.9–13.

10. Cherkaskyy M., Sachenko A. Parametrical Model of Algorithm. Proceedings of IEEE International Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, Rende, Italy: 21-23 September. – 2009, – P. 355–358.

Отримано 20.04.2011

References

1. Mykola Cherkaskyy, Anatoly Sachenko, Oleksandr Osolinskiy. Algorithm Conception Clarification Based on the SH-model // The 4th IEEE Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Appli-

cations. 6-8 September 2007, Dortmund, Germany. – 2007. – P. 300–302 [in English].

2. Mathematical Encyclopedia. Ch. Ed. IM Vinogradov. – Moscow: Soviet Encyclopedia, 1977. – In 5 volumes [in Russian].

3. Stern T., Leiserson C., Rivest R., Stein, K. Algorithms: the construction and analysis of analogous, 2nd edition.: Per.s English. –: Izd. house "Williams", 2005. – 1296 p. [in Russian].

4. Mykola Cherkaskyy, Abdullah Saeed Sadeq. Pseudo SH-model // Computer Systems and Networks. Bulletin NU "Lviv Polytechnic" – Lviv, 2004. – № 523. – P.145–150 [in Ukrainian].

5. Uspensky V.A., Semenov A.L. The theory of algorithms: key discoveries and applications. – Moscow: Nauka Ch. Ed. Sci. Lit., 1987. – 288 p. [in Russian].

6. Cherkaskyy M. Interdependence characteristics of complexity and reliability of computer systems. / Radio electronic computer system. Science and technology magazine. Kharkov Aviation Institute. – Kharkov: – 2008. – № 6 (33). – P.46–51 [in Ukrainian].

7. Cherkaskyy M.V. Computer algorithms: axioms and consequences /Advanced Computer Systems and Networks: Design and Application. Proceedings of the 1st International Conference ACSN. – Lviv: Publishing House of Lviv Polytechnic National University, 2005 – 147 p. [in English].

8. Cherkaskyy N., A. Sachenko technology design based on the theory of computer algorithms / Belarus, Brest, Journal BrGTU, Series Physics, Mathematics, Computer Science. – № 5. – 2008. – P. 20–23 [in Russian].

9. Cherkaskyy M. Pershi vitchiznyani spetsprotessorori // News of Chernivetskogo Naukova universitetu: Zbirnik Sciences. Pratzhen. Vip.423: Fizika. Elektronika: thematically vipusk "Komp'terni of the component." Chastina 1.– Chernivtsi: CHNU. – 2008. – P.9–13 [in Ukrainian].

10. Mykola Cherkaskyy, Anatoly Sachenko. Parametrical Model of Algorithm. Proceedings of IEEE International Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications. – 2009, Rende, Italy. – P. 355–358 [in English].



Черкаський Микола В'ячеславович,
д.т.н., проф.каф. ЕОМ Нац.
ун-ту "Львівська політехніка"



Саченко Анатолій Олексійович,
д.т.н., зав.каф. ІОСУ Тернопільського
нац. економ. ун-ту