

УДК 519.684.4

ПАРАЛЕЛІЗМ АЛГОРИТМУ СИНТЕЗУ МОДЕЛЕЙ ОПТИМАЛЬНОЇ СКЛАДНОСТІ НА ЗАСАДАХ ГЕНЕТИЧНИХ АЛГОРИТМІВ

М. І. Горбійчук

Доктор технічних наук, професор*

E-mail: ksm@nung.edu.ua

В. М. Медведчук

Аспірант*

E-mail: ksm@nung.edu.ua

Б. В. Пашковський

Аспірант*

E-mail: ksm@nung.edu.ua

*Кафедра комп'ютерних систем та мереж
Івано-Франківський національний технічний

університет нафти і газу

вул. Карпатська, 15,

м. Івано-Франківськ, Україна, 76018

Для складних технічних об'єктів з великим числом вхідних змінних затрати машинного часу на побудову математичної моделі є досить помітними. Проведений аналіз методу побудови емпіричних моделей оптимальної складності на засадах генетичних алгоритмів. Показано, що такий алгоритм має внутрішній паралелізм, що дає змогу розробити ефективну програму реалізації, що приведе до скорочення затрат машинного часу

Ключові слова: емпірична модель, внутрішній паралелізм, граф алгоритму, ширина ярусу, паралельна структура

Для сложных технических объектов с большим числом входных переменных затраты машинного времени на построение математической модели достаточно заметны. Проведен анализ метода построения эмпирических моделей оптимальной сложности на основе генетических алгоритмов. Показано, что такой алгоритм имеет внутренний параллелизм, что позволяет разработать эффективную программу реализации, что приведет к сокращению затрат машинного времени

Ключевые слова: эмпирическая модель, внутренний параллелизм, граф алгоритма, ширина яруса, параллельная структура

1. Вступ

Якщо є вичерпні дані про певну систему (об'єкт), то можна отримати модель, яка буде адекватно відображати поведінку такої системи при зроблених допущеннях і при відомих параметрах середовища, з яким взаємодіє система. Такі моделі дістали назву детермінованих і використовуються для розв'язання цілого ряду задач у різних областях людської діяльності.

У більшості випадків системи (об'єкти) є досить складними. Складність їх полягає у наявності багатьох компонентів, які складним чином взаємодіють між собою. Прикладом таких систем можуть бути як технічні, так і екологічні об'єкти [1, 2]. У таких випадках для опису взаємодії між параметрами системи і зовнішнім середовищем, яке безпосередньо діє на систему і викликає зміну її стану, використовують емпіричне моделювання. Отримані у такий спосіб математичні моделі широко використовують для розв'язання таких задач як розпізнавання образів, прогнозування, автоматична класифікація, оптимальне керування та інші [3].

Тому актуальним є виявлення паралелізму у алгоритмі синтезу моделей оптимальної складності на засадах генетичних алгоритмів для створення відповідного програмного забезпечення, що дасть змогу скоротити затрати машинного часу у процесі обчислень.

2. Аналіз літературних даних

В основі емпіричного моделювання багатьох явищ і процесів лежить широко відомий метод найменших квадратів (МНК), започаткований більше ніж 200 років тому назад молодим німецьким математиком Карлом Фрідріхом Гаусом.

МНК передбачає, що відома структура моделі і необхідно за спостереженнями як за вхідними, так і за вихідною величинами побудувати модель, яка найкращим чином апроксимувала би емпіричні дані. Як критерій наближення вибирають суму квадратів відхилень експериментальних даних від розрахункових, що отримані за емпіричною моделлю.

Критерій наближення МНК є внутрішнім критерієм [4] і його застосування приводить до помилкового висновку: чим складніша модель, тим вона точніша. Оскільки на вихід системи накладається перешкода (допускають, що вона адитивна), то надмірна точність моделі може значно спотворити об'єктивно існуючу функціональну залежність між виходом системи і її входами.

Тому для вибору структури емпіричної моделі акад. О. Г. Івахненком був запропонований метод, який дістав назву індуктивний метод самоорганізації моделей [5], ідейну сторону якого визначає теорема Геделя. Ця теорема стверджує, що ніяка система аксіом не може бути логічно замкнутою: завжди знайдеться така теорема, для доведення якої необхідне зовнішнє

доповнення – розширення початкової системи аксіом. Стосовно задачі синтезу моделі геделівський підхід означає застосування зовнішнього критерію для однозначного вибору структури моделі із заданого класу моделей. Критерій називають зовнішнім, якщо його обчислення ґрунтується на даних, які не використовувались при розв'язанні задачі МНК.

Задача синтезу емпіричних моделей з використанням індуктивного методу самоорганізації моделей відноситься до класу задач великої розмірності. Наприклад, якщо вибрана поліноміальна модель, яка має 7 аргументів і п'яту степінь полінома, то загальне число моделей-претендентів, серед яких слід вибрати модель оптимальної складності складе [6] $\sim 2,6 \cdot 10^{238}$. Зрозуміло, що реалізувати такий обсяг обчислень навіть на найновіших ЕОМ неможливо.

Для зниження розмірності задачі авторами роботи [6] запропонований метод синтезу емпіричних моделей на засадах генетичних алгоритмів, суть якого буде подана нижче. Це дало змогу реалізувати відповідний алгоритм у якому число вхідних змінних не перевищує семи, а степінь полінома не більше п'яти. При цьому затрати машинного часу вдалось радикально зменшити, але вони залишаються ще досить помітними. Одним із ефективних шляхів зменшення таких затрат є застосування паралельних обчислень, до яких на сьогоднішній день повернута значна увага як вітчизняних, так і зарубіжних дослідників [3, 7–9].

3. Мета та задачі дослідження

Метою дослідження є виявлення паралелізму у алгоритмі синтезу моделей оптимальної складності на засадах генетичних алгоритмів для створення відповідного програмного забезпечення, що дасть змогу скоротити затрати машинного часу у процесі обчислень

Задачами дослідження є:

- аналіз розробленого алгоритму синтезу емпіричних моделей оптимальної структури на засадах генетичних алгоритмів для виявлення його внутрішнього паралелізму;

- визначення можливості реалізації такого алгоритму на різних обчислювальних системах і аналіз затрат машинного часу при такій реалізації.

4. Дослідження алгоритму синтезу моделей оптимальної складності на засадах генетичних алгоритмів на наявність внутрішнього паралелізму

Реалізація індуктивного методу самоорганізації моделей здійснюється поетапно: перший етап генерування моделей-претендентів (у певному порядку підвищення їх складності); другий вибір найкращої моделі за мінімумом одного із критеріїв селекції. Як критерій селекції вибирають [5]: критерій регулярності

$$\Delta^2(B) = \frac{\sum_{i=1}^{N_B} (Y^{(i)}(B) - y^{(i)}(B))^2}{\sum_{i=1}^{N_B} Y^{(i)}(B)^2} \quad (1)$$

і зміщення

$$\Delta^2(A, B) = \frac{\sum_{i=1}^N (y^{(i)}(A) - y^{(i)}(B))^2}{\sum_{i=1}^N (Y^{(i)})^2} \quad (2)$$

Індуктивний метод самоорганізації моделей передбачає, що всі дані, отримані у результаті експерименту, розбиваються на дві частини – навчальну N_A і перевірну N_B . Тоді у формулах (1) і (2) $y^{(i)}(A)$, $y^{(i)}(B)$ – значення виходу моделі, що обчислені відповідно на множинах експериментальних значень N_A і N_B . Якщо вибраний критерій регулярності (1), то витирають такий розподіл даних експерименту [10]: $N_A = 0,7N$ і $N_B = 0,3N$, а при виборі критерію (2) – $N_A = 0,5N$ і $N_B = 0,5N$.

Найчастіше при побудові емпіричних моделей функціональну залежність між виходом об'єкта і його входами обирають у такому вигляді:

$$y = \sum_{k=0}^r a_k f_k(\bar{x}), \quad (3)$$

де a_k , $k = \overline{1, r}$ – параметри моделі; $\bar{x} = (x_1, x_2, \dots, x_n)^T$ – вектор вхідних величин.

Розрізняють три способи генерування моделей-претендентів. Перший із них комбінаторний [5], за яким вибір моделей-претендентів здійснюється шляхом прирівнювання до нуля деяких коефіцієнтів у виразі (3), що дає змогу отримати сукупність моделей. Вибір найкращої із них здійснюється на основі одного із критеріїв селекції (2) або (3). Другий спосіб відомий як метод групового врахування аргументів (МГУА) [11], у якому генерування моделей-претендентів здійснюється за допомогою багаторядної процедури. У першому ряду селекції комбінують всі можливі пари аргументів (вхідних величин), і для кожної із них утворюють моделі, наприклад, у вигляді повного полінома. Із всіх часткових моделей, утворених у такий спосіб, вибирають K найкращих за одним із критеріїв селекції. Із виходів цих K моделей знову утворюють комбінації всіх можливих пар, які є входами другого ряду селекції. Для кожної із цих пар знову формують часткові моделі, із яких вибирають K найкращих. Процес нарощування рядів селекції відбувається до тих пір поки вибрий критерій селекції зменшує своє значення. Оцінка параметрів часткових моделей здійснюється за допомогою МНК. Третій метод [10] подібний до другого. Різниця лише у тому, що на кожному ряді селекції часткові моделі утворюють шляхом прирівнювання до нуля певного числа їх коефіцієнтів.

Недоліком комбінаторного методу селекції моделей є необхідність великого числа перебору часткових моделей. Якщо початковою моделлю вибраний повний поліном степені m , то загальне число моделей претендентів буде $2^M - 1$, де M – загальне число членів початкового полінома. Навіть сучасні ЕОМ не здатні реалізувати комбінаторний алгоритм селекції моделей при значній кількості вхідних величин і при високій степені початкового полінома. МГУА синтезує моделі, в яких фігурують проміжні змінні, що значно утруднює процес переходу до вхідних змінних системи.

Сказане відноситься і до третього методу, оскільки він по суті є модифікацією МГУА.

Як емпіричну модель виберемо поліном ступіня

$$y = \sum_{i=0}^{M-1} a_i \prod_{j=1}^{s_{ji}} x_j^{s_{ji}}, \tag{4}$$

де a_i – коефіцієнти полінома; s_{ji} – ступіні аргументів, які повинні задовольняти обмеженню:

$$\sum_{j=1}^n s_{ji} \leq m.$$

Число членів M полінома (4) визначають за такою формулою [12]:

$$M = \frac{(m+n)!}{m!n!}. \tag{5}$$

Метод найменших квадратів передбачає, що структура моделі (4) відома. Тоді задача полягає у тому, щоб на основі спостережень за вхідними і вихідною величинами, визначити параметри a_i , $i=0, M-1$ моделі (4), таким чином, щоб якомога точніше наблизити вихід системи до виходу моделі. За критерій такого наближення вибирають суму квадратів відхилень розрахункових $y^{(i)}$ від експериментальних значень $Y^{(i)}$, $i=1, N$. При відомій матриці спостережень

$$X = \begin{bmatrix} x_1^{(1)} & x_2^{(1)} & \dots & x_n^{(1)} \\ x_1^{(2)} & x_2^{(2)} & \dots & x_n^{(2)} \\ \dots & \dots & \dots & \dots \\ x_1^{(N)} & x_2^{(N)} & \dots & x_n^{(N)} \end{bmatrix}$$

і вибраних параметрах моделі (1) можна обчислити її вихід

$$\bar{y} = F\bar{a}, \tag{6}$$

$$\text{де } F = \begin{bmatrix} \prod_{j=0}^n (x_j^{(1)})^{s_{j0}} & \prod_{j=0}^n (x_j^{(1)})^{s_{j1}} & \dots & \prod_{j=0}^n (x_j^{(1)})^{s_{jM-1}} \\ \prod_{j=0}^n (x_j^{(2)})^{s_{j0}} & \prod_{j=0}^n (x_j^{(2)})^{s_{j1}} & \dots & \prod_{j=0}^n (x_j^{(2)})^{s_{jM-1}} \\ \dots & \dots & \dots & \dots \\ \prod_{j=0}^n (x_j^{(N)})^{s_{j0}} & \prod_{j=0}^n (x_j^{(N)})^{s_{j1}} & \dots & \prod_{j=0}^n (x_j^{(N)})^{s_{jM-1}} \end{bmatrix};$$

$\bar{a} = (a_0, a_1, \dots, a_{M-1})$ – вектор параметрів моделі (4);

$\bar{y} = (y^{(1)}, y^{(2)}, \dots, y^{(N)})$ – обчислені значення моделі (4) у кожній точці спостережень.

За відомими значеннями $Y^{(i)}$ та $y^{(i)}$ обчислити критерій наближення (апроксимації)

$$J(\bar{a}) = \sum_{i=1}^N (Y^{(i)} - y^{(i)})^2,$$

який подамо як скалярний добуток двох векторів

$$J(\bar{a}) = (\bar{Y} - \bar{y})^T (\bar{Y} - \bar{y}).$$

Враховуючи значення \bar{y} , яке визначається формулою (6), отримуємо

$$J(\bar{a}) = (\bar{Y} - F\bar{a})^T (\bar{Y} - F\bar{a}). \tag{7}$$

Значення параметрів моделі (4) обчислюють із умови, що критерій наближення (7) набуває мінімального значення відносно вектора параметрів \bar{a} . Мінімізація (7) приводить до такого результату:

$$M_F \bar{a} = F^T \bar{Y}, \tag{8}$$

де $M_F = F^T F$ – матриця Фішера.

У тому випадку, коли розмірність вектора \bar{a} невелика і матриця M добре обумовлена, із рівняння (8) можна безпосередньо визначити

$$\bar{a} = M_F^{-1} F^T \bar{Y}.$$

При комбінаторному методі синтезу моделей оптимальної складності із повного полінома (4) отримують модель, де частина коефіцієнтів a_i , $i=1, M$ моделі (4) набувають нульових значень. Оскільки задача синтезу емпіричних моделей оптимальної складності з використанням комбінаторного методу має велику розмірність, то для зняття проблеми великої розмірності комбінаторного методу застосуємо генетичний підхід. Утворимо впорядковану послідовність, де на i -тому місці буде знаходитись одиниця або нуль в залежності від того чи параметр a_i , $i=1, M$ моделі (4) буде відмінним від нуля або матиме значення нуль. У теорії генетичних алгоритмів така упорядкована послідовність носить назву хромосоми, а атомарний елемент (одиниця або нуль) хромосоми – це ген. Набір хромосом утворює популяцію. Важливим поняттям у теорії генетичних алгоритмів є функція пристосування, яка визначає ступінь пристосованості окремих особин у популяції. Вона дає змогу із усієї популяції вибрати ті особини, що є найбільш пристосованими, тобто ті яким відповідає найменше (найбільше) значення функції пристосування. У задачі синтезу моделей оптимальної складності як функцію пристосування виберемо критерій селекції (1) або (2).

Як і класичний генетичний алгоритм [13], алгоритм синтезу моделей оптимальної складності, що ґрунтується на засадах генетичних алгоритмів, складається із таких кроків.

К1. Формування початкової популяції (ініціалізація). На першому кроці роботи алгоритму випадковим чином формується популяція із I особин, кожна із яких є хромосомою довжиною M . Число генів у хромосомі визначається за формулою (5).

К2. Оцінка пристосованості хромосоми у популяції. Для кожної хромосоми обчислюється значення критерію селекції (1) або (2) наступним чином. Якщо вибраний критерій селекції (1), то формуються матриці F_A і F_B розмірами $N_A \times M$ і $N_B \times M$. Із матриці F_A вилучається i -тий стовпець, якщо на i -тій позиції хромосоми знаходиться нуль; якщо одиниця, то відповідний стовпець залишається без змін. У результаті отримаємо матрицю \tilde{F}_A , із якої вилучено c стовпців (за кількістю нулів у хромосомі). Розмір такої матриці буде $N_A \times (M - c)$. Аналогічно отримаємо матрицю \tilde{F}_B розміром $N_B \times (M - c)$. На множині точок N_A обчислюються ненульові коефіцієнти a_{Aj} , $j=1, M-c$ моделі (4) шляхом розв'язання нормального рівняння Гауса, яке видозмінюється наступним чином:

$$\tilde{M}_{F,A} \bar{a}_A = \tilde{F}_A^T \bar{Y}_A, \quad (9)$$

де $\bar{a}_A = (a_{A0}, a_{A1}, \dots, a_{A_{M-c-1}})^T$ – вектор ненульових параметрів моделі, який асоційований з черговою хромосомою; $\tilde{M}_{F,A} = \tilde{F}_A^T \tilde{F}_A$. $\bar{Y}_A = (Y^{(1)}, Y^{(2)}, \dots, Y^{(N_A)})$ – вектор експериментальних значень на множині N_A .

За знайденими коефіцієнтами \bar{a}_A поліноміальної моделі на множині точок N_B обчислюють значення

$$\bar{y}(B) = \tilde{F}_B \bar{a}_A. \quad (10)$$

Знаючи $\bar{y}(B)$, за формулою (1) обчислюють значення функції пристосування $\Delta_j^2(B)$, $j=1, \bar{I}$ для кожної хромосоми із початкової популяції.

У тому випадку, коли як функцію пристосування використовують критерій зміщення (2), тоді складають рівняння (9), яке розв'язують методом Гауса відносно параметрів \bar{a}_A . Після цього обчислюють $\bar{y}(A) = \tilde{F}_A \bar{a}_A$ за формулою (10). Отримані значення $\bar{y}(A)$ і $\bar{y}(B)$ дають змогу знайти значення $\Delta_j^2(A, B)$, $j=1, \bar{I}$ для кожної хромосоми із популяції.

К3. Перевірка умови зупинки алгоритму.

Визначають

$$\Delta_m^2(B) = \min_j \Delta_j^2(B), \quad (11)$$

$$\Delta_m^2(A, B) = \min_j \Delta_j^2(A, B). \quad (12)$$

Якщо мінімальне значення (11) або (12) критерію селекції (1) або (2) не перевищує деякого заданого значення E , то відбувається зупинка обчислень. Зупинка обчислень також може відбутись у випадку, коли у результаті виконання алгоритму немає суттєвого зменшення функції пристосування або у тому випадку, коли виконано задане число ітерацій.

Після виконання однієї із трьох умов із чергової популяції вибирається та хромосома ch^* , для якої виконується умова (11) або (12). Ця хромосома задає структуру моделі оптимальної складності і формує матрицю F^* таким чином, що із початкової матриці вилучаються стовпці, які асоційовані з нульовими значеннями відповідних генів. Перерахунок параметрів моделі (4) здійснюється на всій множині точок N за допомогою МНК.

К4. Селекція хромосом. За розрахованими на другому кроці алгоритму значеннями функції пристосування здійснюється відбір тих хромосом, які будуть брати участь у створенні потомків для нової популяції. Такий відбір проводиться за принципом природного відбору, коли найбільші шанси у створенні нової популяції мають хромосоми з найкращими значеннями функції пристосування (1) або (2). Найпоширенішим методами селекції є метод рулетки і турнірний метод [13]. Метод рулетки можна застосовувати тоді, коли функція пристосованості додатна, що робить його придатним лише для задач максимізації. Турнірний метод можна використовувати як у задачах максимізації, так і у задачах мінімізації.

При турнірній селекції всі хромосоми розбиваються на підгрупи з наступним вибором із кожної підгрупи хромосоми з найкращою пристосованістю. Підгрупи можуть мати довільний розмір, але частіше за все популяцію ділять на підгрупи по 2–3 особи у кожній.

К5. Формування нової популяції потомків. Над відібраними особинами алгоритму здійснюють відомі зміни за допомогою двох основних операторів: схрещування і мутації. Слід зауважити, що оператор мутації застосовується рідше у порівнянні з оператором схрещування. Вірогідність схрещування досить висока ($0 \leq P_c \leq 1$). Тоді як вірогідність мутації складає $0 \leq P_m \leq 0,1$. Ймовірність мутації P_m можна задати шляхом вибору випадкового числа із інтервалу $[0; 1]$ для кожного гена. Мутації підлягають ті гени (шляхом заміни одиниці на нуль і навпаки), для яких розігране число виявиться меншим або рівним P_m . Мутація може здійснюватись як над пулом родичів, так і над пулом потомків. Оператор схрещування застосовується до пулу потомків у такий спосіб. Із утвореної популяції особин випадковим чином вибирається пара хромосом і генерується випадкове число P_z із інтервалу $[0; 1]$. Якщо виконується умова $P_z \leq P_c$, то над парою хромосом здійснюється схрещування. У протилежному випадку пара хромосом залишається без змін. Для пари хромосом, які підлягають схрещуванню, випадковим чином розігрується позиція гена (локус), яка визначає точку схрещування. Оскільки кожна хромосома має M генів, то точка схрещування L_c – це натуральне число менше M . Тому фіксація точки схрещування зводиться до вибору натурального випадкового числа із інтервалу $[1; L_c - 1]$. Дія оператора схрещування приводить до того, що із пари родичів утворюється нова пара потомків: на позиціях від $L_c + 1$ до M хромосоми у парі обмінюються своїми генами.

К6. Після виконання оператора схрещування відбувається перехід до кроку **К2**.

Реалізація алгоритму синтезу моделей оптимальної складності показала [6], що він дає значний вигравш у часі у порівнянні з комбінаторним методом селекції моделей. Але із збільшенням розмірності задачі синтезу емпіричних моделей зростають затрати машинного часу. Для зменшення таких затрат необхідно вивчити внутрішню структуру алгоритму з метою виявлення його внутрішнього паралелізму.

Внутрішній паралелізм є властивістю алгоритму і не залежить від його реалізації на конкретній ЕОМ.

Виявити внутрішній паралелізм алгоритму можна, побудувавши орієнтований граф [3]. Вершини такого графа можуть бути як прості математичні операції (додавання, віднімання, множення та ділення), так і сукупності операцій, які оформлені у вигляді функцій або підпрограм [14]. На множині вершин операцій певним чином визначений порядок виконання операцій, що дає змогу задати дуги графа. Якщо операція, яка відповідає вершині β , використовує хоча би один аргумент як результат виконання операції відповідної вершини α , то вершини α і β з'єднані орієнтованою дугою. У протилежному випадку дуга між вершинами α і β не проводиться. Побудований у такий спосіб орієнтований граф носить назву графа алгоритму [3].

Граф алгоритму можна розбити на яруси. У середині кожного ярусу немає зв'язків між вершинами. Виходи кожного ярусу є аргументами для наступного ярусу. Ярусам графа, як правило, присвоюють порядкові номери 1, 2, ..., s . Число вершин в одному ярусі визначає його ширину, а число ярусів у паралельній формі – висоту графа алгоритму.

Слід відзначити, що граф алгоритму є значно простішим ніж сам алгоритм. Після того як побудований граф алгоритму аналізуються відповідні паралельні форми і на основі проведеного дослідження вибирається необхідна схема реалізації алгоритму на обчислювальній системі паралельної архітектури.

На рис. 1 показаний граф алгоритму синтезу емпіричних моделей оптимальної складності, де критерієм селекції є критерій регулярності, що визначається формулою (1). Наведений алгоритм є ітераційним і на рис. 1 показаний граф тільки однієї ітерації. Аналіз графу алгоритму показує, що він має внутрішній паралелізм у межах однієї ітерації. Число паралельних гілок графа визначається кількістю хромосом, які генеруються на першому кроці алгоритму.

Граф алгоритму можна розглядати як розгорнутих у часі послідовність ярусів, число яких визначається умовою зупинки алгоритму. Це число і визначає висоту графа алгоритму.

Вершини кожної гілки графа асоційовані з операціями формування матриці $\tilde{M}_{F,A}$, системи лінійних алгебраїчних рівнянь (9), яку у матрично-векторній формі запишемо так:

$$\tilde{A}\tilde{a}_A = \tilde{b}, \tag{13}$$

де $\tilde{A} = \tilde{M}_{F,A}$; $\tilde{b} = \tilde{F}_A^T \tilde{Y}_A$, а також її розв'язуванням. У кожній вітці графа алгоритму за формулою (10) обчислюється значення величини $y(B)$.

Дослідимо на паралелізм алгоритми розв'язання системи лінійних алгебраїчних рівнянь (13) та обчислення значення величини $y(B)$ за формулою (10).

Розв'язання системи (13) здійснюється у два етапи [15]: перший приведення матриці \tilde{A} до верхнього діагонального вигляду з одиницями на головній діагоналі з одночасним перерахунком вільних членів системи рівнянь; другий етап знаходження розв'язків системи рівнянь (13) методом зворотного ходу.

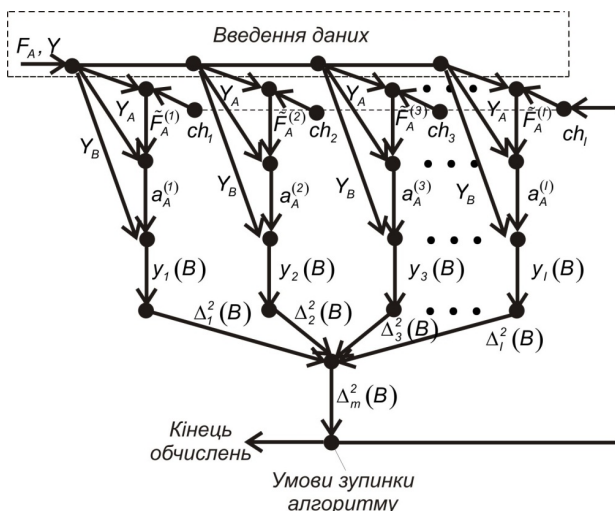


Рис. 1. Граф алгоритму синтезу моделей оптимальної складності

Утворимо розширену матрицю $A_b = [\tilde{A} | \tilde{b}]$ розміром $(M-c) \times (M-c+1)$ і приведемо її до верхнього

діагонального виду, здійснивши такі операції над її елементами [15]:

$$a_{kj}^{(k)} = \frac{a_{kj}^{(k-1)}}{a_{kk}^{(k-1)}}, a_{ij}^{(k)} = a_{ij}^{(k-1)} - a_{ik}^{(k-1)} a_{kj}^{(k)}, i = \overline{k+1, z}, j = \overline{k+1, z+1},$$

$$k = \overline{1, z}, \tag{14}$$

де $z = M - c$.

Результатом перетворення матриці A_b за формулами (14) буде верхня діагональна матриця

$$U = \begin{bmatrix} 1 & u_{12} & u_{13} & \dots & u_{1z} & u_{1,z+1} \\ 0 & 1 & u_{23} & \dots & u_{2z} & u_{2,z+1} \\ 0 & 0 & 1 & \dots & u_{3z} & u_{3,z+1} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 1 & u_{z,z+1} \end{bmatrix}.$$

Зворотний хід, у якому у зворотному порядку визначаються невідомі здійснюється за формулами

$$a_{A,z} = u_{z,z+1}, a_{A,i} = u_{i,z+1} - \sum_{j=i+1}^z u_{ij} a_{A,j}, i = \overline{z-1, 1}. \tag{15}$$

Побудуємо граф алгоритму (14). Вершини графа відповідають операціям типу a/b і $a-cd$. Для того, щоб полегшити аналіз графа алгоритму, вершини графа розмістимо у певному порядку. Розглянемо прямокутну решітку у трьохвимірному просторі з координатами i, j, k всі цілочисельні вузли решітки для $j = \overline{1, z+1}$, $i, k = \overline{1, z}$ помістимо вершини графа. Аналізуючи формули (14), можна прийти до висновку, що у вершину з координатами i, j, k буде передаватися результат виконання операції, який відповідає вершині з координатами $i, j, k-1$. Для $k=2$ граф алгоритму показаний на рис. 2. Паралелізм алгоритму (14) проявляється у тому, що для кожного k перерахунок елементів матриці A_b здійснюється незалежно.

Тепер проаналізуємо алгоритм (15) розв'язування системи лінійних алгебраїчних рівнянь з трикутною матрицею U методом зворотної підстановки. Фізичний зміст задачі синтезу оптимальної структури емпіричної моделі такий, що матриця U є не виродженою.

Формула (15) не визначає алгоритм однозначно, так як не визначений порядок знаходження суми. Розглянемо послідовну операцію взяття суми, що знаходиться у правій частині співвідношення (15). Відповідне рекурентна процедура буде такою:

$$a_{A,z} = u_{z,z+1},$$

$$a_{A,z-i}^{(j)} = a_{A,z-i}^{(j-1)} - u_{z-i,z-j+1} a_{A,z-j+1}^{(j-1)}, a_{A,z-i}^{(0)} = u_{z-i,z+1}, a_{A,z-i} = a_{A,z-i}^{(i)},$$

$$i = \overline{1, z-1}, j = \overline{1, i}. \tag{16}$$

Основна операція алгоритму виду $a-bc$ виконується для всіх значень індексів i, j . Всі інші операції виконують присвоєння. У декартовій системі з осями i, j побудуємо координатну сітку і у її вузлах розмістимо вершини графа, що відповідають операціям $a-bc$ (рис. 3).

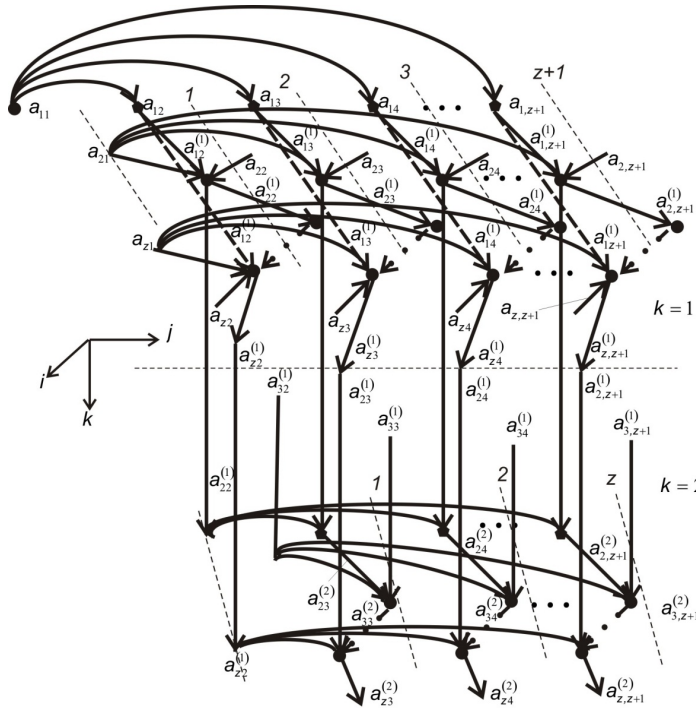


Рис. 2. Граф алгоритму приведення матриці A_B до верхньої трикутної форми

Використовуючи рекурентну процедуру (16), побудуємо граф алгоритму, включивши до нього також вершини, які асоційовані з введенням даних u_{ij} та $u_{i,z+1}$, $i = z-1, 1$. Яруси графа алгоритму відмічені пунктирними лініями. Загальне число ярусів, які вміщують операції типу а-бс, дорівнює $z-1$.

Кожна гілка графа алгоритму синтезу моделей оптимальної складності вміщує операцію множення матриці \tilde{F}_B розміром $N_B \times z$ на вектор \bar{a}_A за формулою (10). У результаті отримуємо вектор $\bar{y}(B)$.

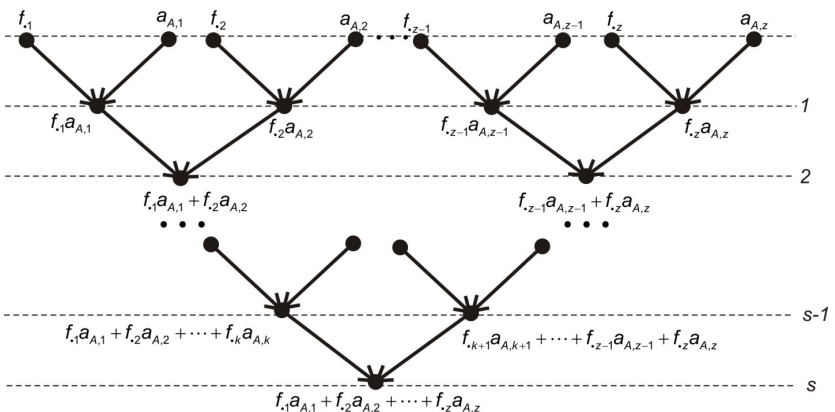


Рис. 4. Граф алгоритму подвоєння

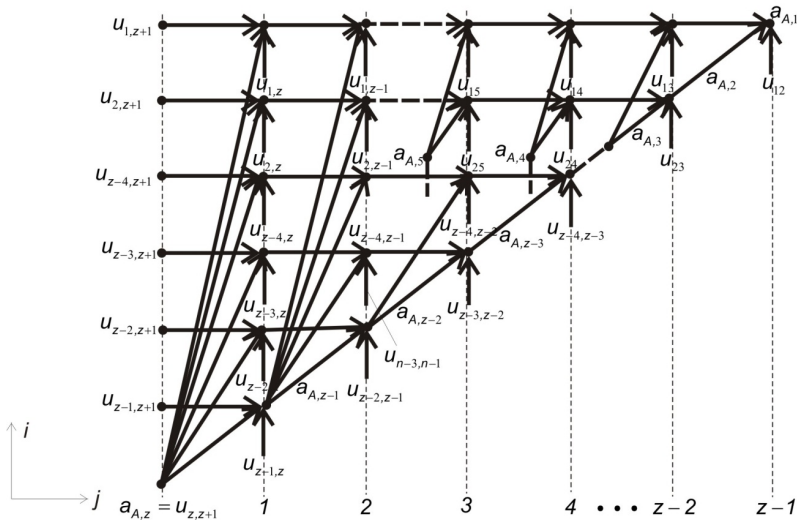


Рис. 3. Граф алгоритму розв'язання системи лінійних алгебраїчних рівнянь

Формулу (10) запишемо в іншому вигляді

$$\bar{y}(B) = \sum_{i=1}^z f_i a_{A,i}, \quad (17)$$

де f_i – i -тий стовпець матриці \tilde{F}_B .

Граф алгоритму, побудований на основі аналізу формули (17), показаний на рис. 4 і він носить назву графа алгоритму подвоєння.

Граф має s ярусів. На першому ярусі виконується бінарна операція множення вектора (стовпця матриці) на число. На решта ярусах маємо бінарні операції додавання. На нульовому ярусі розміщені вершини, які відповідають введенню початкових даних f_i та $a_{A,i}$.

Оскільки на при переході від першого ярусу до кожного наступного ярусу кількість вершин графа зменшується наполовину, то на першому ярусі буде мати $z/2$ вершин (для простоти вважаємо, що z можна подати як степінь числа 2); на другому ярусі буде $z/2^2$; на третьому – $z/2^3$ і т. д. На останньому s -тому ярусі маємо $z/2^s = 1$. З останньої рівності знаходимо висоту алгоритму подвоєння за умови, що z можна подати як ступінь числа 2

$$s = \log_2 z.$$

У загальному випадку значення $z \neq 2^s$. Тоді $s = \lceil \log_2 z \rceil$, де ціла частина числа $\lceil \log_2 z \rceil$ береться з надлишком.

5. Висновки

Аналіз розробленого алгоритму синтезу емпіричних моделей оптимальної структури на засадах генетичних алгоритмів шляхом побудови графа алгоритму показав, що він має внутрішній паралелізм. Число гілок такого графа визначається кількістю хромосом, що генерується на першому кроці алгоритму. У свою чергу, кожна гілка графа алгоритму вміщує вершини які асоційовані з такими операціями як розв'язування системи лінійних алгебраїчних

рівнянь та множення матриці на вектор, які також мають внутрішній паралелізм. Внутрішній паралелізм є характерною властивістю алгоритму і не залежить від обчислювальної системи, на якій реалізується та-

кий алгоритм. Наявність внутрішнього паралелізму в алгоритмі дає змогу реалізувати такий алгоритм на обчислювальній системі з паралельною архітектурою, що призведе до скорочення затрат машинного часу.

Література

1. Горбійчук, М. І. Метод синтезу емпіричних моделей з врахуванням похибок вимірювань [Текст] / М. І. Горбійчук, І. В. Щупак, Т. Осколів // *Методи та прилади контролю якості*. – 2011. – № 2 (27). – С. 67–76.
2. Горбійчук, М. І. Метод побудови математичних моделей складних процесів на засадах генетичних алгоритмів [Текст] / М. І. Горбійчук, М. А. Шуфнарів // *Штучний інтелект*. – 2010. – № 4. – С. 50–57.
3. Воеводин, В. В. Параллельные вычисления [Текст] / В. В. Воеводин, Вл. В. Воеводин. – СПб: БХВ-Петербург, 2002. – 608 с.
4. Ермаков, С. М. Математическая теория оптимального эксперимента [Текст] / С. М. Ермаков, А. А. Жиглявский. – М.: Наука, 1987. – 320 с.
5. Ивахненко, А. Г. Индуктивный метод самоорганизации моделей сложных систем [Текст] / А. Г. Ивахненко. – К.: Наукова думка, 1981. – 296 с.
6. Горбійчук, М. І. Метод синтезу емпіричних моделей на засадах генетичних алгоритмів [Текст] / М. І. Горбійчук, О. Б. Василенко, І. В. Щупак // *Розвідка та розробка нафтових і газових родовищ*. – 2009. – № 4(33). – С. 72–79.
7. Ортега, Дж. Введение в параллельные и векторные методы решения линейных систем [Текст] / Дж. Ортега; пер. с англ. Х. Д. Икрамова, И. Е. Капорина; под ред. Х. Д. Икрамова. – М.: Мир, 1991. – 367 с.
8. Бэбб, Р. Программирование на параллельных вычислительных системах [Текст] / Р. Бэбб, Дж.Мак-Гроу, Т. Акселрод и др.; пер. с англ. А. С. Косачева, Л. В. Шабанова под ред. Ю. Г. Дадаева. – М.: Мир, 1991. – 376 с.
9. Химич, А. Н. Параллельные алгоритмы решения задач вычислительной математики [Текст] / А. Н. Химич, И. Н. Молчанов, А. В. Попов и др. – К.: Наукова думка, 2008. – 248 с.
10. Ивахненко, А. Г. Справочник по типовым программам моделирования [Текст] / А. Г. Ивахненко, Ю. В. Коппа, В. С. Степашко и др.; под ред. А. Г. Ивахненко. – К.: Техніка, 1980. – 180 с.
11. Ивахненко, О. Г. Передбачення випадкових процесів [Текст] / О. Г. Іваненко, В. Г. Лапа. – К.: Наукова думка, 1969. – 420 с.
12. Горбійчук, М. І. Индуктивный метод побудови математичних моделей газоперекачувальних агрегатів природного газу [Текст] / М. І. Горбійчук, М. І. Когутяк, Я. І. Заячук // *Нафтова і газова промисловість*. – 2008. – № 5. – С. 32–35.
13. Рутковская, Д. Нейронные сети, генетические алгоритмы и нечеткие системы [Текст] / Д. Рутковская, М. Пилиньский, Л. Рудковский; пер. с польск. И. Д. Рудинского. – М.: Горячая линия-Телеком, 2004. – 452 с.
14. Воеводин, В. В. Вычислительная математика и структура алгоритмов. 10 лекций [Текст] / В. В. Воеводин. – М.: И-во МГУ, 2008. – 113 с.
15. Волков, Е. А. Численные методы [Текст] / Е. А. Волков; 2-е изд., испр. – М.: Наука, 1987. – 248 с.