УДК 519.6

Математичне моделювання оптимального розподілу потужностей ЕОМ в обчислювальних мережах

Роман Тичковський [1], Григорій Цегелик [2]

[1] асистент, Львівський національний університет імені Івана Франка, вул. Університетська, 1, Львів, 79000, e-mail: kafmmsep@franko.lviv.ua
[2] д. ф.-м. н., професор, Львівський національний університет імені Івана Франка, вул. Університетська, 1, Львів, 79000, e-mail: kafmmsep@franko.lviv.ua

Побудовано математичну модель оптимального розподілу потужностей ЕОМ в обчислювальній мережі при розв'язуванні фіксованої кількості задач різного типу. Модель враховує час виконання задач кожного типу на ЕОМ різних вузлів, обмеження на час використання ЕОМ кожного вузла. При цьому вважається, що ЕОМ різних вузлів можуть мати різну потужність. За критерій оптимальності вибрано сумарний час розв'язування задач. Отримана математична модель зводиться до узагальненої задачі про призначення. Для реалізації моделі запропоновано евристичний алгоритм. Проведено програмну реалізацію цього алгоритму та числовий експеримент.

Ключові слова: математичне моделювання, розподілене розв'язування задач, евристичний алгоритм.

Вступ. Однією з основних проблем проектування і реалізації систем розподіленої обробки інформації є проблема оптимального розподілу інформаційних ресурсів серед вузлів обчислювальної мережі (ОМ) [1, 2, 4]. Однак експлуатація ОМ може передбачати не тільки доступ користувачів до інформаційних ресурсів кожного вузла, а й можливість використання потужностей ЕОМ у будь-якому вузлі. З огляду на це виникає проблема побудови математичних моделей оптимального (на основі певного критерію) використання потужностей ЕОМ в ОМ і методів їхньої реалізації. Саме побудові таких математичних моделей та методам їхньої реалізації і присвячена дана праця.

1. Формулювання задачі

Нехай n — кількість вузлів ОМ; m — кількість різних типів задач, призначених до розв'язування; m_i — кількість задач i-го типу; K_j — j-й вузол ОМ; Z_i — i-та задача; t_{ij} — час виконання задачі Z_i на ЕОМ вузла K_j; t_j — час, протягом якого можна використати ЕОМ вузла K_j.

Треба так розподілити задачі між ЕОМ у мережі, щоб сумарний час розв'язування задач був мінімальний.

2. Побудова математичної моделі

У [3] побудовано математичну модель оптимального використання потужностей ЕОМ у мережі для випадку, коли треба розв'язувати тільки по одній задачі кожного типу. В цьому разі шукані змінні можуть набувати значень 0 або 1. У даній роботі розглядається випадок розв'язування фіксованої кількості задач кожного типу. Нехай x_{ij} — кількість задач *i*-го типу, які планують розв'язати на ЕОМ вузла K_j.

Тоді математичну модель можна записати у такому вигляді

$$L = \sum_{i=1}^{m} \sum_{j=1}^{n} t_{ij} x_{ij} \to \min \tag{1}$$

за умов

$$\sum_{i=1}^{m} t_{ij} x_{ij} \le t_j \quad (j = \overline{1, n}); \tag{2}$$

$$\sum_{j=1}^{n} x_{ij} = m_i \quad (i = \overline{1, m}); \tag{3}$$

$$x_{ij} \in \{\overline{1,\ m_i}\}\ (i = \overline{1, m};\ \ j = \overline{1, n}). \tag{4}$$

У цьому випадку цільова функція (1) виражає сумарний час, необхідний для розв'язування задач; ліва частина нерівності (2) визначає час, упродовж якого використовується ЕОМ вузла K_j; умова (3) означає, що кожна задача Z_i розв'язується на ЕОМ одного з вузлів.

Для реалізації математичної моделі запропоновано евристичний алгоритм.

3. Евристичний алгоритм реалізації математичної моделі

Зазначимо спочатку, таке: якщо сумарний час, призначених для розв'язування на ЕОМ задач в окремому вузлі є більший ніж час, протягом якого можна використати ЕОМ цього вузла, то такий вузол будемо називати переповненим.

Евристичний алгоритм, який пропонуємо використати для реалізації математичної моделі, складається з двох етапів. На першому етапі знаходимо початковий розподіл задач між вузлами. Цей розподіл буде завжди оптимальним, якщо не враховувати умову (2). На другому — перерозподілимо задачі, якщо для початкового розподілу існує хоча б один індекс $j = r$ такий, що умова (2) не справджується. Другий етап складається з низки кроків, до того ж на кожному кроці перерозподіляємо одну задачу з переповненого вузла так, щоб досягти мінімального збільшення значення цільової функції. Другий етап алгоритму виконуємо доти, доки не буде знайдено розподіл, який задовольняє умову (2). Опишемо обидва етапи алгоритму.

Перший етап. Знаходження початкового розподілу.

1. Для всіх i $(i = \overline{1, m})$ визначаємо $\min\limits_{1 \le s \le n} t_{is}$. Нехай $\min\limits_{1 \le s \le n} t_{is} = t_{is_i}$ $\ (i = \overline{1, m})$.

2. Знаходимо початковий розподіл задач, тобто визначаємо матрицю $\mathbf{X} = \left[x_{ij}\right]_{m,n}$, де

$x_{is_i} = m_i \ \left(i = \overline{1,\, m}\right);$

$x_{is} = 0 \ \left(i = \overline{1,\, m}; s = \overline{1,\, n};\ (i,\, s) \neq (i,\, s_i)\right).$

Розподіл $\mathbf{X}$ завжди буде оптимальним, якщо не враховувати умову (2).

Другий етап. Перерозподіл задач між вузлами.

1. Формуємо вектор ознак $E = \left(e_1, e_2, ..., e_n\right)$, де $e_j = 0 \ \left(j = \overline{1,\, n}\right)$. Елементи вектора ознак, які будуть дорівнювати 0, визначатимуть ті стовпці матриці **X**, елементи яких можна змінювати, а елементи вектора ознак, які будуть дорівнювати 1, визначатимуть ті стовпці матриці **X**, які надалі змінюватись не будуть. Якщо деякий вузол переповнений, то в процесі роботи алгоритму відбувається перерозподіл задач із цього вузла в інші. Після перерозподілу відповідному елементу e_j вектора E присвоюємо значення 1 і вузол вважаємо закритим для перерозподілу.

2. Для всіх індексів j, таких що $e_j = 0 \ \left(j = \overline{1,\, n}\right)$, перевіряємо виконання умови (2). Якщо ця умова для всіх j справджується, то на цьому робота алгоритму завершується. Розподіл **X** є оптимальний. Якщо для деякого індексу $j = r$ умова (2) не виконується, то переходимо до пункту 3.

3. Для кожного i, для якого $x_{ir} \neq 0$, знаходимо $\min\limits_{s}\left(t_{is} - t_{ir}\right)$, де мінімум беремо по тих індексах $s \neq r$, для яких $e_s = 0$. Нехай $\min\limits_{s}\left(t_{is} - t_{ir}\right) = t_{is_i} - t_{ir}$.

Далі визначаємо $\min\limits_{i}\left(t_{is_i} - t_{ir}\right)$, де мінімум беремо по тих індексах i, для яких $x_{ir} \neq 0$. Нехай $\min\limits_{i:x_{ir}\neq 0}\left(t_{is_i} - t_{ir}\right) = t_{qs_q} - t_{qr}$. Тоді в матриці **X** приймаємо $x_{qs_q} = x_{qs_q} + 1$, $x_{qr} = x_{qr} - 1$. Це означає, що задача Z_q із вузла K_r скеровується у вузол K_{s_q}. Такому перерозподілу відповідає мінімальне збільшення цільової функції.

4. Превіряємо умову (2) для $j = r$. Якщо вона не виконується, то переходимо до пункту 3. Якщо умова виконується, то вузол вважаємо закритим для перерозподілу задач, елементу e_r присвоюємо значення 1 і переходимо до пункту 5.

5. Для всіх j, таких, що $e_j = 0$, перевіряємо виконання умови (2). Якщо для всіх j умова виконується, то робота алгоритму завершується. Розподіл **X** приймаємо за розв'язок задачі. Якщо для деякого індексу $j = r$ умова (2) не справджується, то переходимо до пункту 3.

Отже, алгоритм дає змогу за скінченну кількість кроків знайти оптимальний або майже оптимальний розподіл задач між вузлами в обчислювальній мережі. Результатом роботи алгоритму є матриця **X**.

Визначимо максимально можливу кількість кроків. Для кожного знайденого r існує не більше ніж m елементів x_{lr}, значення яких можна зменшити на 1, тобто здійснити перерозподіл однієї задачі з r-го вузла в інший (виконуючи пункт 3 другого етапу алгоритму). Оскільки значення x_{lr} може бути не більше m_l, то таких перерозподілів можна зробити не більше ніж m_l. Сумарно для кожного знайденого r можна зробити не більше ніж $\sum_{l=1}^{m} m_l$ перерозподілів. Усіх індексів r, для яких не виконується умова (2), може бути не більше ніж n (до того ж після перегляду (n–1)-го індексу всі вузли будуть закриті для перерозподілу), то максимальна кількість кроків буде не більше ніж $(n-1)\sum_{l=1}^{m} m_l$.

4. Числовий експеримент

Числовий експеримент виконано для різних наборів вхідних даних. Тут подано результати роботи алгоритму для таких вхідних даних $n = 10$, $m = 20$, $m_i = 10 \ (i = \overline{1, 10})$, $m_i = 5 \ (i = \overline{11, 20})$; $t_1 = 1000$, $t_2 = 1000$, $t_3 = 2000$, $t_4 = 2000$, $t_5 = 2000$, $t_6 = 3000$, $t_7 = 4000$, $t_8 = 4000$, $t_9 = 4000$, $t_{10} = 4000$.

$$[t_{ij}] = \begin{vmatrix}
10 & 15 & 20 & 25 & 30 & 35 & 40 & 45 & 50 & 55 \\
20 & 30 & 40 & 50 & 60 & 70 & 80 & 90 & 100 & 110 \\
30 & 45 & 60 & 75 & 90 & 105 & 120 & 135 & 150 & 165 \\
40 & 60 & 80 & 100 & 120 & 140 & 160 & 180 & 200 & 220 \\
50 & 75 & 100 & 125 & 150 & 175 & 200 & 225 & 250 & 275 \\
60 & 90 & 120 & 150 & 180 & 210 & 240 & 270 & 300 & 330 \\
70 & 105 & 140 & 175 & 210 & 245 & 280 & 315 & 350 & 385 \\
80 & 120 & 160 & 200 & 240 & 280 & 320 & 360 & 400 & 440 \\
90 & 135 & 180 & 225 & 270 & 315 & 360 & 405 & 450 & 495 \\
100 & 150 & 200 & 250 & 300 & 350 & 400 & 450 & 500 & 550 \\
20 & 30 & 40 & 50 & 60 & 70 & 80 & 90 & 100 & 110 \\
30 & 45 & 60 & 75 & 90 & 105 & 120 & 135 & 150 & 165 \\
70 & 105 & 140 & 175 & 210 & 245 & 280 & 315 & 350 & 385 \\
80 & 120 & 160 & 200 & 240 & 280 & 320 & 360 & 400 & 440 \\
90 & 135 & 180 & 225 & 270 & 315 & 360 & 405 & 450 & 495 \\
100 & 150 & 200 & 250 & 300 & 350 & 400 & 450 & 500 & 550 \\
10 & 15 & 20 & 25 & 30 & 35 & 40 & 45 & 50 & 55 \\
20 & 30 & 40 & 50 & 60 & 70 & 80 & 90 & 100 & 110 \\
30 & 45 & 60 & 75 & 90 & 105 & 120 & 135 & 150 & 165 \\
40 & 60 & 80 & 100 & 120 & 140 & 160 & 180 & 200 & 220
\end{vmatrix}$$

Після виконання першого етапу алгоритму отримано такий розподіл задач між вузлами обчислювальної мережі

$$
[x_{ij}] = \left|\begin{array}{cccccccccc}
10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
10 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}\right|
$$

При цьому значення цільової функції $L = 7950$. Зауважимо, що отриманий розподіл не задовольняє умову (2).

Після виконання другого етапу алгоритму отримано такий розподіл

$$
[x_{ij}] = \left|\begin{array}{cccccccccc}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 10 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 7 & 3 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 2 & 8 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 10 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 6 & 4 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 8 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 7 & 3 & 0 & 0 & 0 & 0 & 0 & 0 \\
5 & 5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 5 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 5 & 0 \\
0 & 0 & 0 & 0 & 0 & 5 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 5 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 5 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 5 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 5 & 0 & 0
\end{array}\right|
$$

Тут значення цільової функції $L = 25100$. Значення цільової функції зросло, але отриманий допустимий розподіл уже задовольняє умову (2). При виконанні другого етапу алгоритму було здійснено 888 перерозподілів задач з одних вузлів у інші.

Зазначимо, що при розв'язуванні цієї ж задачі алгоритмом гілок і меж було отримано таке оптимальне значення цільової функції $L = 24865$. Бачимо, розподіл отриманий за допомогою евристичного алгоритму не є оптимальним, але він близький до оптимального. Перевага евристичного алгоритму над алгоритмом гілок і меж полягає в простоті реалізації. Крім того, евристичний алгоритм вимагає значно меншого часу при розв'язуванні задач.

Висновки. Побудовано математичну модель оптимального розподілу задач серед ЕОМ вузлів обчислювальної мережі. За критерій оптимальності вибрано сумарний час розв'язування задач. Математичною моделлю задачі є узагальнена задача про призначення, для розв'язування якої запропоновано евристичний алгоритм. Цей алгоритм дає змогу за скінченну кількість кроків знайти оптимальний або майже оптимальний розподіл задач серед вузлів ОМ. Недоліком алгоритму є те, що він не дає гарантії отримання оптимального розподілу в разі розв'язування задачі.

Література

[1] *Демидович О. В.* Математичні моделі оптимального розподілу інформаційних ресурсів серед вузлів обчислювальних мереж і методи їх реалізації: Автореф. дис. канд. техн. наук: 01.05.02 / НУ «Львівська Політехніка»— Львів, 2001. — 20 с.

[2] *Цегелик Г. Г.* Системы распределённых баз даных. — Львов: Свит, 1990. — 186 с.

[3] *Цегелик Г. Г., Тичковський Р. О.* Математичне моделювання оптимального використання потужностей ЕОМ в обчислювальних мережах // Вісник Львівського університету. Сер. прикл. матем. та інформ. — 2003. — Вип 6. — С. 178-181.

[4] *Lee H., Jang G.* Data file and workload allocation on a local multi-access computer networks: incorporating local processing and communication overhead // Int. Journal of System Science. — 1996. — Vol. 27, № 9. — P. 831-837.

The Mathematical Modeling for Optimal Distribution of Computers Power in Computing Networks

Roman Tychkovskiy, Grigoriy Tsegelyk

The mathematical model for optimal distribution powers of computers in computing networks for solving fixed count of different types' tasks is made. The model considers time performance of solving tasks on computers in different network nodes, time of using of computers nodes. The computer in different network nodes could have different powers. The total time of solving tasks is chosen for criterion of optimality. The obtained mathematical model is reduced to the generalized assignment problem. The heuristic algorithm for realization this model is proposed. The program realization of this algorithm and numerical experiments are made.

Математическое моделирование оптимального распределения мощностей ЭВМ в вычислительних сетях

Роман Тычковский, Григорий Цегелык

Построена математическая модель оптимального распределения мощностей ЭВМ в вычислительной сети при решении фиксированного количества задач каждого типа. Модель учитывает время выполнения задач на ЭВМ различных узлов, ограничения на время использования ЭВМ каждого узла. При этом считается, что ЭВМ различных узлов могут иметь разную мощность. В качестве критерия оптимальности выбрано суммарное время решения задач. Полученная математическая модель сводится к обобщённой задаче о назначениях. Для реализации модели предложен эвристический алгоритм. Приведена программная реализация этого алгоритма и численный эксперимент.

Отримано 06.09.05