УДК 519.85

О. О. Ємець, Є. М. Ємець, Ю. Ф. Олексійчук

Поліноміальний метод наближеного розв'язання комбінаторної задачі знаходження максимального потоку в мережі

(Представлено академіком НАН України І. В. Сергієнком)

Розглядається комбінаторна задача знаходження максимального потоку в мережі, яка зводиться до евклідової комбінаторної задачі на розміщеннях. Запропоновано наближений алгоритм для її розв'язання, визначено поліноміальну оцінку його складності.

Задача знаходження максимального потоку широко досліджена [1–7] та відомі поліноміальні методи для її розв'язання, зокрема, методи Форда і Фалкерсона [1], Едмондса і Карпа [4], Дініца [5], Карзанова [6] та ін. Задача, що розглядається, є узагальненням задачі знаходження максимального потоку. Комбінаторну задачу знаходження максимального потоку можна звести [8] до евклідової комбінаторної задачі оптимізації на розміщеннях [9–11], для розв'язання якої відомі неполіноміальні алгоритми. Тому побудова наближених поліноміальних методів є актуальною.

У роботі розглядається жадібний метод розв'язання комбінаторної задачі знаходження максимального потоку в мережі і аналізується його складність.

Постановка задачі. Нехай дано зв'язний орієнтований граф $\Gamma = (V, U)$, де V — множина вершин; U — множина дуг. Дугу, що сполучає вершини v_i і v_j, позначимо u_{ij}; v_i, $v_j \in V$.

Означення 1. Транспортною мережею називають орієнтований граф $\Gamma = (V, U)$, в якому кожній із дуг u_{ij} привласнене невід'ємне число $b_{ij} \geqslant 0$, яке називають пропускною спроможністю дуги. Хоча б одна із вершин має лише вихідні дуги. Таку вершину називають джерелом и позначають v_s. Також є принаймні одна вершина, що має лише вхідні дуги. Її називають стоком і позначають v_t.

Далі будемо розглядати транспортні мережі з одним джерелом і одним стоком. Вважатимемо, що кожна дуга u_{ij} в протилежному напрямку має пропускну спроможність $b_{ji} = 0$.

Означення 2 [7]. Потоком називають функцію $w\colon U \to \mathbb{R}$ з такими властивостями для будь-якої дуги u_{ij}:

1) значення функції w на дузі u_{ij} не може перевищувати пропускну здатність дуги, тобто $w(u_{ij}) \leqslant b_{ij}$;

2) збереження балансу у всіх вершинах, крім стоку и джерела, тобто $\sum\limits_{u_{iz}\in U} w(u_{iz}) = \sum\limits_{u_{zj}\in U} w(u_{zj})$ $\forall v_z,\ v_i,\ v_j \in V,\ v_z \ne v_s,\ v_z \ne v_t$.

Означення 3. Величиною потоку $|w|$ будемо називати суму значень функції w по дугах, що виходять із джерела: $|w| = \sum\limits_{u_{si}\in U} w(u_{si})$, де $v_i \in V$.

Величина потоку $|w|$ також дорівнює сумі значень функції w по дугах, що входять в стік.

Задача знаходження максимального потоку формулюється так: знайти потік w, для якого величина потоку $|w|$ є максимальною.

Означення 4. Залишковою пропускною спроможністю ребра u_{ij} називають величину $h_{ij} = b_{ij} - w(u_{ij})$.

Якщо в транспортній мережі замінити пропускні спроможності на залишкові пропускні спроможності, отримаємо мережу, яку назвемо залишковою.

Потоком на дузі u_{ij} будемо називати величину $w(u_{ij})$.

Введемо додаткові обмеження на потік. Нехай потік по дугах $u_{ij} \in U_c \subseteq U$ може набувати значень, які не перевищують число $x_{ij} = g_l \in G$, тобто $w(u_{ij}) \leqslant x_{ij}$, де мультимножина $G = \{g_1, g_2, \ldots, g_\eta\}$; причому вектор із чисел x_{ij} є розміщенням елементів із G [9–11], тобто $x = (x_{i_1j_1}, x_{i_2j_2}, \ldots, x_{i_kj_k}) \in E^k_{\eta n}(G)$, де $E^k_{\eta n}(G)$ — множина розміщень, $k = |U_c|$; n — кількість різних в G елементів.

Назвемо задачу знаходження максимального потоку з додатковими комбінаторними обмеженнями комбінаторною задачею знаходження максимального потоку.

Математична модель. Розглянемо математичну модель задачі. Нехай потік по дузі u_{ij} дорівнює y_{ij}, тобто $y_{ij} = w(u_{ij})$.

Задача полягає у знаходженні максимального значення функції f і відповідних значень x_{ij}, y_{ij}

$$f = \sum_{u_{jt} \in U} y_{jt} \to \max \tag{1}$$

при виконанні таких умов:

збереження балансу у вершинах

$$\sum_{u_{iz} \in U} y_{iz} = \sum_{u_{zj} \in U} y_{zj}, \qquad \forall\, z, \qquad z \neq t, \qquad z \neq s, \tag{2}$$

обмеження на пропускну здатність дуг

$$0 \leqslant y_{ij} \leqslant b_{ij} \qquad \forall\, i\, \forall\, j, \qquad u_{ij} \in U, \tag{3}$$

додаткові комбінаторні обмеження

$$y_{ij} \leqslant x_{ij} \qquad \forall\, i\, \forall\, j, \qquad u_{ij} \in U_c, \tag{4}$$

$$x = (x_{i_1j_1}, x_{i_2j_2}, \ldots, x_{i_kj_k}) \in E^k_{\eta n}(G). \tag{5}$$

Задача (1)–(5) є частковим випадком задачі евклідової комбінаторної оптимізації на розміщеннях.

Розробці методів розв'язання задач комбінаторної оптимізації, дослідженню властивостей опуклих оболонок комбінаторних множин присвячено багато робіт (див., зокрема, [9–12]). Методи розв'язання комбінаторних задач на розміщеннях з додатковими обмеженнями запропоновані, наприклад, в [11] і [12]; всі вони є неполіноміальними.

Зазначимо, що комбінаторну задачу знаходження максимального потоку в мережі можна звести до задачі евклідової комбінаторної оптимізації на переставленнях. Для цього достатньо побудувати нову мультимножину, взявши k найбільших елементів мультимножини G, де k — кількість дуг, на які накладені комбінаторні обмеження.

У [8] розглядається метод розв'язання задачі (1)–(5). Але з урахуванням неполіноміальності методу актуальною є розробка наближених методів розв'язання задачі.

Жадібний метод. Розглянемо метод, який дозволяє знайти наближений розв'язок поставленої задачі. В методі використовуються ідеї Едмондса і Карпа [4] для знаходження максимального потоку. На кожному етапі максимально насичується найкоротший шлях, тому метод названий жадібним.

К р о к 0. Покладаємо потік рівним нулю. Розширимо мультимножину G, додавши k нулів, де k — кількість дуг, на які накладено додаткові комбінаторні обмеження. Для всіх дуг покладаємо $x_{ij} = 0$ і будемо вважати всі ненульові елементи мультимножини G невикористаними. Залишкова мережа на початковому етапі збігається з початковою.

К р о к 1. Знаходимо в залишковій мережі, не враховуючи додаткових комбінаторних обмежень, найкоротший шлях із джерела в стік з максимальною пропускною здатністю, використовуючи модифікований пошук у ширину. Нехай шлях складається із p дуг і має пропускну здатність, що дорівнює b. (Під пропускною здатністю шляху розуміємо мінімальну пропускну здатність серед дуг цього шляху.) Якщо такого шляху немає — зупинка алгоритму.

К р о к 2. Переберемо всі дуги знайденого шляху у порядку спадання пропускних спроможностей b_{ij}. Привласнимо всім x_{ij}, для яких $x_{ij} < b + y_{ij}$, мінімальні значення із мультимножини G, які незайняті і перевищують $b + y_{ij}$, де y_{ij} — поточне значення потоку по дузі u_{ij}. Якщо такого елемента немає, беремо найбільший елемент із G. Вважатимемо ці елементи мультимножини зайнятими, а попередні значення x_{ij} — незайнятими.

К р о к 3. Пускаємо через знайдений шлях максимально можливий потік (враховуючи комбінаторні обмеження).

К р о к 4. Модифікуємо залишкову мережу. Змінюємо пропускні здатності і видаляємо дуги, для яких виконується хоча б одна умова:

а) $y_{ij} = b_{ij}$;

б) $y_{ij} = x_{ij}$ і в G немає вільних елементів, що перевищують x_{ij}.

При цьому, якщо умови *а* і *б* більше не виконуються для деякої раніше видаленої дуги, повертаємо її в залишкову мережу.

Переходимо на крок 1.

Модифікований пошук в ширину дозволяє знайти найкоротший шлях з джерела до стоку. Якщо таких шляхів кілька, метод дозволяє знайти серед них шлях з найбільшою пропускною здатністю.

Для кожної вершини v_i потрібно задати такі поля:

1) стан вершини, можливі значення: не помічена, помічена, переглянута;

2) предок — інша вершина графу;

3) відстань від джерела d_i;

4) потенційний потік p_i.

Модифікований пошук в ширину.

К р о к 0. Джерело вважаємо поміченим, $p_s = \infty$, $d_s = 0$. Всі інші вершини — не поміченими, $p_i = 0$, $\forall\, i$, $i \neq s$. Створюємо чергу із помічених вершин.

К р о к 1. Якщо черга порожня — шляху не існує, зупинка алгоритму, інакше — вважаємо першу вершину черги v_c переглянутою і видаляємо її із черги.

К р о к 2. Розглядаємо всі дуги u_{ci}, що виходять з вершини v_c, і перебираємо вершини v_i, які або містяться в черзі і $d_i > d_c$, або є непоміченими. Якщо $p_i < \min(p_c, b_{ci})$, то $p_i = \min(p_c, b_{ci})$, предком вершини v_i стає вершина v_c, $d_i = d_c + 1$ та, якщо вершина непомічена, помічаємо її та додаємо в кінець черги.

К р о к 3. Якщо v_t міститься в черзі і $d_t = d_z$, де v_z — перша в черзі вершина, то зупинка алгоритму. (Зворотний до шуканого шлях можна отримати, почавши зі стоку і переходячи від вершини до її предка, доки не досягнемо джерела.) Інакше — перехід до кроку 1.

Оцінка складності жадібного методу. Нехай $|V|$ — кількість вершин графу, $|U|$ — кількість дуг.

Теорема 1. *Час роботи модифікованого алгоритму пошуку в ширину* $T(n) = O(|U| + + |V|)$.

Зауваження. Оскільки в роботі розглядаються зв'язні графи, то час роботи модифікованого алгоритму пошуку в ширину дорівнює $T(n) = O(|U|)$.

Теорема 2. *Час роботи жадібного алгоритму розв'язання комбінаторної задачі знаходження максимального потоку* $T(n = O(|U| \cdot n \cdot |w^*|)$, *де* $|w^*|$ — *шукана величина максимального потоку;* n — *кількість різних елементів у мультимножині* G.

Таким чином, в роботі розглянута комбінаторна задача знаходження максимального потоку, запропонований наближений метод її розв'язання та зроблена оцінка часу роботи цього методу. Метод може застосовуватися для знаходження початкового розв'язку в точних методах. В подальшому доцільно оцінити точність роботи алгоритму та провести обчислювальні експерименти.

1. *Форд Л., Фалкерсон Д.* Потоки в сетях. – Москва: Мир, 1966. – 277 с.
2. *Ху Т. Ч., Шинг М. Т.* Комбинаторные алгоритмы. – Нижний Новгород: Изд-во Нижегород. гос. ун-та им. Н. И. Лобачевского, 2004. – 330 с.
3. *Ху Т.* Целочисленное программирование и потоки в сетях – Москва: Мир, 1974. – 519 с.
4. *Edmonds J., Karp R. M.* Theoretical improvements in algorithmic efficiency for network flow problems // J. ACM. – 1972. – **19**, No 2. – P. 248–264.
5. *Диниц Е. А.* Алгоритм решения задачи о максимальном потоке в сети со степенной оценкой // Докл. АН СССР. – 1970. – **194**, № 4. – С. 754–757.
6. *Карзанов А. А.* Нахождение максимального потока в сети методом предпотоков // Там же. – 1974. – **215**, № 1. – С. 49–53.
7. *Кормен Т., Лейзерсон Ч., Ривест Р., Штайн К.* Алгоритмы: построение и анализ. 2-е изд. – Москва: ИД "Вильямс", 2005. – 1296 с.
8. *Ємець О. О., Ємець Є. М., Олексійчук Ю. Ф.* Знаходження максимального потоку в мережі з додатковими комбінаторними обмеженнями // Таврич. вестник информатики и математики. – 2011. – № 1. – С. 43–50.
9. *Стоян Ю. Г., Ємець О. О.* Теорія і методи евклідової комбінаторної оптимізації – Київ: ІСДО, 1993. – 188 с. (див. також http://informatics.org.ua/uploads/books/stoyan_emets_eko.pdf).
10. *Стоян Ю. Г., Ємець О. О., Ємець Є. М.* Оптимізація на полірозміщеннях: теорія та методи. – Полтава: РВЦ ПУСКУ, 2005. – 103 с. (http://informatics.org.ua/uploads/books/st_em_em_monography.pdf).
11. *Емец О. А., Барболина Т. Н.* Комбинаторная оптимизация на размещениях: Монография. – Киев: Наук. думка, 2008. – 159 с. (http://informatics.org.ua/uploads/books/em_bar_monography.pdf).
12. *Ємець О. О., Ємець Є. М., Олексійчук Ю. Ф.* Прямий метод відсікання для задач комбінаторної оптимізації на розміщеннях // Вісн. Запорізьк. нац. ун-ту: Зб. наук. статей. Фіз.-мат. науки. – 2011. – № 1. – С. 36–43.

Полтавський університет економіки і торгівлі

Надійшло до редакції 08.06.2012

О. А. Емец, Е. М. Емец, Ю. Ф. Олексийчук

Полиномиальный метод приближенного решения комбинаторной задачи нахождения максимального потока в сети

Рассматривается комбинаторная задача нахождения максимального потока в сети, которая сводится к эвклидовой комбинаторной задаче на размещениях. Предложен приближенный алгоритм для ее решения, определена полиномиальная оценка его сложности.

O. O. Iemets, Ye. M. Yemets, Yu. F. Oleksiichuk

An approximate polynomial method for solving a combinatorial problem of finding the maximum flow in a network

The combinatorial problem finding of the maximal flow in a network is considered. This problem is a Euclidean combinatorial one on arrangements. An approximate algorithm for the solution of this problem is proposed. The polynomial estimation of the complexity of this algorithm is found.