ДОКЛАДЫ АКАДЕМИИ НАУК, 2014, том 455, № 5, с. 529–532

ИНФОРМАТИКА

УДК 519.1

ЭФФЕКТИВНОЕ ВЫЧИСЛЕНИЕ ДОПУСКОВ В ЗАДАЧЕ О ВЗВЕШЕННОМ НЕЗАВИСИМОМ МНОЖЕСТВЕ ДЛЯ НЕКОТОРЫХ КЛАССОВ ГРАФОВ

© 2014 г. Д. С. Малышев, П. М. Пардалос

Представлено академиком Ю.Г. Евтушенко 11.10.2013 г.

Поступило 11.10.2013 г.

DOI: 10.7868/S0869565214110097

После получения оптимального решения задачи комбинаторной оптимизации (ЗКО) следующим естественным шагом является анализ его чувствительности. Анализ чувствительности полученного оптимального решения ЗКО состоит в выяснении зависимости этого решения от изменения исходных данных. В простейшем анализе чувствительности исследуются возмущения только одного элемента оптимального решения. Целью изучения таких возмущений является нахождение допусков, определяемых как максимальное изменение индивидуальной стоимости (веса, времени и т.п.), сохраняющее оптимальность заданного решения при условии сохранения неизменными остальных данных ЗКО.

Допуски представляют интерес в связи с тем, что минимальное значение допусков элементов оптимального решения задачи является оценкой снизу радиуса устойчивости этого оптимального решения и служит основой для построения и улучшения алгоритмов решения некоторых ЗКО. Первое неявное алгоритмическое использование допусков можно найти в методе Фогеля [12] для нахождения решения, наиболее близкого к оптимальному базисному решению в симплекс-методе решения транспортной задачи. В числе случаев успешного использования допусков отметим точные алгоритмы для решения задачи коммивояжера для ориентированных графов [6] и линейной задачи о назначении [1].

Особое внимание уделяется эффективно решаемым классам ЗКО, для которых возможно одновременное вычисление оптимального решения и всех допусков. Примерами таких задач являются задачи о минимальном остовном дереве [14], о кратчайшем пути [11, 13], о назначении [15], о взвешенном независимом множестве для деревьев [2].

Настоящая работа является продолжением работы [2]. Она посвящена одновременному определению оптимального решения и всех соответствующих допусков в задаче о взвешенном независимом множестве (ЗВНМ) для некоторых классов графов. Данная задача для обыкновенного графа с положительными весами вершин состоит в том, чтобы найти в нем множество попарно несмежных вершин наибольшего веса. Рассматриваются двудольные и интервальные графы и показывается, что:

1) для двудольного графа с n вершинами и m ребрами оптимальное решение ЗВНМ находится за время $O(nm)$, а все допуски вычисляются за время $O(n^2)$;

2) для интервального графа с n вершинами и m ребрами и оптимальное решение ЗВНМ и все допуски вычисляются за линейное время $O(n + m)$.

Полученные результаты не только обобщают результат из [2], но и будут полезными при разработке точных и приближенных алгоритмов решения ЗВНМ в общем случае. С этой целью алгоритм из [2] уже был использован в работе [7]. Использование ЗВНМ для интервальных графов является перспективным в качестве релаксации некоторых NP-полных задач из области сенсорных сетей [4].

1. ЗАДАЧА КОМБИНАТОРНОЙ ОПТИМИЗАЦИИ И ДОПУСКИ ЕЕ ЭЛЕМЕНТОВ

Задача комбинаторной оптимизации определяется четверкой параметров $\{\Gamma, c, F, f_c\}$, где $c\colon \Gamma \to \mathbb{R}_+$ – весовая функция универсального множества Γ, $F \subseteq 2^{\Gamma}$ – множество допустимых решений задачи, f_c – целевая функция. Для задан-

Нижегородский филиал Национального исследовательского университета, "Высшая школа экономики"

Нижегородский государственный университет им. Н.И. Лобачевского

University of Florida, USA

ной конкретной четверки $\{\Gamma, c, F, f_c\}$ соответствующая ЗКО состоит в том, чтобы отыскать такой элемент $S^* \in F$, на котором целевая функция принимает свое экстремальное (минимальное или максимальное) значение. Такое множество S^* называется оптимальным решением задачи. Задача комбинаторной оптимизации называется аддитивной, если для любого $S \in F$ справедливо $f_c(S) = \sum_{s \in S} c(s)$.

Пусть задана аддитивная ЗКО на максимум, некоторое ее оптимальное решение S^* и элементы $x \in S^*$ и $y \in \Gamma \backslash S^*$. Данную задачу обозначим через П. По этой задаче образуем две другие задачи Π_1 и Π_2. Задача Π_1 получается из П путем уменьшения веса элемента x на неотрицательное число w_1. Задача Π_2 получается из П путем увеличения веса элемента y на некоторое неотрицательное число w_2. Вернемся к рассмотрению задачи П. Нижним допуском элемента x относительно заданного оптимального решения S^* (обозначаемым через $l_{S^*}(x)$) называется супремум тех чисел w_1, для которых множество S^* остается оптимальным решением Π_1 при условии сохранения всех остальных данных П неизменными. Верхним допуском элемента y относительно оптимального решения S^* (обозначаемым через $u_{S^*}(y)$) называется супремум тех чисел w_2, для которых множество S^* остается оптимальным решением Π_2 при условии сохранения всех остальных данных П неизменными. Смысл понятий нижнего и верхнего допусков состоит в том, что их минимальные значения являются оценками радиуса устойчивости решения S^* задачи П. Более точно, для задачи П нет решений S', $S'' \in F$, что $f_c(S^*) > f_c(S') > f_c(S^*) - l$ или $f_c(S^*) > f_c(S'') > f_c(S^*) - u$, где $l = \min_{x \in S^*} l_{S^*}(x)$ и $u = \min_{y \notin S^*} u_{S^*}(y)$.

С задачей П ассоциируем следующие обозначения:

$$F_-(x) = \{S \in F: x \notin S\},$$
$$F_+(y) = \{S \in F: y \in S\};$$

F^* – множество оптимальных решений задачи П;

$$F_-^*(x) = F_-(x) \cap F^*, \quad F_+^*(y) = F_+(y) \cap F^*;$$

$$f_c(F') = \max_{S \in F'} f_c(S), \text{ где } F' \text{ – подмножество } F.$$

Справедлива (см., например, [3] или [10]) следующая

Л е м м а 1. *Пусть* $S^* \in F^*$, $x \in S^*$ *и* $y \notin S^*$. *Тогда справедливы равенства* $l_{S^*}(x) = f_c(F^*) - f_c(F_-^*(x))$ *и* $u_{S^*}(y) = f_c(F^*) - f_c(F_+^*(y))$.

2. СЛУЧАИ СОВМЕСТНОГО ЭФФЕКТИВНОГО ВЫЧИСЛЕНИЯ ОПТИМАЛЬНОГО РЕШЕНИЯ ЗКО И ВСЕХ ДОПУСКОВ

По лемме 1 при известном значении $f_c(F^*)$ для вычисления нижних и верхних допусков элементов x и y достаточно вычислять $f_c(F_-^*(x))$ и $f_c(F_+^*(y))$. Вместе с тем, информацию, накопленную при вычислении $f_c(F^*)$, целесообразно использовать для определения $f_c(F_-^*(x))$ и $f_c(F_+^*(y))$ (а, скажем, не запускать алгоритм вновь для совокупности данных без/с x/y). Это одна из идей для вычисления всех допусков с вычислительной сложностью, не превосходящей (или слегка большей) трудоемкости вычисления оптимального решения.

В работе [13] было показано, что для заданных взвешенного по ребрам графа с n вершинами и некоторых оптимальных решений задач о максимальном потоке и о кратчайшем пути можно вычислить допуски всех ребер за время $O(n^2)$. Р. Тарьян в [14] показал, что для задачи о минимальном остовном дереве при известном оптимальном решении все допуски относительно него вычисляются за время $O(m\alpha(n, m))$ ($\alpha(n, m)$ – весьма медленно растущая обратная функция Аккермана). Особенный интерес вызывают случаи, когда и оптимальное решение и все допуски относительно него могут быть вычислены за линейное время. В статье [5] рассматривались задачи о кратчайшем пути и о минимальном остовном дереве и было показано, что для данных задач и планарного графа с n вершинами оптимальное решение и все допуски вычисляются за время $O(n)$. За это же время можно вычислить оптимальное решение и все допуски в ЗВНМ для дерева с n вершинами [2].

В настоящей работе будет показано, что для ЗВНМ и случаев двудольных и интервальных графов все допуски вычисляются за полиномиальное время, не большее времени нахождения оптимального решения.

3. ДВУДОЛЬНЫЕ И ИНТЕРВАЛЬНЫЕ ГРАФЫ

Граф G называется двудольным, если множество $V(G)$ можно разбить на две части X и Y (называемые долями G) так, что $E(G) \subseteq X \times Y$. Граф G называется интервальным, если с каждой его вершиной можно ассоциировать интервал числовой прямой так, что любые две вершины G являются смежными тогда и только тогда, когда соответствующие им интервалы пересекаются. Набор интервалов называется интервальным представлением G.

4. ПРИМЕНЕНИЕ ПОТОКОВЫХ АЛГОРИТМОВ К ВЫЧИСЛЕНИЮ ДОПУСКОВ В ЗВНМ ДЛЯ ДВУДОЛЬНЫХ ГРАФОВ

Хорошо известно сведение ЗВНМ для двудольных графов к поиску максимального потока в сети. Опишем суть этого сведения. Пусть заданы произвольный двудольный граф G с долями X и Y и весовая функция c: $X \cup Y \to \mathbb{R}_+$. Построим взвешенный по ребрам граф G' следующим образом. Добавим к G две вершины s и t, s соединим ребрами со всеми вершинами доли X, t соединим ребрами со всеми вершинами из Y. Каждому ребру $e = (x, y)$ графа G' назначим вес $c'(e)$ в соответствии с правилом: если $x = s$, то $c'(e) = c(y)$; если $y = t$, то $c'(e) = c(x)$; во всех остальных случаях $c'(e) = \infty$ (бесконечность можно заменить на сумму $\sum\limits_{v \in X \cup Y} c(v)$). Напомним, что (s, t)-разрезом графа G' называется любое разбиение множества его вершин на части S и T, такое что $s \in S$ и $t \in T$. Весом этого разреза называется сумма $\sum\limits_{v \in S, u \in T} c'(v, u)$. Хорошо известно, что если (S, T) – разрез G' минимального веса, то $(A \cap S) \cup (B \cap T)$ – независимое множество графа G наибольшего веса. По теореме Форда–Фолкерсона минимальный по весу разрез графа G' может быть найден путем его превращения в сеть (естественной ориентацией ребер G' с источником s и стоком t и пропускными способностями дуг, равными соответствующим весам) и применением к ней любого потокового алгоритма. На сегодняшний день наилучшим алгоритмом для вычисления максимального потока в сети с n вершинами и m дугами является алгоритм Орлина [9] с трудоемкостью $O(nm)$. Поэтому для двудольного графа с n вершинами и m ребрами ЗВНМ может быть решена за время $O(nm)$ и по результату из [13] допуски относительно найденного оптимального решения вычисляются за время $O(n^2)$.

5. ВЫЧИСЛЕНИЕ ДОПУСКОВ В ЗВНМ ДЛЯ ИНТЕРВАЛЬНЫХ ГРАФОВ ЗА ЛИНЕЙНОЕ ВРЕМЯ

В этом разделе работы предполагается, что $\{\Gamma, c, F, f_c\}$ – ЗВНМ для интервального графа G с интервальным представлением $I(G)$. Будем считать, что вершины $v_1, v_2, \ldots, v_n$ графа G линейно упорядочены так, что $i \le j \Leftrightarrow a_i \le a_j$, где a_i – координата левого конца интервала из $I(G)$, соответствующего вершине v_i. По графу G представление $I(G)$ и это упорядочивание вычисляются за линейное время от суммы количеств его вершин и ребер (следствие из алгоритма 9 работы [8]). Оптимальное решение ЗВНМ для графа G и допуски всех вершин относительно этого решения предлагается определять при помощи метода динамического программирования. Для этого для каждого $1 \le i \le n$ введем следующие обозначения:

1) $j_i' = \min(\{j: (v_j, v_i) \in E(G)\} \cup \{i\}) - 1$;

2) $j_i'' = \max(\{j: (v_j, v_i) \in E(G)\} \cup \{i\}) + 1$;

3) G_i' (соответственно G_i'') – граф, получающийся из G удалениями вершин $v_{i+1}, v_{i+2}, \ldots, v_n$ (соответственно $v_1, v_2, \ldots, v_{i-1}$);

4) $S'(i)$ и $S''(i)$ – оптимальные независимые множества графов G_i' и G_i'' соответственно;

5) $S_{\text{in}}'(i)$ и $S_{\text{out}}'(i)$ (соответственно $S_{\text{in}}''(i)$ и $S_{\text{out}}''(i)$) – независимые множества графа G_i' (соответственно графа G_i''), имеющие наибольший вес среди всех его независимых множеств, содержащих/не содержащих вершину v_i;

6) $W'(i)$, $W_{\text{in}}'(i)$, $W_{\text{out}}'(i)$, $W''(i)$, $W_{\text{in}}''(i)$, $W_{\text{out}}''(i)$ – веса множеств $S'(i)$, $S_{\text{in}}'(i)$, $S_{\text{out}}'(i)$, $S''(i)$, $S_{\text{in}}''(i)$, $S_{\text{out}}''(i)$ соответственно, а также доопределим $W'(0) = W_{\text{in}}'(0) = W_{\text{out}}'(0) = W''(n+1) = W_{\text{in}}''(n+1) = W_{\text{out}}''(n+1) = 0$ и $S'(0) = S_{\text{in}}'(0) = S_{\text{out}}'(0) = S''(n+1) = S_{\text{in}}''(n+1) = S_{\text{out}}''(n+1) = \varnothing$.

Свойство интервальности графа G приводит к следующим рекуррентным уравнениям: $W_{\text{out}}'(i) = W'(i-1)$, $W_{\text{in}}'(i) = c(v_i) + W'(j_i')$, $S_{\text{out}}'(i) = S'(i-1)$, $S_{\text{in}}'(i) = S'(j_i') \cup \{v_i\}$. Ясно, что $W'(i) = \max(W_{\text{in}}'(i), W_{\text{out}}'(i))$ и что $S'(i) = S_{\text{out}}'(i)$, если $W'(i) \ge W_{\text{in}}'(i)$, и $S'(i) = S_{\text{in}}'(i)$ в противном случае. По аналогии выписываются рекуррентные уравнения для $W''(i)$, $W_{\text{in}}''(i)$, $W_{\text{out}}''(i)$, $S''(i)$, $S_{\text{in}}''(i)$, $S_{\text{out}}''(i)$.

Из леммы 1 и интервальности G следует, что для любой вершины $v_i \notin S^* = S'(n)$ выполнено равенство $u_{S^*}(v_i) = W'(n) - (W_{\text{in}}'(i) + W''(j_i''))$, причем $S_{\text{in}}'(i) \cup S''(j_i'') \in F_+^*(v_i)$. Из того же свойства интервальности G следует, что для любой вершины $v_i \in S^*$ величина $f_c(F_-^*(v_i))$ равна

$$\max\Bigl(\max_{k \in \overline{j_i'+1, j_i''-1} \setminus \{i\}} (W_{\text{in}}'(k) + W''(j_k'')), W'j_i') + W''(j_i'')\Bigr).$$

Соответствующим образом определяется некоторый элемент множества $F_-^*(v_i)$. Таким образом, для любой вершины $v_i \in S^*$ нижний допуск $l_{S^*}(v_i)$ равен $W'(n) - \max\Bigl(\max\limits_{k \in \overline{j_i'+1, j_i''-1} \setminus \{i\}} (W_{\text{in}}'(k) + W''(j_k'')), W'(j_i') + W''(j_i'')\Bigr)$.

Псевдокод решения ЗВНМ для графа G и последующего вычисления всех допусков представлен далее. В нем опущено вычисление значений переменных $S''(i)$, $S''_{\text{in}}(i)$, $S''_{\text{out}}(i)$, в котором нет необходимости для подсчета всех допусков. При необходимости определения элемента из $F^*_-(v_i)$ для $v_i \in S^*$ (элемента из $F^*_+(v_i)$ для $v_i \notin S^*$) он может быть естественным образом дополнен без потери линейности времени.

Алгоритм 1.

$W'(0) = 0$; $W'_{\text{in}}(0) = 0$; $W'_{\text{out}}(0) = 0$; $W''(n+1) = 0$; $W''_{\text{in}}(n+1) = 0$; $W''_{\text{out}}(n+1) = 0$;

$S'(0) = \phi$; $S'_{\text{in}}(0) = \phi$; $S'_{\text{out}}(0) = \phi$;

for $(i \in 1, 2, \ldots, n)$

{

$j'_i = \min(\{j: (v_j, v_i) \in E(G)\} \cup \{i\}) - 1$;

$j''_i = \max(\{j: (v_j, v_i) \in E(G)\} \cup \{i\}) + 1$;

}

for $(i \in \overline{1, n})$

{

$W'_{\text{out}}(i) = W'(i-1)$; $W'_{\text{in}}(i) = c(v_i) + W'(j'_i)$;

$S'_{\text{out}}(i) = S'(i-1)$; $S'_{\text{in}}(i) = S'(j'_i) \cup \{v_i\}$;

$W''_{\text{out}}(n-i+1) = W''(n-i+2)$; $W'_{\text{in}}(n-i+1) = c(v_{n-i+1}) + W'(j''_{n-i+1})$;

$W'(i) = \max(W'_{\text{in}}(i), W''_{\text{out}}(i))$; $W''(n-i+1) = \max(W''_{\text{in}}(n-i+1), W''_{\text{out}}(n-i+1))$;

if $(W'(i) > W'_{\text{out}}(i))$ $S'(i) = S'_{\text{in}}(i)$;

else $S'(i) = S'_{\text{out}}(i)$;

}

for $\{v_i \in S'(n)\}$

{

$$l_{S^*}(v_i) = W'(n) - \max\Big(\max_{k \in \overline{j'_i+1, j''_i-1} \setminus \{i\}} (W'_{\text{in}}(k) + W''(j''_k)), W'(j'_i) + W''(j''_i)\Big);$$

}

for $(v_i \in S'(n))$ $\{u_{S^*}(v_i) = W'(n) - W'_{\text{in}}(i) - W''(j''_i);\}$

Будем хранить множества $S'(i)$, $S'_{\text{in}}(i)$, $S'_{\text{out}}(i)$ односвязными списками; это обеспечивает возможность объединения таких списков за время $O(1)$. Нетрудно убедиться в том, что при таком хранении вычислительная сложность представленного псевдокода есть $O(|V(G)| + |E(G)|)$.

Работа выполнена при поддержке Лаборатории алгоритмов и технологий анализа сетевых структур НИУ ВШЭ, грант правительства РФ, договор 11.G34.31.0057.

СПИСОК ЛИТЕРАТУРЫ

1. *Бойко В., Голденгорин Б., Кузьменко В.* В кн.: Теорія оптимальних рішень: Нац. АН України; Ін-т кібернетики ім. В.М. Глушкова, 2006. Т. 5. С. 98–104.
2. *Голденгорин Б.И., Малышев Д.С., Пардалос П.М.* // ДАН. 2013. Т. 450. № 4. С. 393–396.
3. *Леонтьев В.К., Гордеев Э.Н.* // Кибернетика. 1986. № 5. С. 82–89.
4. *Boginski V., Commander C., Pardalos P., Ye Y.* Sensors: Theory, Algorithms, and Applications. B.: Springer, 2012.
5. *Booth H., Westbrook J.* // Algorithmica. 1994. V. 11. P. 341–352.
6. *Germs R., Goldengorin B., Turkensteen M.* // Comput. and Oper. Res. 2012. V. 39. P. 291–298.
7. *Goldengorin B., Malyshev D., Pardalos P., Zamaraev V.* // J. Combin. Opt. 2013. doi: 10.1007/s10878-013-9606-z.
8. *Habib M., McConnell R., Paul C., Viennot L.* // Theor. Comput. Sci. 2000. V. 234. P. 59–84.
9. *Orlin J.* // STOC. 2013. P. 765–774.
10. *Libura M.* // Discrete Appl. Math. 1991. V. 30. P. 197–211.
11. *Ramaswamy R., Orlin J., Chakravarti N.* // Math. Program. 2005. V. 102. P. 355–369.
12. *Reinfeld N., Vogel W.* Mathematical Programming. Englewood Cliffs (N. J.): Prentice-Hall, 1958.
13. *Shier D., Witzgall C.* // Networks. 1980. V. 10. P. 277–291.
14. *Tarjan R.* // Inform. Process. Lett. 1982. V. 14. P. 30–33.
15. *Volgenant A.* // Europ. J. Oper. Res. 2006. V. 169. P. 338–339.