ДИСКРЕТНЫЙ АНАЛИЗ И ИССЛЕДОВАНИЕ ОПЕРАЦИЙ
Сентябрь—октябрь 2009. Том 16, № 5. С. 34–40

УДК 519.718

ОБ ОДНОЙ ДВУХКРИТЕРИАЛЬНОЙ ЗАДАЧЕ НА ГРАФАХ

В. Г. Визинг

Аннотация. Предполагается, что каждое ребро графа имеет две числовые характеристики — длину и ширину. *Длиной* подграфа называется сумма длин его рёбер, *шириной* подграфа — минимальная ширина его ребра. Длина подграфа является негативной характеристикой, ширина — позитивной. Подграфы определённого вида называются допустимыми. Рассматривается двухкритериальная задача отыскания оптимального по Парето допустимого подграфа.

Ключевые слова: допустимый подграф, индикатор качества подграфа, оптимальный по Парето подграф.

Введение. Постановка задачи

Будем рассматривать конечные неориентированные графы без петель и кратных рёбер [2]. Граф называется *нечётким* [3], если для каждого его ребра e указана функция принадлежности $\mu(e)$, удовлетворяющая неравенствам $0 < \mu(e) \leqslant 1$. Функцию $\mu(e)$ можно считать весом ребра e, поэтому нечёткий граф можно рассматривать как взвешенный граф. В тех задачах, которым посвящена настоящая заметка, предполагается, что каждое ребро нечёткого графа, кроме функции принадлежности, имеет ещё одну числовую характеристику. Это приводит к необходимости рассматривать двухкритериальные оптимизационные задачи.

Тот факт, что функция принадлежности не больше 1, для рассматриваемых нами задач несуществен. Поэтому под графом G будем понимать четвёрку $G = (V, E, l, w)$, где V — множество вершин; E — множество рёбер; l и w — функции, определённые на рёбрах и принимающие положительные значения; $l(e)$ и $w(e)$ называются соответственно *длиной* и *шириной* ребра $e \in E$. Граф $G' = (V', E', l', w')$ называется *подграфом* графа $G = (V, E, l, w)$, если $V' \subseteq V$, $E' \subseteq E$ и $l'(e) = l(e)$, $w'(e) = w(e)$ при всех $e \in E'$. Если G' — подграф графа G, то G называется *надграфом* графа G'.

Мы будем рассматривать задачи, в которых требуется для данного графа найти оптимальный подграф среди подграфов определённого вида, называемых *допустимыми подграфами.* В разд. 1 понятие допустимого подграфа используется в абстрактной форме, в разд. 2 допустимым

подграфом для связного графа считается его остов, т. е. подграф, являющийся деревом с тем же множеством вершин, в разд. 3 под допустимым подграфом понимается простая цепь, соединяющая две заданные вершины.

Будем предполагать, что допустимый подграф для графа G является допустимым подграфом для любого надграфа графа G с тем же множеством вершин.

Каждый допустимый подграф H имеет две числовые характеристики: $L(H)$ — сумму длин рёбер, называемую длиной H, и $W(H)$ — минимальную ширину ребра, называемую шириной H. Такую упорядоченную пару (x, y), где x — длина, y — ширина подграфа, будем называть *индикатором качества* (или, короче, индикатором) подграфа. Первая компонента индикатора является негативной характеристикой качества подграфа, т. е. чем она больше — тем хуже, вторая компонента является позитивной характеристикой качества.

Допустимый подграф H графа G и его индикатор качества (x_0, y_0) называются *оптимальными по Парето* при выполнении следующего условия: если H' — допустимый подграф графа G с индикатором качества (x', y') такой, что $x' \leqslant x_0$, $y' \geqslant y_0$, то $x' = x_0$, $y' = y_0$.

Настоящая заметка посвящена задаче выявления всех оптимальных по Парето индикаторов качества. Для каждого оптимального индикатора строится допустимый подграф, индикатором качества которого является этот оптимальный индикатор. Тем самым получается *система различных представителей оптимальных по Парето подграфов.* Алгоритм для отыскания такой системы, изложенный в разд. 1, применяется в разд. 2 для построения оптимальных остовов, в разд. 3 — для построения оптимальных цепей между заданными вершинами графа.

1. Система различных представителей оптимальных по Парето подграфов

Пусть $G = (V, E, l, w)$ — граф, имеющий допустимые подграфы, A_L — алгоритм для построения допустимого подграфа с минимальной длиной, A_W — алгоритм, определяющий максимальную ширину допустимого подграфа.

Алгоритм для построения системы различных представителей оптимальных по Парето подграфов состоит в следующем.

Алгоритм A_0.

Шаг 1. С помощью алгоритма A_W определим максимальную ширину допустимого подграфа. Пусть эта ширина равна w_1. Рассмотрим рёбра

графа G, ширины которых не больше w_1. Упорядочим ширины таких рёбер $w_1 > \ldots > w_s$. Обозначим через E_i множество рёбер, имеющих ширину не меньше w_i, и рассмотрим графы $G_i = (V, E_i, l, w)$ $(i = 1, \ldots, s)$.

Шаг 2. Для каждого графа G_i с помощью алгоритма A_L построим допустимый подграф H_i минимальной длины. Получим последовательность подграфов $H_1, \ldots, H_s$ таких, что $L(H_1) \geqslant \ldots \geqslant L(H_s)$. Удалим из этой последовательности каждый из тех подграфов, длина которых равна длине предыдущего подграфа. Оставшиеся подграфы являются результатом работы алгоритма и называются *итоговыми подграфами*, построенными алгоритмом A_0.

Теорема 1. *Итоговые подграфы графа $G = (V, E, l, w)$, построенные алгоритмом A_0, являются оптимальными по Парето и образуют систему различных представителей оптимальных по Парето подграфов.*

Доказательство. Пусть H_j — построенный алгоритмом A_0 итоговый подграф и (L_j, w_j) — его индикатор качества $(1 \leqslant j \leqslant s)$. Докажем оптимальность по Парето подграфа H_j. Пусть H' — допустимый подграф с индикатором (L', w') такой, что $L' \leqslant L_j$, $w' \geqslant w_j$. Если $w' = w_j$, то $L' \geqslant L_j$, так как подграф H_j имеет наименьшую длину среди всех допустимых подграфов ширины w_j. Таким образом, при $w' = w_j$ имеем $L' = L_j$. Пусть теперь $w' > w_j$. Так как w_1 — максимальная ширина допустимого подграфа графа G, то из неравенства $w' > w_j$ вытекает, что $j \geqslant 2$, $w' \geqslant w_{j-1}$ и $L(H') \geqslant L(H_{j-1})$. Но H_j — итоговый подграф и в соответствии с алгоритмом $L(H_{j-1}) > L(H_j)$. Следовательно, $L' = L(H') \geqslant L(H_{j-1}) > L(H_j) = L_j$, т. е. при $w' > w_j$ неравенство $L' \leqslant L_j$ невозможно. Значит, H_j — оптимальный по Парето подграф графа G.

Докажем теперь, что итоговые подграфы, построенные алгоритмом A_0, образуют систему различных представителей оптимальных по Парето подграфов. Пусть H — оптимальный по Парето подграф графа G. Его ширина $W(H)$ равна одному из чисел $w_1, \ldots, w_s$. Пусть $W(H) = w_k$. Рассмотрим подграф H_k. Имеем $W(H_k) = w_k = W(H)$. Так как H_k — подграф, имеющий минимальную длину среди всех допустимых подграфов с шириной w_k, то $L(H) \geqslant L(H_k)$, откуда в силу оптимальности по Парето подграфа H следует, что $L(H) = L(H_k)$. Осталось показать, что H_k — итоговый подграф, построенный алгоритмом A_0. Действительно, в противном случае имеем $L(H_{k-1}) = L(H_k)$, $W(H_{k-1}) = w_{k-1} > w_k$, т. е. $L(H_{k-1}) = L(H)$, $W(H_{k-1}) > W(H)$, что противоречит оптимальности по Парето подграфа H. Теорема 1 доказана.

Следствие 1. *Если G — граф с m рёбрами, то система различных представителей оптимальных по Парето подграфов содержит не более m подграфов.*

Оценим сложность алгоритма A_0. Для произвольного алгоритма A обозначим через $\tau(A, n, m)$ верхнюю границу сложности, которую имеет алгоритм в классе графов, число вершин которых не больше n, а число рёбер — не больше m. Считая, что, как обычно, формулы для сложности алгоритмов содержат мультипликативные константы и наиболее трудоёмкая часть каждого пункта алгоритма A_0 связана с применением алгоритмов A_w и A_L, получим

$$\tau(A_0, n, m) = \tau(A_W, n, m) + m \cdot \tau(A_L, n, m). \tag{1}$$

Распространённым подходом к решению многокритериальной задачи является сведéние её к однокритериальной задаче математического программирования. В нашем случае это сведéние состоит в следующем. Каждый допустимый подграф имеет индикатор качества (x, y), где x — длина подграфа, y — его ширина. Введём «штрафную» функцию $F(x, y)$, которая монотонно возрастает по первой переменной и монотонно убывает по второй. Допустимый подграф будем считать оптимальным, если его индикатор качества минимизирует функцию F. Ясно, что оптимальный в указанном смысле подграф будет оптимальным по Парето. Установим, какие рассматриваемые индикаторы качества, оптимальные по Парето, могут минимизировать линейную штрафную функцию.

Пусть штрафная функция имеет вид

$$f(x, y) = \alpha x - \beta y, \quad \alpha > 0,\ \beta > 0. \tag{2}$$

Обозначим через Ω систему различных представителей оптимальных по Парето подграфов. Если Ω содержит только один подграф, то его индикатор качества, очевидно, минимизирует функцию (2) при любых (положительных) α и β. Пусть теперь Ω состоит из подграфов $H_1, H_2, \ldots, H_k$, $k \geqslant 2$, с индикаторами качества $(x_1, y_1), (x_2, y_2), \ldots, (x_k, y_k)$ соответственно, причём

$$x_i > x_{i+1}, \quad y_i > y_{i+1} \quad (i = 1, \ldots, k-1). \tag{3}$$

Индикатор (x_q, y_q), $1 \leqslant q \leqslant k$, минимизирует функцию f тогда и только тогда, когда при всех $j \in [1, k] \setminus \{q\}$ коэффициенты α и β удовлетворяют неравенствам $\alpha x_q - \beta y_q \leqslant \alpha x_j - \beta y_j$. Запишем получившуюся систему неравенств в виде

$$\alpha(x_q - x_j) \leqslant \beta(y_q - y_j), \quad j \in [1, k] \setminus \{q\}. \tag{4}$$

Рассмотрим следующие случаи.

Случай 1. Пусть $q = 1$. Учитывая (3), систему (4) можно записать так: $\alpha/\beta \leqslant (y_1 - y_j)/(x_1 - x_j)$, $j \in [2, k]$. Поэтому при

$$\frac{\alpha}{\beta} \leqslant \min_{2 \leqslant j \leqslant k} \frac{y_1 - y_j}{x_1 - x_j}$$

индикатор (x_1, y_1) минимизирует функцию (2).

Случай 2. Пусть $q = k$. Тогда с учётом (3) система (4) записывается так: $\alpha/\beta \geqslant (y_k - y_j)/(x_k - x_j)$, $j \in [1, k-1]$. Поэтому при

$$\frac{\alpha}{\beta} \geqslant \max_{1 \leqslant j \leqslant k-1} \frac{y_k - y_j}{x_k - x_j}$$

индикатор (x_k, y_k) минимизирует функцию (2).

Случай 3. Пусть $1 < q < k$, $k \geqslant 3$. В этом случае система (4) записывается в виде двух подсистем неравенств: $\alpha/\beta \geqslant (y_q - y_j)/(x_q - x_j)$, $j = 1, \ldots, q-1$, и $\alpha/\beta \leqslant (y_q - y_j(/(x_q - x_j)$, $j = q+1, \ldots, k$. Поэтому индикатор качества (x_q, y_q) может минимизировать линейную функцию тогда и только тогда, когда

$$\max_{1 \leqslant j \leqslant q-1} \frac{y_q - y_j}{x_q - x_j} \leqslant \min_{q+1 \leqslant j \leqslant k} \frac{y_q - y_j}{x_q - x_j}.$$

2. Оптимальные остовы

Пусть $G = (V, E, l, w)$ — связный n-вершинный граф с m рёбрами. Будем считать, что допустимыми подграфами являются остовы графа G. Построить минимальный по длине остов графа можно, например, с помощью алгоритма Краскала [5]. Таким образом, алгоритм A_L для рассматриваемого случая известен. Чтобы применить алгоритм A_0, нужен алгоритм, определяющий максимальную ширину остова. Для решения этой задачи можно предложить два алгоритма. Один из них состоит в том, что рёбра графа упорядочиваются по неубыванию их ширин, а затем удаляются по очереди до тех пор, пока граф остаётся связным. Ширина первого ребра, которое удалить нельзя, и будет максимальной шириной остова. Другой алгоритм с оценкой сложности опишем подробнее.

Алгоритм A_1.

Шаг 0. Упорядочим рёбра графа по невозрастанию их ширин: $e_1, e_2, \ldots, e_m$. Удалив все рёбра, получим пустой граф G_0. Всем вершинам графа G_0 ставим в соответствие различные числовые метки.

Шаг k, $1 \leqslant k \leqslant m$. Пусть уже построен лес G_{k-1}, вершинам которого поставлены в соответствие числовые метки таким образом, что вершины имеют одинаковые метки тогда и только тогда, когда они принадлежат одной компоненте связности. Рассматриваем ребро e_k. Пусть оно инцидентно вершинам x и y, которые в графе G_{k-1} имеют метки i и j соответственно. Если $i = j$, то полагаем $G_k = G_{k-1}$. Если же $i \neq j$, то добавляем к графу G_{k-1} ребро e_k, метки тех вершин, которые в графе G_{k-1} были равны i или j, полагаем равными $\min(i, j)$. Полученный граф обозначаем G_k. Затем в случае $k < m$ выполняем шаг $k+1$, в противном случае — останов.

Ширина последнего ребра, включённого в остов G_m, будет, очевидно, максимальной шириной остова графа G.

Процедура упорядочения m элементов требует $O(m \log m)$ сравнений [1]. Поэтому алгоритм A_1 имеет сложность $O(m \log m)$. Такую же сложность имеет алгоритм Краскала [1]. По формуле (1) получаем, что алгоритм A_0, решающий задачу построения системы различных представителей оптимальных по Парето остовов графа, имеет сложность $O(m^2 \log m)$.

3. Оптимальная цепь между вершинами графа

Пусть $G = (V, E, l, w)$ — n-вершинный граф с m рёбрами, a и b — две его вершины, принадлежащие одной компоненте связности. Всякую простую цепь, соединяющую вершины a и b, будем считать допустимым подграфом.

Для построения цепи минимальной длины, соединяющей вершины a и b, можно использовать алгоритм Дийкстры [4] сложности $O(n^2)$.

Изложим алгоритм для определения максимальной ширины простой цепи, соединяющей вершины a и b. В ходе работы алгоритма вершинам графа, отличным от вершины a, будут ставиться в соответствие числовые временные или постоянные метки; временная метка для вершины v обозначается через $t(v)$, постоянная — через $c(v)$. Содержательный смысл этих меток таков. Временная метка равна 0 до тех пор, пока не выявлено ни одной цепи, соединяющей вершину a с вершиной v; если же такие цепи выявлены, то временная метка — это наибольшая ширина из таких цепей. Временная метка становится постоянной, когда выясняется, что цепь, соединяющая вершину a с вершиной v, имеет максимально возможную ширину.

Алгоритм A_2.

Шаг 1. Всем вершинам графа, кроме вершины a, ставим метку 0. Пусть $\Gamma(a)$ — множество вершин графа, смежных с вершиной a. Для

каждого $x \in \Gamma(a)$ полагаем $t(x) = w(a, x)$, где $w(a, x)$ — ширина ребра (a, x). Отыскиваем вершину с наибольшей временной меткой и делаем эту метку постоянной.

Шаг 2 (общий шаг). Рассматриваем вершину, которая на предыдущем шаге получила постоянную метку. Пусть это вершина p. Обозначим через $\Gamma'(p)$ множество вершин графа, смежных с p и имеющих временную метку. Для каждой вершины $y \in \Gamma'(p)$ (если $\Gamma'(p) \neq \varnothing$) пересчитываем временную метку так: $t(y) = \max\{t(y), \min\{c(p), w(p, y)\}\}$. Отыскиваем вершину с наибольшей временной меткой и делаем эту метку постоянной.

Алгоритм останавливается, когда вершина b получит постоянную метку. Эта метка и будет максимальной шириной цепи, соединяющей a и b.

Корректность алгоритма очевидным образом следует из тех пояснений относительно содержательного смысла временных и постоянных меток, которые приводились перед изложением алгоритма. Далее, так как для n-вершинного графа общий шаг алгоритма требует $O(n)$ действий, то сложность алгоритма A_2 равна $O(n^2)$.

Теперь, используя алгоритмы A_2 и Дийкстры, можно с помощью алгоритма A_0 построить систему различных представителей оптимальных по Парето цепей, соединяющих вершины a и b. По формуле (1) находим, что сложность такого алгоритма равна $O(mn^2)$.

ЛИТЕРАТУРА

1. **Ахо А., Хопкрофт Дж., Ульман Дж.** Построение и анализ вычислительных алгоритмов. — М.: Мир, 1979. — 536 с.
2. **Зыков А. А.** Основы теории графов. — М.: Вузовская книга, 2004. — 663 с.
3. **Кофман А.** Введение в теорию нечётких множеств. — М.: Радио и связь, 1982. — 432 с.
4. **Dijkstra E. W.** A note on two problems in connection with graphs // Numer. Math. — 1959. — Vol. 1. — P. 269–271.
5. **Kruskal J. B., Jr.** On the shortest spanning subtree of a graph and the traveling salesman problem //Proc. Amer. Math. Soc. — 1956. — Vol. 7, N 1. — P. 48–50.

Визинг Вадим Георгиевич,
e-mail: vizing@paco.net

Статья поступила
23 апреля 2009 г.