ДИСКРЕТНЫЙ АНАЛИЗ И ИССЛЕДОВАНИЕ ОПЕРАЦИЙ
Июль—сентябрь 2003. Серия 1. Том 10, № 3, 54–66

УДК 519.8

ДВЕ ЗАДАЧИ НА НАСЛЕДСТВЕННЫХ СИСТЕМАХ

В. П. Ильев, А. С. Талевнин

Исследуются задачи о максимальном независимом и минимальном зависимом множестве наследственной системы, которые могут быть рассмотрены как задачи о максимальном независимом множестве вершин и минимальном вершинном покрытии в гиперграфе соответственно. Для приближенного решения невзвешенной задачи о независимом множестве предложен алгоритм градиентного типа. В предположении, что гиперграф не содержит ребер мощности 1, доказано, что этот алгоритм всегда дает решение, которое не более чем в $(\overline{d}+2)/2$ раз хуже оптимального, где $\overline{d}$ – средняя степень вершин гиперграфа. Показана эквивалентность задачи о минимальном зависимом множестве задаче о покрытии множества, что позволяет применить для ее решения известный алгоритм Хватала. Этот алгоритм находит решение, отличающееся от оптимального не более чем в $1 + \ln \Delta$ раз, где Δ — максимальная степень вершин гиперграфа.

Введение

Пусть V — конечное множество. *Системой независимости* на V называется пара $(V, \mathcal{A})$, где $\mathcal{A} \subseteq 2^V$ — непустое семейство подмножеств множества V такое, что если

$$A \in \mathcal{A} \text{ и } A' \subseteq A, \text{ то } A' \in \mathcal{A}$$

(*аксиома наследственности*). Множества семейства $\mathcal{A}$ называются *независимыми*, а все подмножества из $2^V \setminus \mathcal{A}$ — *зависимыми*. Семейство всех зависимых множеств обозначим через $\mathcal{D}$. Легко видеть, что $\mathcal{D}$ обладает свойством *наследственности «вверх»*: если

$$D \in \mathcal{D} \text{ и } D \subseteq D', \text{ то } D' \in \mathcal{D}.$$

Очевидно, что каждое из семейств $\mathcal{A}$, $\mathcal{D}$ однозначно определяет другое. Поэтому естественно рассматривать их как различные представления одного и того же объекта наследственной системы.

Определим *наследственную систему* $\mathcal{S}$ на множестве V как разбиение семейства 2^V всех подмножеств множества V на два непересекающихся подсемейства $\mathcal{A}$ и $\mathcal{D}$, где $(V, \mathcal{A})$ — система независимости, а $\mathcal{D} \subseteq 2^V$. *Базами* системы $\mathcal{S}$ называются максимальные по включению независимые, а *циклами* — минимальные по включению зависимые множества. Семейства всех баз и всех циклов системы $\mathcal{S}$ будем обозначать через $\mathcal{B}$ и $\mathcal{C}$ соответственно. Очевидно, что как семейство $\mathcal{B}$, так и семейство $\mathcal{C}$, однозначно определяет наследственную систему $\mathcal{S}$. Будем записывать $\mathcal{S} = (V, \mathcal{A})$, $\mathcal{S} = (V, \mathcal{B})$, $\mathcal{S} = (V, \mathcal{C})$ или $\mathcal{S} = (V, \mathcal{D})$ в зависимости от того, какая информация о наследственной системе нас интересует в данный момент.

С каждой наследственной системой $\mathcal{S} = (V, \mathcal{A})$ тесно связана *дополнительная система* $\mathcal{S}'$, семейство зависимых множеств которой определяется как $\mathcal{D}' = \{V \setminus A \mid A \in \mathcal{A}\}$. Очевидно, что семейства независимых множеств, баз и циклов системы $\mathcal{S}'$ могут быть заданы как $\mathcal{A}' = \{V \setminus D \mid D \in \mathcal{D}\}$, $\mathcal{B}' = \{V \setminus C \mid C \in \mathcal{C}\}$, $\mathcal{C}' = \{V \setminus B \mid B \in \mathcal{B}\}$. Ясно, что $(\mathcal{S}')' = \mathcal{S}$, т. е. наследственные системы $\mathcal{S}$ и $\mathcal{S}'$ взаимно дополнительны.

Кроме того, для каждой наследственной системы $\mathcal{S} = (V, \mathcal{B}) = (V, \mathcal{C})$ можно определить два типа *двойственных систем* $\mathcal{S}^* = (V, \mathcal{B}^*)$ и $\mathcal{S}^o = (V, \mathcal{C}^o)$ по правилам:

$$\mathcal{B}^* = \{V \setminus B \mid B \in \mathcal{B}\}, \;\; \mathcal{C}^o = \{V \setminus C \mid C \in \mathcal{C}\}.$$

Нетрудно показать, что $\mathcal{S} = (((\mathcal{S}^*)')^o)' = (((\mathcal{S}^o)')^*)'$.

Заметим, что двойственность систем $\mathcal{S}$ и $\mathcal{S}^*$ является обобщением классической двойственности матроидов, введенной Уитни [6].

Пример 1. Наследственная система графа и дополнительная к ней.

Рассмотрим произвольный (связный) граф $G = (V, E)$ и наследственную систему $\mathcal{S}_G = (V, \mathcal{A}_G)$, где $\mathcal{A}_G$ — семейство всех независимых множеств вершин графа G. Тогда $\mathcal{B}_G$ есть семейство всех максимальных по включению независимых множеств вершин. Легко видеть, что любой цикл системы $\mathcal{S}_G$ имеет мощность 2 и соответствует ребру графа G. Известно, что вершинными покрытиями в графе являются дополнения независимых множеств и только они. Поэтому семейство $\mathcal{D}'_G$ всех вершинных покрытий в произвольном графе $G = (V, E)$ можно рассматривать как семейство зависимых множеств наследственной системы $\mathcal{S}'_G = (V, \mathcal{D}'_G)$, дополнительной к наследственной системе графа. Циклами системы $\mathcal{S}'_G$ являются минимальные по включению вершинные покрытия, а базами — максимальные непокрытия, т. е. дополнения ребер

графа G. Заметим, что система $\mathcal{S}'_G$ обычно задается с помощью графа G, например, перечнем его ребер, что равносильно заданию наследственной системы $\mathcal{S}'_G$ с помощью семейства баз $\mathcal{B}'_G$.

В настоящей статье рассматриваются следующие две задачи комбинаторной оптимизации на наследственных системах.

Пусть $c : V \to R_+$ — произвольная аддитивная функция.

Задача 1 (*о максимальном независимом множестве*). Дана наследственная система $\mathcal{S} = (V, \mathcal{C})$. Найти такое $X^* \in \mathcal{A}$, что

$$c(X^*) = \max\{c(X) \mid X \in \mathcal{A}\} = \max\{c(X) \mid X \in \mathcal{B}\}.$$

Задача 2 (*о минимальном зависимом множестве*). Дана наследственная система $\mathcal{S} = (V, \mathcal{B})$. Найти такое $X^* \in \mathcal{D}$, что

$$c(X^*) = \min\{c(X) \mid X \in \mathcal{D}\} = \min\{c(X) \mid X \in \mathcal{C}\}.$$

Многие сложные в вычислительном отношении задачи комбинаторной оптимизации являются частными случаями задач 1 и 2. Например, если $\mathcal{S} = \mathcal{S}_G$ — наследственная система графа, то задача 1 является задачей о максимальном независимом множестве вершин графа, а если $\mathcal{S} = \mathcal{S}'_G$ — дополнительная наследственная система графа, то задача 2 становится задачей о минимальном вершинном покрытии графа.

Произвольной наследственной системе $\mathcal{S} = (V, \mathcal{C})$ поставим в соответствие гиперграф $H = (V, E)$, ребра которого взаимно однозначно соответствуют циклам этой системы. Наоборот, любой гиперграф H порождает наследственную систему $\mathcal{S}_H = (V, \mathcal{C}_H)$, циклами которой являются минимальные по включению ребра гиперграфа H. Поэтому наследственную систему можно отождествлять с гиперграфом, в котором никакое ребро не содержит другое ребро в качестве подмножества.

Тогда, полагая $\mathcal{S} = \mathcal{S}_H$ в задаче 1, получаем задачу о максимальном независимом множестве вершин гиперграфа H. Если же положить $\mathcal{S} = \mathcal{S}'_H$, то задача 2 может быть рассмотрена как задача о минимальном вершинном покрытии ребер того же гиперграфа H. Подробнее об этом будет сказано в §§ 1 и 2.

Заметим, что в задаче 1 входом является семейство циклов, а в задаче 2 — семейство баз наследственной системы $\mathcal{S}$. В силу вышесказанного в обеих задачах входом можно считать гиперграф H, ребра которого взаимно однозначно соответствуют циклам системы $\mathcal{S} = \mathcal{S}_H$ в задаче 1 и базам системы $\mathcal{S} = \mathcal{S}'_H$ в задаче 2.

Уже в невзвешенном случае задачи 1 и 2 являются NP-трудными. Кроме того, в работе [5] доказано, что в весьма широком классе алгорит-

мов с оракулами для задачи 1, включающем почти все известные комбинаторные эвристики, не существует полиномиального по $|V|$ алгоритма. Целью настоящей статьи является выяснение взаимосвязей между задачами 1 и 2, а также получение гарантированных оценок погрешности приближенных алгоритмов их решения.

В § 1 для решения невзвешенной задачи 1 предложен вариант жадного алгоритма. В предположении, что H не содержит ребер мощности 1, доказано, что этот алгоритм всегда дает решение, которое не более чем в $(\overline{d}+2)/2$ раз хуже оптимального, где $\overline{d}$ — средняя степень вершин в гиперграфе H.

В § 2 показано, что задача 2 эквивалентна задаче о покрытии множества. Это позволяет применить для решения задачи 2 известный алгоритм Хватала, который находит решение задачи 2, отличающееся от оптимального не более чем в $1+\ln\Delta$ раз, где Δ — максимальная степень вершин в гиперграфе H.

§ 1. Оценка погрешности алгоритма решения задачи о наибольшем независимом множестве

Гиперграфом называется пара $H = (V, E)$, где $V = V(H)$ — непустое конечное множество, а $E = E(H)$ — семейство подмножеств множества V. Элементы множества V называются *вершинами*, а элементы семейства E — *ребрами* гиперграфа H.

Множество вершин гиперграфа называется *независимым*, если никакое ребро гиперграфа не является его подмножеством.

Рассмотрим следующую задачу.

Задача $\mathbf{1^0}$ (*о наибольшем независимом множестве вершин*). Дан гиперграф $H = (V, E)$. Найти такое независимое множество вершин $U \subseteq V$, что

$$|U| = \max\{|W| \mid W \text{ независимо}\}.$$

Очевидно, что при удалении из H ребра, содержащего другое ребро как подмножество, семейство допустимых решений этой задачи не изменяется. Поэтому далее будем рассматривать только гиперграфы, в которых ни одно ребро не содержит другое в качестве подмножества. Напомним, что любой такой гиперграф H порождает наследственную систему $\mathcal{S}_H$, в которой независимыми множествами являются независимые множества вершин H. Поэтому задача 1^0 о наибольшем независимом множестве вершин гиперграфа H — это невзвешенная задача 1 на системе $\mathcal{S}_H$.

Полиномиальный алгоритм называется *ρ-приближенным*, если для рассматриваемой задачи оптимизации он находит допустимое решение, на котором значение целевой функции не более чем в ρ раз хуже оптимального значения. Величина ρ называется *гарантированной оценкой погрешности* этого алгоритма.

В работе [3] для задачи 1, которую можно рассматривать как взвешенную задачу о максимальном независимом множестве вершин гиперграфа, предложен $|V|/\log|V|$-приближенный алгоритм. В случае, когда H является графом, известен $|V|/\log^2|V|$-приближенный алгоритм [1], а один из вариантов жадного алгоритма является одновременно $(\overline{d}+2)/2$- и $(\Delta+2)/3$-приближенным алгоритмом решения задачи 1^0 на графе [4], где $\overline{d}$ и Δ — средняя и максимальная степени вершин в графе соответственно.

Далее нам потребуются следующие понятия и обозначения. Будем говорить, что $e \in E$ является *k-ребром* гиперграфа H, если $|e| = k$, и обозначим через E_k множество всех его k-ребер. Для каждой вершины $v \in V$ множество *инцидентных* ей ребер обозначим через $E(v)$, т. е. $E(v) = \{e \in E \mid v \in e\}$. Введем понятия *степени* и *k-степени* вершины $v \in V$ соответственно как $d(v) = |E(v)|$ и $d_k(v) = |E(v) \bigcap E_k|$. Для каждой вершины $v \in V$ через $N(v) = \{u \in V \mid uv \in E_2\}$ обозначим множество вершин, *смежных* с v. *Среднюю степень* $\overline{d}(H)$ вершин в гиперграфе H определим как $\sum\limits_{v \in V(H)} d(v)/|V(H)|$.

Для приближенного решения задачи 1^0 предлагается жадный алгоритм, которому предшествует процедура предварительного упрощения гиперграфа. Этот алгоритм является обобщением на случай гиперграфов известного жадного алгоритма для графов [4], который на каждой итерации выбирает вершину минимальной степени и удаляет ее вместе со всеми смежными ей вершинами.

Процедура S(H)

1. Любые два ребра e_1, e_2 такие, что $|e_1 \bigcap e_2| \geqslant 2$, заменяются одним 2-ребром, соединяющим произвольную пару вершин из $e_1 \bigcap e_2$.

2. Любое ребро e такое, что $|e| > 3$, заменяется 2-ребром, соединяющим произвольную пару вершин из e.

3. Если существуют ребра $e_1 = \{v, v_1, v_2\}$, $e_2 = \{v, v_3, v_4\}$ (т. е. $e_1 \bigcap e_2 = \{v\}$), то они заменяются на 2-ребра $e_1' = \{v, v_1\}$ и $e_2' = \{v_3, v_4\}$.

Жадный алгоритм Gr(H)

$\mathbf{S}(H)$;

$S_G \leftarrow \emptyset$;

Пока $H \neq \emptyset$ выполнять

Шаг 1. Выбрать вершину v такую, что

$$d_2(v) + d(v) = \min_{u \in V(H)} \big(d_2(u) + d(u)\big);$$

Шаг 2. $S_G \leftarrow S_G \cup \{v\}$;

Шаг 3. $e \leftarrow e \setminus \{v\}$ для всех $e \in E(v) \bigcap E_3$;

Шаг 4. $H \leftarrow H \setminus (\{v\} \bigcup N(v))$;

конец цикла;

конец алгоритма.

Замечание 1. Любое из преобразований вида 1, 2, 3 процедуры **S** приводит к убыванию суммы степеней всех вершин гиперграфа не менее чем на 2. При этом для преобразования вида 3 обе концевые вершины хотя бы одного нового 2-ребра не лежат в оптимальном решении.

Замечание 2. В процедуре **S** и на шаге 3 алгоритма **Gr** возможно появление ребер, все вершины которых содержатся в S_O, где S_O – оптимальное решение. В этом случае такие ребра будем называть *мнимыми* относительно оптимального решения S_O.

Замечание 3. На шаге 4 алгоритма **Gr** из гиперграфа H удаляется вершина v вместе со всеми смежными ей вершинами, а также все ребра, инцидентные этим вершинам.

Правило выбора вершины на шаге 1 можно заменить на следующее (по аналогии с [4]): выбрать такую вершину v, что

$$d_2(v) + d(v) \leqslant \sum_{u \in N(v)} \frac{d_2(u) + d(u)}{d_2(v)}.$$

Очевидно, что вершина v с минимальным значением суммы $d_2(v) + d(v)$ удовлетворяет этому условию.

Теперь установим оценку погрешности алгоритма **Gr**, используя технику, аналогичную технике из [4].

Теорема 1. *Пусть S_O — оптимальное решение задачи 1^0 и S_G — решение, полученное алгоритмом* **Gr**. *Тогда*

$$|S_O| \leqslant \frac{\overline{d} + 2}{2} |S_G|. \tag{1}$$

Доказательство. Сначала докажем неравенство

$$|S_G| \geqslant \frac{(1+\tau^2)n}{\overline{d}+1+\tau}, \tag{2}$$

где $n = |V(H)|$, а $\tau = |S_O|/n$.

Рассмотрим исходный гиперграф $H = (V, E)$, который подается на вход процедуры **S**. Фиксируем некоторое оптимальное решение S_O в гиперграфе H и пусть $\alpha = |S_O|$. Обозначим через $I = \{1, \dots, t\}$ множество номеров итераций главного цикла алгоритма **Gr**. Пусть H_i — гиперграф, полученный к началу i-й итерации, $i \in I$ (H_1 — гиперграф, полученный из H на выходе процедуры **S**).

Пусть v_i — вершина гиперграфа H_i, выбранная на i-й итерации алгоритма **Gr**, а k_i — число вершин, входящих в S_O и удаленных на i-й итерации главного цикла, $i \in I$. Обозначим через x_i число таких 2-ребер гиперграфа H_i, что каждое из них инцидентно ровно одной вершине из $\{v_i\} \bigcup N(v_i)$; через y_i число таких 2-ребер H_i, что каждое из них инцидентно двум вершинам из $\{v_i\} \bigcup N(v_i)$, причем хотя бы одна вершина не принадлежит S_O; а через w_i число мнимых относительно S_O 2-ребер (см. замечание 2), каждое из которых инцидентно двум вершинам множества $\{v_i\} \bigcup N(v_i)$. Пусть $w_i = w_i' + \sum(w_i^j \mid j \in I,\ j < i)$, где w_i' — число 2-ребер среди ребер w_i, полученных после применения к H процедуры **S**, а w_i^j — число 2-ребер среди ребер w_i, появившихся после выполнения шага 3 на j-й итерации алгоритма ($i,\ j \in I$). Через f_{il} обозначим число 3-ребер в H_i, каждое из которых имеет в пересечении с $N(v_i)$ ровно l вершин ($l \in \{1, 2, 3\}$). Тогда

$$d_2(v_i) + \sum_{u \in N(v_i)} \frac{d(u) + d_2(u)}{2} = x_i + 2(y_i + w_i) + \frac{1}{2}\sum_{l=1}^{3} l \cdot f_{il}.$$

В силу критерия выбора вершины v_i алгоритмом **Gr** имеем

$$\sum_{u \in N(v_i)} \bigl(d(u) + d_2(u)\bigr) \geqslant d_2(v_i)\bigl(d(v_i) + d_2(v_i)\bigr).$$

Поэтому

$$x_i + 2(y_i + w_i) + \frac{1}{2}\sum_{l=1}^{3} l \cdot f_{il} \geqslant d_2(v_i) + d_2(v_i)\frac{d(v_i) + d_2(v_i)}{2}.$$

Так как для любого $i \in I$ в гиперграфе H_i нет ребер мощности более 3 (см. пункт 2 процедуры **S**), то $d(v_i) - d_2(v_i) = d_3(v_i)$. Учитывая это,

предыдущее неравенство перепишем в виде

$$x_i + 2(y_i + w_i) + \frac{1}{2}\sum_{l=1}^{3} l \cdot f_{il} \geqslant d_2(v_i)(d_2(v_i)+1) + \frac{d_2(v_i)}{2}d_3(v_i). \qquad (3)$$

Для y_i справедливо неравенство $2y_i \leqslant d_2(v_i)(d_2(v_i)+1) - k_i(k_i-1)$. Умножив его на –2 и сложив с неравенством (3), умноженным на 2, получим

$$\begin{aligned} &2(x_i + y_i + w_i) + 2w_i + \sum_{l=1}^{3} l \cdot f_{il} \\ &\geqslant d_2(v_i)(d_2(v_i)+1) + k_i(k_i-1) + d_2(v_i)d_3(v_i). \end{aligned} \qquad (4)$$

Пусть теперь $\delta_i = \sum\limits_{u \in H_i} d(u) - \sum\limits_{u \in H_{i+1}} d(u)$. Тогда

$$\delta_i \geqslant \sum_{j>i} w_j^i + 2(x_i + y_i + w_i) + 3\sum_{l=1}^{3} f_{il}$$

(см. шаг 3 алгоритма **Gr** и замечание 3). Поэтому для каждого $i \in I$ имеем

$$\delta_i - \sum_{j>i} w_j^i \geqslant 2(x_i + y_i + w_i) + \sum_{l=1}^{3} l \cdot f_{il}.$$

Используя это неравенство при суммировании (4) по $i \in I$, получаем

$$\begin{aligned} &\sum_{i \in I}\Big(\delta_i + 2w_i - \sum_{j>i} w_j^i\Big) \\ &\geqslant \sum_{i \in I}\Big(d_2(v_i)(d_2(v_i)+1) + k_i(k_i-1) + d_2(v_i)d_3(v_i)\Big). \end{aligned}$$

Так как $w_i = w_i' + \sum\limits_{s<i} w_i^s$, то

$$\begin{aligned} &\sum_{i \in I}\big(\delta_i + 2w_i - \sum_{j>i} w_j^i\big) = \sum_{i \in I}\big(\delta_i + 2w_i' + \sum_{s<i} w_i^s\big) \\ &+ \sum_{i \in I}\big(\sum_{s<i} w_i^s - \sum_{j>i} w_j^i\big) = \sum_{i \in I}\big(\delta_i + 2w_i' + \sum_{s<i} w_i^s\big). \end{aligned}$$

Значение $\sum\limits_{i \in I}(\delta_i + 2w_i') = 2\sum\limits_{i \in I} w_i' + \sum\limits_{u \in H_1} d(u)$ не превосходит (в силу определения w_i' и замечания 1) величины $\overline{d}n$. Следовательно,

$$\overline{d}n \geqslant \sum_{i \in I}\big(d_2(v_i)(d_2(v_i)+1) + k_i(k_i-1) + d_2(v_i)d_3(v_i) - \sum_{s<i} w_i^s\big). \qquad (5)$$

Пусть множество $I_0 \subset I$ таково, что при любом $i \in I_0$ на шаге 3 итерации i образуется мнимое ребро и $d_2(v_i) = 0$. Пусть $I_1 = I \setminus I_0$. Тогда

$$\sum_{i \in I_1} (d_2(v_i) + 1) = n - |I_0|, \quad \sum_{i \in I_1} k_i = \alpha. \tag{6}$$

Ввиду того, что $\sum\limits_{i \in I} \sum\limits_{s<i} w_i^s = \sum\limits_{i \in I} \sum\limits_{j>i} w_j^i$, неравенство (5) можно записать в виде

$$\overline{d}n \geqslant \sum_{i \in I_1} \Big(d_2(v_i)(d_2(v_i)+1) + k_i(k_i-1) \Big) + \sum_{i \in I_1} \Big(d_2(v_i) d_3(v_i) - \sum_{j>i} w_j^i \Big) - |I_0|,$$

где $|I_0| \geqslant \sum\limits_{i \in I_0} \sum\limits_{j>i} w_j^i$, поскольку при любом $i \in I$ в H_i нет вершины с более чем одним инцидентным 3-ребром (см. пункт 3 процедуры **S**), т. е. $\sum\limits_{j>i} w_j^i \leqslant 1$. Так как при $i \in I_1$ справедливо неравенство

$$d_2(v_i) d_3(v_i) - \sum_{j>i} w_j^i \geqslant 0,$$

то

$$\overline{d}n + |I_0| \geqslant \sum_{i \in I_1} \Big(d_2(v_i)(d_2(v_i)+1) + k_i(k_i-1) \Big). \tag{7}$$

Используя неравенство Йенсена для выпуклой функции $g(x) = x^2 + x$ и равенства (6), получаем

$$\sum_{i \in I_1} d_2(v_i)(d_2(v_i)+1) \geqslant \sum_{i \in I_1} d_2(v_i) \Big(\sum_{i \in I_1} \frac{d_2(v_i)}{|I_1|} + 1 \Big) = \frac{(n - |I_0| - |I_1|)(n - |I_0|)}{|I_1|}$$

и

$$\sum_{i \in I_1} k_i(k_i - 1) \geqslant \sum_{i \in I_1} k_i \Big(\sum_{i \in I_1} \frac{k_i}{|I_1|} - 1 \Big) = \frac{\alpha(\alpha - |I_1|)}{|I_1|}.$$

Таким образом, из (7) следует, что

$$\overline{d}n + |I_0| \geqslant \frac{(n - |I_0|)^2 + \alpha^2}{|I_1|} - (n - |I_0|) - \alpha.$$

Преобразуя последнее неравенство, получаем

$$|I_1| \geqslant \frac{n^2 + \alpha^2 + |I_0|^2 - 2n|I_0|}{\overline{d}n + n + \alpha}.$$

Поэтому

$$|I| \geqslant \frac{n^2 + \alpha^2 + |I_0|^2 - 2n|I_0|}{\overline{d}n + n + \alpha} + |I_0|,$$

или

$$|I| \geqslant \frac{n^2+\alpha^2+|I_0|(\overline{d}n+|I_0|-n+\alpha)}{\overline{d}n+n+\alpha}.$$

Так как любая вершина, не принадлежащая оптимальному решению, имеет хотя бы одно инцидентное ей ребро, то $\overline{d}n-(n-\alpha) \geqslant n-\alpha \geqslant 0$. Следовательно,

$$|I| \geqslant \frac{n^2+\alpha^2}{\overline{d}\cdot n+n+\alpha}.$$

Разделив числитель и знаменатель на n^2, получим неравенство (2).

Вспомнив, что $|S_O|=\alpha=\tau n$ и используя (2), получаем

$$\frac{\tau n}{|S_G|} \leqslant \frac{\tau(\overline{d}+1+\tau)}{1+\tau^2}=g(\tau).$$

Поскольку $\tau \in [0,1]$ и функция $g(\tau)$ возрастает, ее максимум достигается при $\tau=1$. Таким образом, $|S_O|/|S_G| \leqslant g(1)=(\overline{d}+2)/2$.
Теорема 1 доказана.

Замечание 4. Оценка (1) достижима. Как показано в работе [4], существуют серии таких графов G, что

$$\frac{|S_O|}{|S_G|} \geqslant \frac{\overline{d}+2}{2}-O(1/\overline{d}).$$

§ 2. Оценка погрешности алгоритма решения задачи о минимальном зависимом множестве

В этом параграфе рассмотрим задачу 2 как задачу о минимальном вершинном покрытии ребер гиперграфа.

Множество вершин гиперграфа H, имеющее непустое пересечение с каждым ребром, называется *вершинным покрытием*.

Отметим, что, удаляя из H ребро, содержащее другое ребро как подмножество, мы не изменим семейство вершинных покрытий H. Поэтому будем рассматривать гиперграфы, в которых ни одно ребро не содержит другое в качестве подмножества. В этом случае дополнения ребер гиперграфа H являются базами наследственной системы $\mathcal{S}'_H$. Следовательно, любое вершинное покрытие гиперграфа H является зависимым множеством системы $\mathcal{S}'_H$. Значит, задача 2 о минимальном зависимом множестве $\mathcal{S}'_H$ есть задача о минимальном вершинном покрытии H.

Покажем, что задача 2 эквивалентна задаче о покрытии множества.

Задача 3 (*о покрытии множества*). Пусть J — конечное множество, $c : J \to R_+$ — аддитивная весовая функция. Задано семейство конечных множеств $\{P_j \mid j \in J\}$, $I = \bigcup_{j \in J} P_j$. Найти множество $J^* \subseteq J$ минимального веса такое, что $I = \bigcup_{j \in J^*} P_j$.

Для любой вершины $v \in V$ обозначим через P_v множество ребер гиперграфа $H = (V, E)$, которым инцедентна вершина v, т. е. $J = V$. Получим задачу о покрытии множества E множествами семейства $\{P_v \mid v \in V\}$. Таким образом, задача 2 на системе $\mathcal{S}'_H$ может быть сформулирована как задача 3 о покрытии множества ребер гиперграфа H.

Заметим, что обратное утверждение тоже верно, т. е. всякая задача 3 о покрытии множества может быть рассмотрена как задача 2 о минимальном зависимом множестве наследственной системы. Однако чтобы определить семейство баз соответствующей наследственной системы, нужно удалить, в случае необходимости, некоторые элементы покрываемого множества I (смотри пример 2 ниже). Полученная таким образом задача уже может быть рассмотрена как задача о минимальном вершинном покрытии гиперграфа, а значит, как задача 2.

Пример 2. Рассмотрим невзвешенную задачу о покрытии множества $I = \{a, b, c\}$ множествами $P_1 = \{a, b\}, P_2 = \{a, b, c\}, P_3 = \{b, c\}$, $P_4 = \{c\}$, т. е. $J = \{1, 2, 3, 4\}$. Обозначим через D_a множество индексов тех P_j, которые содержат элемент a. Аналогично определим множества D_b и D_c: $D_a = \{1, 2\}, D_b = \{1, 2, 3\}, D_c = \{2, 3, 4\}$. Семейство $\{D_a, D_b, D_c\}$ можно рассматривать как множество ребер некоторого гиперграфа H_0 на множестве вершин $V = J = \{1, 2, 3, 4\}$. Удалив ребро D_b, содержащее ребро D_a, получим гиперграф H. Удаление ребра D_b можно понимать как удаление элемента b из I. Заметим, что любое вершинное покрытие ребер гиперграфа H является также покрытием ребер гиперграфа H_0 и, следовательно, наименьшее вершинное покрытие в гиперграфе H будет наименьшим вершинным покрытием в гиперграфе H_0. Ребра D_a, D_c гиперграфа H взаимно однозначно соответствуют циклам некоторой наследственной системы $\mathcal{S}_H = (V, \mathcal{C}_H)$. Тогда дополнения циклов $B'_a = \{3, 4\}, B'_c = \{1\}$ будут базами системы $\mathcal{S}'_H = (V, \{B'_a, B'_c\})$. Получаем задачу 2 на наследственной системе $\mathcal{S}'_H$.

В общем случае пусть $D_i = \{j \in J \mid i \in P_j\}$. Семейство $\{D_i \mid i \in I\}$ рассмотрим как множество ребер гиперграфа H_0, множество вершин которого $V = J$. Удалим из H_0 все ребра D_i, содержащие другие в качестве

подмножеств, что равносильно удалению из множества I соответствующих элементов i. Полученный в результате гиперграф обозначим через H. Аддитивную функцию на множестве V положим тождественно равной функции $c : J \to R_+$ из задачи 3. Тогда задача 3 о покрытии множества становится задачей о вершинном покрытии гиперграфа H, которая эквивалентна задаче 2.

Если задачу о покрытии множества понимать как задачу о покрытии множества строк булевой матрицы множеством столбцов, то эта матрица будет матрицей инциденций гиперграфа H_0, ее столбцы соответствуют вершинам, а строки — ребрам гиперграфа H_0.

Из сказанного следует, что задачи 2 и 3 эквивалентны. Это позволяет применить для решения задачи 2 на наследственной системе $\mathcal{S} = (V, \mathcal{B}) = \mathcal{S}'_H$ известный алгоритм Хватала [2]. Обозначим через $\mathcal{B}_v$ семейство баз системы $\mathcal{S}$, не содержащих элемент $v \in V$.

В терминах задачи 2 алгоритм Хватала может быть описан следующим образом.

Алгоритм Ch для задачи 2

$S_C \leftarrow \emptyset$;
Пока $\mathcal{B}_v \neq \emptyset$ выполнять
  Шаг 1. Выбрать вершину u такую, что

$$\frac{c_u}{|\mathcal{B}_u|} = \min_{\mathcal{B}_v \neq \emptyset} \frac{c_v}{|\mathcal{B}_v|};$$

  Шаг 2. $S_C \leftarrow S_C \cup \{u\}$;
  Шаг 3. $\mathcal{B}_v \leftarrow \mathcal{B}_v \setminus \mathcal{B}_u \ \forall v \in V$.
конец цикла;
конец алгоритма.

Из эквивалентности задач 2 и 3 вытекает следующая гарантированная оценка погрешности алгоритма **Ch** для задачи 2.

Теорема 2. *Пусть S_O — оптимальное решение задачи 2, S_C — решение, полученное алгоритмом* **Ch**, *и $\Delta = \max_{v \in V} |\mathcal{B}_v|$. Тогда*

$$\frac{c(S_C)}{c(S_O)} \leqslant 1 + \ln \Delta. \tag{8}$$

Доказательство. В работе [2] показано, что алгоритм Хватала для задачи 3 имеет гарантированную оценку погрешности

$$(S_C) \leqslant \mathcal{H}(\Delta) \cdot (S_O), \tag{9}$$

где $\Delta = \max_{v \in V} |P_v|$, а $\mathcal{H}(\Delta) = \sum_{i=1}^{\Delta} 1/i$. В рассматриваемом случае $P_v = \{V \setminus B \mid B \in \mathcal{B}_v\}$, т. е. P_v есть семейство дополнений баз исходной системы $\mathcal{S} = (V, \mathcal{B})$, не содержащих элемент v. Другими словами, P_v — это множество ребер гиперграфа H, содержащих вершину v. Поэтому Δ — максимальная степень гиперграфа. Следовательно, $|P_v| = |\mathcal{B}_v|$. В свою очередь, несложно показать, что $\mathcal{H}(\Delta) \leqslant 1 + \ln \Delta$. Отсюда и из (8) следует оценка (7). Теорема доказана.

Авторы выражают признательность анонимному рецензенту за полезные замечания.

Литература

1. **Boppana R. B., Halldórsson M. M.** Approximating maximum independent sets by excluding subgraphs // BIT. 1992. V. 32, N 2. P. 180–196.
2. **Chvatal V.** A greedy heuristic for the set-covering problem // Math. Oper. Res. 1979. V. 4, N 3. P. 233–235.
3. **Halldórsson M. M.** Approximations of weighted independent set and hereditary subset problems // J. Graph Algorithms Appl. 2000. V. 4, N 1. P. 1–16.
4. **Halldórsson M. M., Radhakrishnan J.** Greed is good: approximating independent sets in sparse and bounded-degree graphs // Algorithmica. 1997. V. 18, N 1. P. 145–163.
5. **Hausmann D., Korte B.** Lower bounds on the worst-case complexity of some oracle algorithms // Discrete Math. 1978. V. 24, N 3. P. 261–276.
6. **Whitney H.** On the abstract properties of linear dependence // Amer. J. Math. 1935. V. 57. P. 509–533.

Адрес авторов:

Омский государственный университет
пр. Мира 55А, 644077 Омск,
Россия,
E-mail: iljev@iitam.omsk.net.ru
talants@rambler.ru

Статья поступила
15 мая 2003 г.
Переработанный вариант
6 июня 2003 г.