

УДК 519.854

ПОЛИНОМИАЛЬНЫЕ АЛГОРИТМЫ РЕШЕНИЯ МИНИМАКСНОЙ КВАДРАТИЧНОЙ ЗАДАЧИ О НАЗНАЧЕНИЯХ НА СЕТЯХ

Г. Г. Забудский, А. Ю. Лагздин

Аннотация. Рассматривается квадратичная задача о назначениях с минимаксным критерием в терминах теории графов: необходимо разместить вершины графа в узлы сети таким образом, чтобы максимальная связь между смежными вершинами была минимальной. Предлагаются полиномиальные алгоритмы решения этой задачи на специальных типах сетей.

Ключевые слова: квадратичная задача о назначениях, задача размещения, минимаксный критерий, полиномиальный алгоритм, граф.

Введение

Квадратичная задача о назначениях (КЗН) является классическим примером задачи размещения взаимосвязанных объектов в дискретной постановке. Основное внимание в литературе уделяется исследованию и решению КЗН с критерием минимальной суммарной стоимости связей. В общем случае задача NP-трудна. На её основе строятся модели многих прикладных задач оптимального размещения в автоматизации проектирования генеральных планов предприятий, планов цехов, печатных плат и в других областях. Исследованием задачи занимались В. А. Демиденко, Н. Н. Метельский, М. И. Свириденко, С. И. Сергеев, И. Х. Сигал, Буркард, Купманс и др. Для её решения были разработаны алгоритмы ветвей и границ, отсечений, полиномиальные алгоритмы и аппроксимационные схемы для специальных случаев [2–4, 6–8].

Однако на практике часто возникает потребность минимизировать наибольшую стоимость связей. Например, в организации экономного хранения и преобразования разреженных матриц для решения систем линейных алгебраических уравнений большой размерности [5] и при минимизации максимальной длины канала связи в распределённых системах [12]. Постановка КЗН, в которой необходимо минимизировать максимальную стоимость связей, предложена в [13] для задачи размещения

элементов на печатной плате. Отметим, что задача в такой постановке NP-трудна [9, 11]. В [5] предложены полиномиальные алгоритмы её решения для частных случаев, когда производится размещение звёзд и деревьев, гомеоморфных звёздам, на цепи. В [12] построены приближённые алгоритмы на цепи для специальных деревьев. Задача на плоскости рассматривалась в [10]. Наряду с разработкой алгоритмов решения задачи часто возникают вопросы получения верхних и нижних оценок оптимального значения целевой функции [9].

В нашей статье рассматривается следующая постановка КЗН. Структура связей между размещаемыми объектами представлена в виде рёберно-взвешенного обыкновенного графа, в котором вершины соответствуют объектам, а рёбра — связям между ними. Позиции, в которые размещаются объекты, — это узлы сети. Необходимо найти размещение, в котором максимальная связь между объектами будет минимальной. Предложены полиномиальные алгоритмы для специальных типов указанных графов и сетей.

1. Постановка задачи

Задан неориентированный взвешенный граф $L = (N, E)$ без петель и кратных рёбер. Каждому ребру $\{n_i, n_j\} \in E$ приписан вес $w(n_i, n_j) > 0$. Пусть $M = (V, U)$ — неориентированная сеть без петель и кратных рёбер. Ребру $\{v_i, v_j\} \in U$ приписан вес $r(v_i, v_j) > 0$. Считаем, что $|V| = |N| = m$. Размещением графа L на сети M будем называть взаимно однозначное отображение $\pi : N \rightarrow V$. Необходимо разместить вершины графа L в узлы сети M таким образом, чтобы максимальная связь между вершинами графа L была минимальной (минимаксный критерий). Целевая функция имеет вид

$$F(\pi) = \max_{(n_i, n_j) \in E} w(n_i, n_j) \rho(\pi(n_i), \pi(n_j)) \rightarrow \min_{\pi},$$

где $\rho(\pi(n_i), \pi(n_j))$ — кратчайшее расстояние между узлами $\pi(n_i)$ и $\pi(n_j)$ в сети M , в которых размещены вершины n_i и n_j графа L , $\pi(n_i) \neq \pi(n_j)$. Такую задачу размещения будем обозначать тройкой (L, M, F) .

2. Алгоритм на произвольных деревьях

Рассмотрим задачу размещения цепи $Ch = (N, E)$ на древовидной сети $T = (V, U)$ в случае, когда веса рёбер цепи Ch и сети T единичные, т. е. задачу (Ch, T, F) . В этом разделе для сформулированной задачи предложен полиномиальный алгоритм. Заметим, что задача (T, Ch, F) ,

где T — произвольное дерево, а Ch — цепь с единичными весами рёбер, является NP-трудной [1].

Для простоты обозначений в дальнейшем будем считать, что вершины цепи последовательно перенумерованы числами $\{1, \dots, m\}$. Через $d(v)$ будем обозначать степень узла $v \in V$. Выделим в сети T некоторую цепь $C_0 = (V_0, U_0)$, $V_0 = (k_1, \dots, k_p)$, соединяющую узлы k_1 и k_p , которую назовём *главной*. Каждому узлу $k_i \in V_0$ сопоставим поддерево $T_i = (V_i, U_i)$. Множество V_i состоит из узлов k_i и всех узлов дерева T , которые соединены с k_i цепями, не содержащими рёбер из U_0 . Множество U_i образовано рёбрами таких цепей.

В описанном ниже алгоритме осуществляется поиск в поддеревьях T_i сетей вида G_1 и G_2 , которые имеют следующую структуру. Пусть для некоторого $i = 2, \dots, p-1$ существуют узлы $v_r, v_{r+1}, v_s, v_t \in V_i$, v_{r+1} — узел, смежный с v_r , $\rho(v_{r+1}, k_i) > \rho(v_r, k_i)$, $\rho(v_s, v_{r+1}) < \rho(v_s, v_r)$, $\rho(v_t, v_{r+1}) < \rho(v_t, v_r)$ и $\rho(v_{r+1}, v_s) = \rho(v_{r+1}, v_t) = 2$, при этом цепи, концевыми узлами которых являются v_{r+1} и v_s , v_{r+1} и v_t , не пересекаются по рёбрам. Тогда сеть G_1 будет образована узлами и рёбрами цепей, соединяющих v_r с v_s и v_r с v_t . Узел v_r будем называть *корнем сети G_1* (рис. 1).

Если для некоторого $i = 2, \dots, p-2$ существуют различные узлы $v_r, v_s, v_t \in V_i$ такие, что $\rho(k_i, v_r) = \rho(k_i, v_s) = \rho(k_i, v_t) = 2$, и цепи, соединяющие указанные узлы с k_i , не пересекаются, то сеть G_2 образована узлами и рёбрами цепей, соединяющих k_i с v_r, v_s и v_t . Корнем сети G_2 назовём вершину k_i (рис. 1).

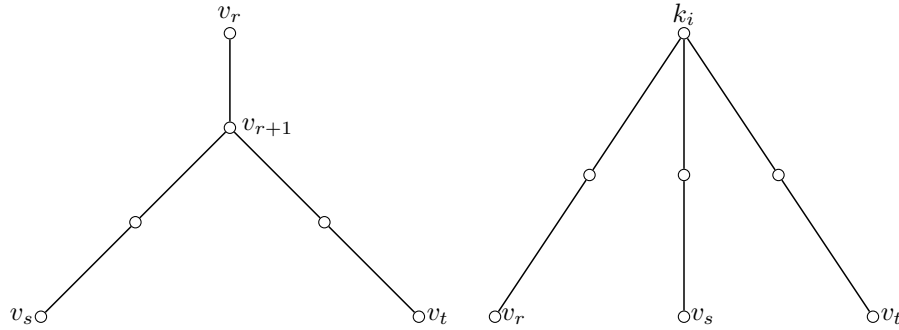


Рис. 1. Сети вида G_1 и G_2

АЛГОРИТМ А1 РЕШЕНИЯ ЗАДАЧИ (Ch, T, F)

ШАГ 1. Рассматриваем очередное представление дерева T с выделенной главной цепью, соединяющей висячие узлы дерева. Если рассмотрены все представления дерева, то переход на шаг 3 для T с произвольной

главной цепью. Для фиксированного представления с помощью алгоритма $A1_G$ в дереве T осуществляем поиск сетей вида G_1 и G_2 .

ШАГ 2. Если была найдена сеть вида G_1 или G_2 , то переход на шаг 1. Если сетей вида G_1 и G_2 нет, то переход на шаг 3 для текущего представления T .

ШАГ 3. Применяем алгоритм $A1_\pi$ последовательного размещения вершин цепи Ch в узлы дерева T с выделенной главной цепью.

Детальное описание алгоритмов $A1_G$ и $A1_\pi$ приведено в приложении.

Заметим, что если нашлось представление дерева T , не являющегося цепью, без сетей вида G_1 или G_2 , то алгоритмом $A1_\pi$ будет получено решение π^* исходной задачи с оптимальным значением целевой функции $F(\pi^*) = 2$. Если во всех рассмотренных представлениях дерева T имеется сеть вида G_1 или G_2 , то оптимальное значение $F(\pi^*)$ равно 3. Если T — цепь, то $F(\pi^*) = 1$.

Изложим идею алгоритма $A1_\pi$. Последовательно размещаем вершины цепи Ch в узлы главной цепи C_0 до тех пор, пока $d(k_i) \geq 3$ для очередного узла k_i , $i = 1, \dots, p$, т. е. в данном представлении дерева T узел k_i является корнем поддерева T_i . Упорядочим ветви в T_i по неубыванию количества входящих в них узлов. Последовательно рассматриваем ветви и размещаем вершины цепи Ch в их узлы (занимаем узлы) через один, двигаясь от корня. Узлы, которые оставляем незанятыми при таком размещении, будем называть *просмотренными*. При достижении висячего узла в текущей ветви двигаемся к узлу k_i , размещая очередные вершины Ch в просмотренные, но незанятые узлы поддерева.

Далее под размещением π^* будем понимать размещение цепи Ch на сети T , полученное алгоритмом $A1$.

Пусть дано некоторое размещение π цепи Ch на сети T . Для размещения π главной цепью C_0 будем считать цепь, соединяющую узлы $\pi(1)$ и $\pi(m)$.

Лемма 1. Если узел $k_i \in C_0$ является корнем сети вида G_1 или G_2 , причём $\pi(1), \pi(m) \notin T_i$, то существуют вершины $j, j+1 \in N$ такие, что $\rho(\pi(j), \pi(j+1)) \geq 3$.

ДОКАЗАТЕЛЬСТВО приведём для случая, когда k_i — корень сети вида G_1 . Поскольку $\pi(1), \pi(m) \notin T_i$, любая вершина, размещённая в сети G_1 , имеет две смежные.

Пусть в узле $v_1 \in G_1$ размещена вершина $s+1$, $s = 2, \dots, m-8$, цепи Ch (рис. 2). Одна смежная с ней вершина s размещена вне G_1 в v_0 или v_7 , а другая смежная $s+2$ — в сети G_1 в v_3 или в v_4 , так как

в противном случае $F(\pi) \geq 3$.

Пусть $\pi(s+2) = v_3$ (для $\pi(s+2) = v_4$ симметричный случай). Тогда вершина $s+3$ расположена в v_5 , v_4 или v_2 . Если $\pi(s+3) = v_4$ или $\pi(s+3) = v_2$, то при размещении следующих вершин цепи получим два подряд занятых узла дерева, а значит, $F(\pi) \geq 3$. Следовательно, $\pi(s+3) = v_5$, $\pi(s+4) = v_2$ (иначе $F(\pi) \geq 3$). Вершина $s+5$ расположена в v_6 или в v_4 . Если $\pi(s+5) = v_4$, то при размещении следующих вершин цепи получим $F(\pi) \geq 3$. Таким образом, $\pi(s+5) = v_6$. Аналогично при $\pi(s+6) = v_4$ выполнено неравенство $F(\pi) \leq 3$. Так как $\pi(1), \pi(m) \notin T_i$, вершина $s+7$ расположена вне сети G_1 в v_0 или v_7 . Таким образом, $F(\pi) \geq 3$. Получили, что существует вершина цепи такая, что хотя бы одна из смежных с ней вершин размещена в сети на расстоянии не менее трёх.

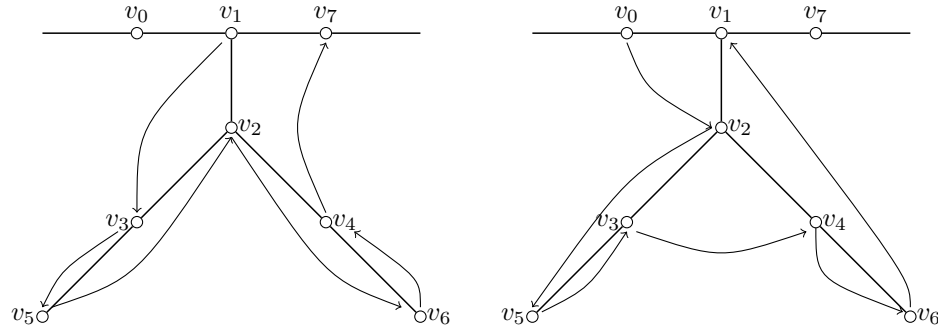


Рис. 2. Варианты размещения на сети вида G_1

При рассмотрении случая $\pi(s) = v_0$, $\pi(s+1) = v_2$ (рис. 2) рассуждения аналогичны.

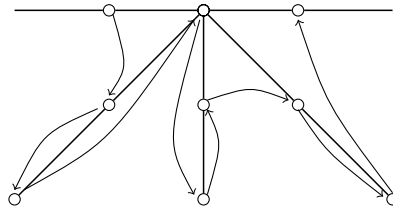


Рис. 3. Вариант размещения на сети вида G_2

Если k_i является корнем сети вида G_2 , то доказательство аналогично (рис. 3). Лемма 1 доказана.

Следствие. Если $G_1 \subseteq T_i$ или $G_2 \subseteq T_i$ для некоторого $i = 2, \dots, p-1$, то $F(\pi) \geq 3$ для любого размещения π при условии, что $\pi(1), \pi(m) \notin T_i$.

Замечание. $F(\pi^*) = 1$ тогда и только тогда, когда T — цепь.

Лемма 2. Пусть дерево T не является цепью. Если в его представлении с выделенной главной цепью C_0 не содержится сетей вида G_1 и G_2 , то $F(\pi^*) = 2$.

ДОКАЗАТЕЛЬСТВО. Заметим, что непересекающиеся по рёбрам цепи длины не менее двух могут выходить только из узла $k_i \in C_0$ и таких цепей не может быть больше двух, иначе $G_2 \subseteq T$. При последовательном размещении вершин цепи Ch в узлы дерева T переход от одной ветви к следующей показан на рис. 4, где v_r^1, v_r^2 — узлы, смежные с k_i и принадлежащие разным цепям. Если для некоторой вершины t цепи Ch выполнено $\pi^*(t) = v_r^1$, то согласно алгоритму $\pi^*(t+1) = v_r^2$. Если же $\pi^*(t) = k_i$, то $\pi^*(t+1) = v_{r+1}^1$. Таким образом, если для некоторого $t \in N$ вершины t и $t+1$ размещены в какие-либо из указанных узлов, то $\rho(\pi^*(t), \pi^*(t+1)) \leq 2$.

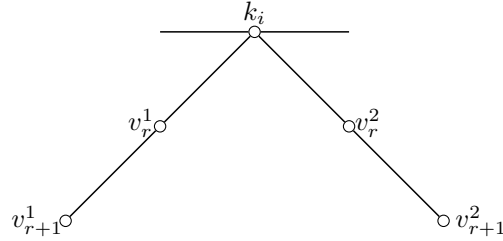


Рис. 4. Размещение в узлы, смежные с корнем поддерева

Рассмотрим случаи, возникающие при размещении вершин цепи Ch на одной из ветвей поддерева T_i с корнем k_i . Выделим в этой ветви самую длинную цепь и обозначим её через C_1 . Концевыми узлами этой цепи будут k_i и v_s . Так как $G_1 \not\subseteq T_i$, любой узел $v_r \in V_i \setminus \{k_i\}$ такой, что $d(v_r) \geq 3$, имеет смежные узлы степени 1. Обозначим через $V_r = v_r^1, \dots, v_r^z$ множество узлов, смежных с v_r , таких, что $d(v_r^k) = 1$, $k = 1, \dots, z$. Покажем, что при этих условиях $F(\pi^*) = 2$.

СЛУЧАЙ 1. Каждому узлу v_r цепи C_1 , кроме k_i и v_s , соответствует поддерево с множеством узлов V_r . Рассмотрим случай $|V_r| = 0$ для всех узлов v_r цепи C_1 . Согласно алгоритму A1 размещение π^* имеет вид, изображённый слева на рис. 5. Пусть $\pi^*(t-2) = v_{r-2}$. Тогда вершины цепи Ch , начиная с $t-1$ до $t-1 + |V_{r-1}|$, размещены в узлах множества V_{r-1} , а $\pi^*(t-1 + |V_{r-1}| + 1) = v_r$. Таким образом, когда дойдём до конечного узла цепи C_1 , незанятыми останутся узлы этой цепи через один и соответствующие множества V_r .

Пусть a — вершина цепи Ch , размещённая в узел v_s . Тогда вершины $q \in N$ для $q < a$ размещены во все узлы $v_{r-2}, V_{r-1}, v_r, V_{r+1}, \dots, v_s$, причём $V_{r-2}, v_{r-1}, V_r, v_{r+1}$ остаются свободными. Вершины от $a + 1$ до $a + 1 + |V_{s-1}|$ размещаются в узлы множества V_{s-1} , вершина $a + 1 + |V_{s-1}| + 1$ — в узел v_{s-2} , и т. д.

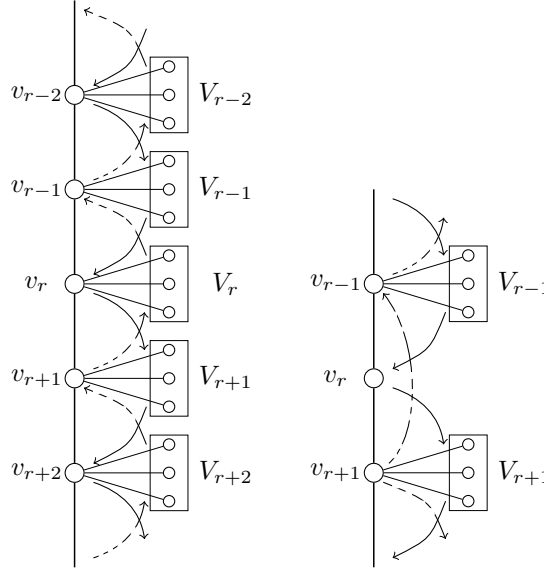


Рис. 5. Случаи 1 и 2

Рассмотрим произвольную вершину q цепи Ch , размещённую в данной ветви. Пусть $\pi^*(q) = v_r$, тогда вершины $q - 1$ и $q + 1$ размещены в узлах множеств V_{r-1} и V_{r+1} . Заметим, что расстояние от любого узла указанных множеств до v_r равно 2. Аналогично, если $\pi^*(q) \in V_r$, то $\rho(\pi^*(q), \pi^*(q + 1)) \leq 2$. Таким образом, $\rho(\pi^*(q), \pi^*(q + 1)) \leq 2$ для любой $q \in N$, размещённой в указанной ветви.

СЛУЧАЙ 2. Пусть $|V_r| = 0$, $|V_{r-1}| \neq 0$ и $|V_{r+1}| \neq 0$. Тогда в размещении π^* имеем (рис. 5): если $\pi^*(t) = v_r$, то вершина $t - 1$ цепи Ch размещена в узел из множества V_{r-1} , вершина $t + 1$ размещена в узел из V_{r+1} и т. д. Заметим, что две вершины цепи Ch , размещённые в v_{r-1} и v_{r+1} , являются смежными. Следовательно, для всех $q = 1, \dots, m - 1$ выполнено неравенство $\rho(\pi^*(q), \pi^*(q + 1)) \leq 2$.

СЛУЧАЙ 3. Пусть $|V_r| = 0$ и $|V_{r+1}| = 0$, а $|V_{r-1}| \neq 0$ и $|V_{r+2}| \neq 0$. Тогда если $\pi^*(t) = v_r$, то вершина $t - 1$ размещена в узел из множества V_{r-1} (рис. 6), $\pi^*(t + 1) = v_{r+2}$ и т. д. Две вершины цепи Ch , размещённые в v_{r-1} и v_{r+1} , являются смежными. Вторая вершина, смежная

с вершиной, расположенной в v_{r+1} , размещена в узел из множества V_{r+2} , и т. д. Таким образом, для всех $q = 1, \dots, m - 1$ справедливо неравенство $\rho(\pi^*(q), \pi^*(q + 1)) \leq 2$.

СЛУЧАЙ 4. Пусть $|V_{r-1}| = 0$, $|V_r| = 0$, $|V_{r+1}| = 0$, а $|V_{r-2}| \neq 0$ и $|V_{r+2}| \neq 0$. Тогда в размещении π^* имеем (рис. 6): если $\pi^*(t - 1) = v_{r-1}$, то вершина $t - 2$ размещена в узел из V_{r-2} и $\pi^*(t) = v_{r+1}$, вершина $t + 1$ размещена в узел из V_{r+2} , и т. д. Две вершины цепи Ch , размещённые в v_{r-2} и v_r , являются смежными. Вершины, размещённые в v_r и v_{r+2} , также смежные, и т. д. Следовательно, $\rho(\pi^*(q), \pi^*(q + 1)) \leq 2$ для всех $q = 1, \dots, m - 1$.

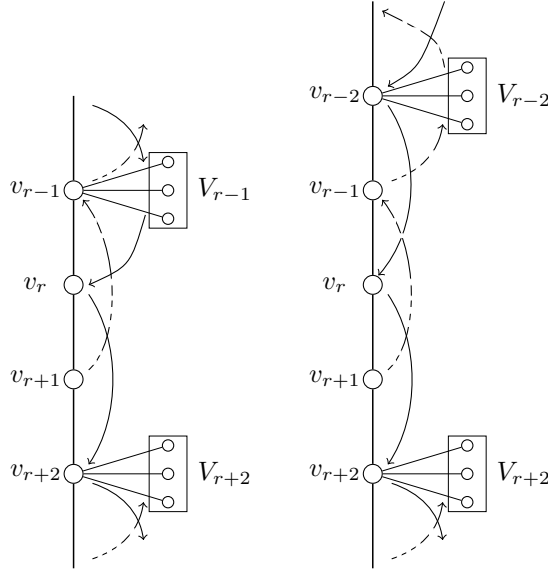


Рис. 6. Случаи 3 и 4

Заметим, что любая структура поддерева, не содержащего сетей вида G_1 и G_2 , сводится к рассмотренным выше случаям, следовательно, $F(\pi^*) = 2$. Лемма 2 доказана.

Лемма 3. Если π — размещение цепи Ch на сети T и $F(\pi) = 2$, то существует представление дерева T с выделенной главной цепью C_0 такое, что $G_1, G_2 \not\subseteq T_i$ для всех $i = 2, \dots, p - 1$.

ДОКАЗАТЕЛЬСТВО. Выделим в дереве T цепь C_1 , соединяющую $\pi(1)$ и $\pi(m)$, узлы которой являются корнями поддеревьев T_j^* , $j = 1, \dots, q$. Если узел $\pi(1)$ не является висющим, то выберем произвольный концевой узел $v_a \in T_1^*$, иначе $v_a = \pi(1)$. Аналогично определим узел $v_b \in T_q^*$.

Рассмотрим представление дерева с цепью, соединяющей узлы v_a и v_b , как главной. Если $G_1 \subseteq T_i$ или $G_2 \subseteq T_i$ для некоторого поддерева T_i , то по лемме 1 получаем противоречие с существованием размещения π такого, что $F(\pi) = 2$. Лемма 3 доказана.

Следствие. Если существует размещение π цепи Ch на сети T такое, что $F(\pi) = 2$, то $F(\pi^*) = 2$.

ДОКАЗАТЕЛЬСТВО. Действительно, существует представление дерева T с выделенной главной цепью C_0 такое, что $G_1, G_2 \not\subseteq T_i$ для всех $i = 2, \dots, p-1$ по лемме 3. Поскольку в алгоритме A_1 рассматриваются всевозможные главные цепи, будет рассмотрено и представление T с главной цепью C_0 . По лемме 2 $F(\pi^*) = 2$.

Лемма 4. Для любой древовидной сети T справедливо неравенство $F(\pi^*) \leq 3$.

ДОКАЗАТЕЛЬСТВО. Отметим, что по следствию из леммы 1 и лемме 2 неравенство $F(\pi^*) \geq 3$ выполнено, когда $G_1 \subseteq T_i$ либо $G_2 \subseteq T_i$ для $i \in \{2, \dots, p-1\}$. Пусть $G_1 \subseteq T_i$ для некоторого i . Покажем, что $F(\pi^*)$ равно 3. Случай $G_2 \subseteq T_i$ рассматривается аналогично.

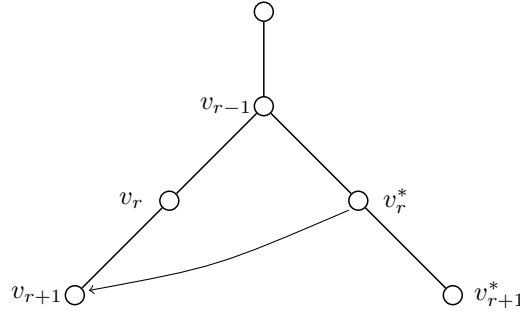


Рис. 7. Размещение в разные ветви сети G_1

Отметим, что найдутся две вершины $t-1$ и t цепи Ch , размещённые в разные ветви сети с подкорнем v_{r-1} , содержащей G_1 (рис. 7). Заметим, что по алгоритму $A1_\pi$ если в некоторую ветвь поддерева размещаются вершины $q, \dots, q+j$, то в узел, смежный с подкорнем ветви, всегда будет размещена вершина $q+j$. Пусть в узел v_r^* , смежный с v_{r-1} , размещена вершина $t-1$. Заметим, что при «переходе» в другую ветвь с подкорнем v_{r-1} (размещение вершины t) все узлы текущей ветви заняты.

Покажем, что узел v_{r-1} при этом также занят. Действительно, пусть узел v_{r-1} свободен. Тогда для некоторого j найдутся вершины $t-j-1$

и $t - j$ такие, что $t - j$ размещена в узел, находящийся дальше от корня поддерева T_i относительно v_r^* , а $t - j - 1$ — ближе к корню относительно v_{r-1} . Но по алгоритму вершина $t - j$ размещена либо в v_{r-1} , либо в v_r^* , так как при размещении $t - j$ они были свободными, следовательно, v_{r-1} занят. Тогда узел v_r остаётся свободным, на шаге 5 рассматривается v_{r+1} , смежный с v_r , и $\pi^*(t) = v_{r+1}$. Таким образом, $\rho(\pi^*(t - 1), \pi^*(t)) = 3$.

По построению алгоритма $A1_\pi$ при размещении вершин Ch на цепи в сети T никакие две смежные вершины Ch не могут быть размещены в узлы, находящиеся на расстоянии более 3 друг от друга. При переходе из одной ветви произвольной подсети в другую это свойство показано выше.

Таким образом, для любой древовидной сети T справедливо неравенство $F(\pi^*) \leq 3$. Лемма 4 доказана.

Теорема. Полученное алгоритмом $A1$ решение задачи (Ch, T, F) является оптимальным.

ДОКАЗАТЕЛЬСТВО. По следствию леммы 1 если $G_1 \subseteq T_i$ или $G_2 \subseteq T_i$ для некоторого $i = 2, \dots, p - 1$, то $F(\pi) \geq 3$ для любого размещения π . По лемме 4 $F(\pi^*) \leq 3$. По следствию леммы 3 если существует размещение π , для которого $F(\pi) = 2$, то $F(\pi^*) = 2$. Если T — цепь, то $F(\pi^*) = 1$. Теорема доказана.

Отметим, что трудоёмкость решения задачи (Ch, T, F) не превосходит $O(m^4 \log m)$. Действительно, алгоритм $A1$ состоит из перебора главных цепей древовидной сети T , поиска в ней сетей вида G_1 и G_2 (алгоритм $A1_G$) и непосредственного размещения (алгоритм $A1_\pi$). Число главных цепей не больше $m(m - 1)/2$, трудоёмкость алгоритма $A1_G$ не превосходит $O(m^2 \log m)$, алгоритма $A1_\pi$ — $O(m^2 \log m)$, и, следовательно, трудоёмкость $A1$ не превосходит $O(m^4 \log m)$.

3. Алгоритмы на специальных и общих сетях

Определение. Граф $S = (V, U)$, $|V| = m$, называется *звездой*, если имеется вершина $v_1 \in V$, степень которой равна $m - 1$, а степени остальных вершин равны 1, при этом v_1 — *центр звезды* [5].

Рассмотрим задачу (SW, MW, F) размещения взвешенной звезды $SW = (N, E)$ на произвольной взвешенной сети $MW = (V, U)$.

Утверждение 1. Задача (SW, MW, F) разрешима за $O(m^3)$ операций.

АЛГОРИТМ А2

ШАГ 1. Строим матрицу кратчайших расстояний между всеми парами узлов сети MW . Матрица имеет m строк и $m - 1$ столбец: не учитываем нулевые диагональные элементы.

ШАГ 2. Упорядочиваем элементы строк полученной матрицы по неубыванию, получаем матрицу $K = (k_{ij})$, $i = 1, \dots, m$, $j = 1, \dots, m - 1$.

ШАГ 3. Упорядочиваем вершины графа SW по невозрастанию стоимости связи с центром, т. е. $w(n_1, n_2) \geq w(n_1, n_3) \geq \dots \geq w(n_1, n_m)$.

ШАГ 4. Для каждой строки матрицы K вычисляем произведения её элементов с соответствующими компонентами вектора стоимостей связи с центром звезды (первый элемент строки — с первой компонентой вектора, второй — со второй, и т. д.) и выбираем из них максимальное. Далее среди этих максимумов находим минимальное значение. Пусть такой минимум достигается в строке, соответствующей узлу v_1 , причём столбцы в этой строке соответствуют узлам v_2, \dots, v_m .

ШАГ 5. Полагаем $\pi(n_1) = v_1$, $\pi(n_2) = v_2$, \dots , $\pi(n_m) = v_m$.

Трудоёмкость предложенного алгоритма определяется числом операций $O(m^3)$, необходимых для построения кратчайших расстояний между всеми парами узлов, и $O(m^2 \log m)$ операций по упорядочению строк матрицы K , поэтому не превосходит $O(m^3)$. Оптимальность алгоритма основана на следующем свойстве.

Свойство 1. Для двух произвольных наборов неотрицательных чисел $a = (a_1, a_2, \dots, a_n)$ и $b = (b_1, b_2, \dots, b_n)$ максимальное покомпонентное произведение минимально тогда и только тогда, когда один набор упорядочен в порядке невозрастания, а другой — в порядке неубывания.

ДОКАЗАТЕЛЬСТВО. Пусть $a_1 \geq a_2 \geq \dots \geq a_n$, а набор b не упорядочен. Тогда если $b_1 \geq b_2$, то, очевидно, справедливо неравенство

$$f = \max\{a_1 b_1; a_2 b_2; a_3 b_3; \dots; a_n b_n\} \geq \max\{a_1 b_2; a_2 b_1; a_3 b_3; \dots; a_n b_n\}.$$

Далее сравниваем b_1 и b_3 и т. д. Производим, если нужно, соответствующие парные замены, не увеличивая при этом значения функции f . После n^2 -кратного проведения сравнения получим

$$f \geq \max\{a_1 b_{i_1}; a_2 b_{i_2}, \dots, a_n b_{i_n}\},$$

причём $b_{i_1} \leq b_{i_2} \leq \dots \leq b_{i_n}$. Свойство 1 доказано.

Рассмотрим задачу (Ch, SW, F) размещения цепи $Ch = (N, E)$ с единичными весами рёбер на взвешенной звезде $SW = (V, U)$.

Утверждение 2. Задача (Ch, SW, F) разрешима за $O(m \log m)$ операций.

Отметим, что решение указанной задачи сводится к разбиению набора положительных чисел на пары таким образом, чтобы максимальная сумма в парах была минимальной. Указанный набор в задаче размещения — это веса лучей звезды.

Пусть задан набор a из чётного числа положительных чисел $\{a_1, a_2, \dots, a_n\}$. Будем считать, что его элементы упорядочены по невозрастанию, т. е. справедливы неравенства $a_1 \geq a_2 \geq \dots \geq a_n$.

Свойство 2. Максимальная сумма пар элементов (a_j, a_k) набора a таких, что справедливо равенство $j + k = n + 1$, будет минимальной по всем разбиениям набора на непересекающиеся пары.

ДОКАЗАТЕЛЬСТВО. От противного. Пусть $\min_{(l,m)} \max\{a_l + a_m\} = a_i + a_j$.

Рассмотрим случай $i + j \neq n + 1$. Отметим, что неравенство $i + j > n + 1$ неверно, иначе существует пара (s, t) такая, что $a_s \geq a_i$ и $a_t \geq a_j$, и тогда $a_s + a_t \geq a_i + a_j$ и $s + t < n + 1$.

Пусть $i + j < n + 1$, тогда существует пара (s, t) , для которой $s + t > n + 1$. Пусть $i > j$ и $s > t$. Образует пары (i, t) и (j, s) путём парных замен. Получаем $a_i + a_t \leq a_i + a_j$ и $a_j + a_s \leq a_i + a_j$; противоречие с тем, что $a_i + a_j$ — минимальная из максимальных сумм по всем разбиениям на пары массива a .

Заметим, что пары элементов (a_j, a_k) такие, что $j + k = n + 1$, образуют разбиение набора a на непересекающиеся пары, следовательно, минимальной из максимальных сумм всех разбиений будет именно максимальная сумма данного разбиения. Свойство 2 доказано.

АЛГОРИТМ АЗ

ШАГ 1. Упорядочиваем узлы сети SW с центром в v_1 в порядке невозрастания весов: $r(v_1, v_2) \geq r(v_1, v_3) \geq \dots \geq r(v_1, v_m)$.

ШАГ 2. Полагаем $\pi(n_1) = v_2$, $\pi(n_2) = v_1$, $\pi(n_3) = v_3$, $\pi(n_4) = v_m$, $\pi(n_5) = v_4$ и т. д. Общая формула для вершин цепи n_j , $j \geq 4$, имеет вид $\pi(n_{2k}) = v_{m-k+2}$, $\pi(n_{2k+1}) = v_{k+2}$, $k = 2, 3, \dots, [m/2]$.

Отметим, что алгоритм АЗ разбивает лучи звезды именно в таком порядке, как показано в свойстве 2, откуда следует, что он находит оптимальное решение задачи (Ch, SW, F) . Если число лучей в SW чётное, то указанный в свойстве набор имеет нечётное число элементов. В этом случае «середина» набора не рассматривается.

Пусть $Cy = (N, E)$ — цикл с единичными весами рёбер, $SW = (V, U)$ — взвешенная звезда. Алгоритм решения задачи (Cy, SW, F) отличается от алгоритма *A3* шагом 2: на нём полагаем $\pi(n_1) = v_2$, $\pi(n_2) = v_1$, $\pi(n_3) = v_3$, $\pi(n_4) = v_{m-1}$, $\pi(n_5) = v_4$ и т. д. Общая формула имеет вид $\pi(n_{2k}) = v_{m-k+1}$, $\pi(n_{2k+1}) = v_{k+2}$, $k = 2, 3, \dots, [m/2]$. Трудоемкость построенного алгоритма не превосходит $O(m \log m)$.

Определение. Если $T = (Y, Q)$ — дерево, в котором имеется вершина $y_1 \in Y$ степени не менее трёх, а степени других вершин не превосходят двух, то дерево T *гомеоморфно звезде* HS [5].

Рассмотрим задачу размещения цепи $Ch = (N, E)$ с единичными весами рёбер на сети, гомеоморфной звезде $HS = (V, U)$ с единичными весами рёбер.

Утверждение 3. Задача (Ch, HS, F) разрешима за $O(m)$ операций.

Приведём алгоритм построения оптимального решения указанной задачи.

АЛГОРИТМ *A4*

ШАГ 1. Размещаем последовательно вершины цепи Ch в узлы сети HS , начиная с концевой узла произвольного луча с количеством узлов не менее двух, до узла, смежного с центром HS .

ШАГ 2. Если имеются свободные смежные с центром сети HS узлы $v_i \in V$ такие, что $d(v_i) = 1$, то последовательно размещаем в них вершины цепи Ch . Если таких узлов нет и центр HS не занят, то размещаем очередную вершину в незанятый смежный с центром узел $v^1 \in V$ произвольного луча. Если центр занят и остался один незанятый луч, то размещаем вершину в v^1 и переходим на шаг 4. Иначе очередная вершина размещается в узел $v^2 \in V$, смежный с v^1 .

ШАГ 3. Размещаем вершины цепи в узлы луча через один и доходим до висячего узла луча. Далее, двигаясь в обратном направлении к центру, размещаем вершины цепи аналогично. После размещения вершины цепи Ch в последний незанятый узел луча, очередную вершину сети Ch размещаем в центр HS , если он не занят. Переход на шаг 2.

ШАГ 4. Последовательно размещаем вершины цепи Ch в узлы текущего луча, двигаясь к висячему узлу. Конец алгоритма.

Нетрудно заметить, что для полученного алгоритмом *A4* размещения π будет выполнено неравенство $F(\pi) \leq 3$.

Приведём пример минимальной по количеству узлов сети HS , гомеоморфной звезде, для которой $F(\pi^*) = 3$ для оптимального решения π^*

задачи (Ch, HS, F) . Сеть состоит из 11 узлов и 5 лучей, на каждом из которых по два узла. Обозначим эту сеть через HS_5^2 .

Отметим, что $F(\pi^*) = 3$ для оптимального размещения π^* цепи Ch на произвольной сети, гомеоморфной звезде HS и содержащей подсеть HS_5^2 . Если такой подсети нет, то оптимальное значение целевой функции $F(\pi^*)$ равно 2. Действительно, отсутствие такой подсети означает, что либо число лучей не более четырёх, либо хотя бы один луч содержит только один узел. Нетрудно проверить, что в данном случае для полученного алгоритмом $A4$ размещения π справедливо $F(\pi) = 2$. Из приведённых рассуждений следует, что алгоритм $A4$ находит оптимальное решение задачи (Ch, HS, F) .

Рассмотрим задачу размещения цикла $Cy = (N, E)$ с единичными весами рёбер на сети, гомеоморфной звезде $HS = (V, U)$ с единичными весами рёбер. Алгоритм решения данной задачи отличается от алгоритма $A4$ в следующем: на шаге 1 произвольная вершина цикла Cy размещается в центр HS , а на шаге 3 отсутствует проверка, является ли единственным следующим незанятый луч. Шаг 4 в алгоритме отсутствует, а алгоритм заканчивает работу, когда все вершины Cy размещены в узлы сети HS . Трудоёмкость построенного алгоритма не превосходит $O(m)$ операций. Аналогично можно указать минимальную сеть для задачи (Cy, HS, F) . В ранее введённых обозначениях это сеть HS_3^2 .

4. Заключение

Рассмотрена квадратичная задача о назначениях с минимаксным критерием, когда структура связей между размещаемыми объектами представлена в виде неориентированного графа, а область, в которой производится размещение, — неориентированная сеть. Предложены полиномиальные алгоритмы размещения цепи с единичными весами рёбер на дереве и на сети, гомеоморфной звезде с единичными весами рёбер, а также на взвешенной звезде; цикла с единичными весами рёбер на взвешенной звезде и на сети, гомеоморфной звезде с единичными весами рёбер; взвешенной звезды на произвольной взвешенной сети.

5. Приложения

АЛГОРИТМ $A1_G$ ПОИСКА СЕТЕЙ ВИДА G_1 И G_2

Рассмотрим представление дерева T с выделенной главной цепью $C_0 = (V_0, U_0)$, $V_0 = \{k_1, \dots, k_p\}$, $i := 2$.

ШАГ 1. $k := k_i$. Если $d(k) = 1$, то конец алгоритма, сети вида G_1 и G_2 не найдены. Иначе

при $d(k) = 2$ полагаем $i := i + 1$ и переходим на начало шага 1,
 при $d(k) = 3$ упорядочиваем цепи поддерева T_i по неубыванию числа входящих в них узлов. Если существует три непересекающихся по рёбрам цепи длины не менее двух, то конец алгоритма, найдена сеть вида G_2 .

В противном случае полагаем $v := k$, помечаем все цепи поддерева с корнем v как непросмотренные.

ШАГ 2. C_v — очередная непросмотренная цепь поддерева с корнем v . Если такой нет, то при $v = k$ полагаем $i := i + 1$ и переходим на начало шага 1, иначе u — смежный с v узел, находящийся ближе к корню поддерева T_i , помечаем соответствующую цепь C_u как просмотренную, полагаем $v := u$ и переходим на начало шага 2. Если имеется цепь C_v и u — узел C_v , смежный с v , то

при $d(u) = 1$ помечаем цепь C_v как просмотренную и переходим на начало шага 2;

при $d(u) = 2$ имеем w — узел C_v , смежный с u , $w \neq v$, полагаем $v := w$ и переходим на начало шага 2;

при $d(u) = 3$ обозначим через j число непересекающихся по рёбрам цепей длины не менее двух поддерева с корнем u ;

при $j = 0$ помечаем цепь C_v как просмотренную и переходим на начало шага 2;

при $j = 1$ имеем w — узел соответствующей цепи, смежный с u , $w \neq v$, полагаем $v := w$ и переходим на начало шага 2;

при $j \geq 2$ найдена сеть вида G_1 . Конец алгоритма.

АЛГОРИТМ $A1_\pi$ РАЗМЕЩЕНИЯ ЦЕПИ Ch НА ДЕРЕВЕ T

Рассматриваем представление дерева T с выделенной главной цепью $C_0 = (V_0, U_0)$, $V_0 = \{k_1, \dots, k_p\}$.

ШАГ 1. $\pi(1) := k_1$, $t := 2$, $i := 1$.

ШАГ 2. $i := i + 1$. Если $\pi(t - 1) \neq k_{i-1}$, то $\pi(t) := k_i$, $t := t + 1$.

ШАГ 3. При $d(k_i) = 1$ если k_i не занят, то $\pi(t) := k_i$. Конец алгоритма.

При $d(k_i) = 2$ если k_i не занят, то полагаем $\pi(t) := k_i$ и $t := t + 1$ и переходим на начало шага 3.

При $d(k_i) \geq 3$ упорядочиваем цепи поддерева T_i по неубыванию числа входящих в них узлов, полагаем $v := k_i$, помечаем все цепи и узлы поддерева с корнем v как непросмотренные.

ШАГ 4. Пусть C_v — очередная непросмотренная цепь поддерева с корнем v .

Если такой цепи нет, то последовательно размещаем вершины цепи Ch в просмотренные узлы цепей единичной длины (если такие име-

ются); если v — просмотренный узел, то полагаем $\pi(t) := v$, $t := t + 1$. Если $v = k_i$, то переходим на шаг 3, иначе пусть w — смежный с v , находящийся ближе к k_i , помечаем соответствующую цепь C_w как просмотренную, полагаем $v := w$ и переходим на начало шага 4.

Если имеется непросмотренная цепь C_v и u — узел C_v , смежный с v , и при этом v не занят, то $\pi(t) := u$, иначе u помечаем как просмотренный. Полагаем $v := u$.

Шаг 5.

Если $d(v) = 1$, то полагаем $\pi(t) := v$, $t := t + 1$. Пусть u — смежный узел с v . Полагаем $v := u$ и переходим на шаг 4.

Если $d(v) = 2$ и w — смежный с v , расположенный дальше от k_i , u — другой смежный с v , то

при $d(w) = 1$ полагаем $\pi(t) := w$, $t := t + 1$ и если v свободен, то $\pi(t) := v$, $t := t + 1$, полагаем $v := u$ и переходим на шаг 4;

при $d(w) \geq 2$ если v не занят, то $\pi(t) := w$ и $t := t + 1$, затем полагаем $v := w$ и переходим на шаг 4;

при $d(v) \geq 3$ если расположенный ближе к k_i узел u , смежный с v , занят, то переходим на шаг 4, иначе полагаем $\pi(t) := v$, $t := t + 1$.

Шаг 6. Рассматриваем очередную непросмотренную цепь C_v поддерева с корнем v . Если такой цепи нет, то переходим на шаг 4. Иначе пусть w — узел C_v , смежный с v . Тогда

при $d(w) = 1$ помечаем узел w и цепь C_v как просмотренные и переходим на начало шага 6;

при $d(w) \geq 2$ полагаем $v := w$ и переходим на шаг 4.

ЛИТЕРАТУРА

1. Гэри М., Джонсон Д. Вычислительные машины и труднорешаемые задачи // М.: Мир, 1982. — 416 с.
2. Демиденко В. М. Обобщение условий сильной разрешимости квадратичной задачи о назначениях с матрицами анти-Монжа и Теплица // Докл. НАН Беларуси, 2003. — Т. 47, № 2. — С. 15–18.
3. Забудский Г. Г. О некоторых задачах размещения на графах // Тр. XI Байкальской междунар. школы-семинара «Методы оптимизации и их приложения». — Иркутск: ИСЭ им. Л. А. Мелентьева СО РАН, 1998. — С. 135–138.
4. Забудский Г. Г., Лагздин А. Ю. Алгоритм решения квадратичной задачи о назначениях с минимаксным критерием на дереве // Материалы VII Междунар. научно-техн. конф. «Динамика систем, механизмов и машин». — Омск: ОмГТУ, 2009. — Т. 3. — С. 23–27.

5. **Иорданский М. А.** О минимаксных нумерациях вершин графа // Межвуз. сб. «Комбинаторно-алгебраические методы в прикладной математике». — Горький: ГГУ им. Лобачевского. — 1986. — С. 60–73.
6. **Метельский Н. Н.** Методы локальной оптимизации для задачи размещения двудольных графов // Журн. вычисл. математики и мат. физики. — 1984. — Т. 24, № 9. — С. 1428–1432.
7. **Сергеев С. И.** Квадратичная задача назначения. I // Автоматика и телемеханика. — 1999. — № 8. — С. 127–147.
8. **Arkin E., Hassin R., Sviridenko M.** Approximating the maximum quadratic assignment problem // Inf. Process. Lett. — 2001. — Vol. 77. — P. 13–16.
9. **Burkard R. E., Dell’Amico M., Mortello S.** Assignment problems. — Philadelphia: SIAM, 2009. — 402 p.
10. **Burkard R. E., Fincke U.** On random quadratic bottleneck assignment problems // Math. Programming. — 1982. — Vol. 23, N 1. — P. 227–232.
11. **Farahani R. Z., Hekmatfar M.** Facility location: concepts, models, algorithms and case studies. — Heidelberg: Physica-Verl., 2009. — 543 p.
12. **Haralambides J., Makedon F.** Approximation algorithms for the bandwidth minimization problem for a large class of trees // Theory Comput. Syst. — 1997. — Vol. 30, N 1. — P. 67–90.
13. **Steinberg L.** The backboard wiring problem: a placement algorithm // SIAM. — 1961. — Vol. 3, N 1. — P. 37–50.

Забудский Геннадий Григорьевич,
e-mail: zabudsky@ofim.oscsbras.ru

Лаздин Артём Юрьевич,
e-mail: art.lagzdin@gmail.com

Статья поступила
31 мая 2010 г.

Переработанный вариант —
20 июня 2011 г.