НАХОЖДЕНИЕ КРИТИЧЕСКИХ ПУТЕЙ В СЕТЕВЫХ ГРАФИКАХ

В. С. МИХЕЛЬСОН

(МОСКВА)

Многие задачи комбинаторного анализа, теории автоматов, сетевого планирования сводятся к нахождению различных путей в сетевых графиках. Например, в системах планирования и управления комплексами работ в качестве динамической модели, отображающей процесс их выполнения, используется сетевой график. Сетевая модель изображается в виде графа, состоящего из дуг и вершин. Дуги соответствуют работам, вершины — событиям. Если за длину дуги считать продолжительность выполнения соответствующей работы, то длина наидлиннейшего — критического — пути соответствует сроку окончания всех работ, изображенных в сетевом графике. Обычно в сетевых графиках две соседние вершины соединяются не более чем одной дугой. Если же имеются параллельные работы, которые могут выполняться только одновременно, то в сети две соседние вершины могут соединяться более чем одной дугой. В этом случае сетевой график называется мультиграфом. Для нахождения работ (дуг), расположенных на напряженных (наидлиннейших) путях, приходится находить не только самый длинный путь мультиграфа. Иногда приходится строить пути различных длин, проходящие через заданные вершины и не проходящие через другие заранее известные. Такая задача возникает при разбиении сетевых графиков на части, для определения параметров больших сетей в ЭВМ с ограниченной памятью.

В данной статье приводятся алгоритмы для построения наибольших (критических) путей, проходящих через заданные вершины мультиграфа.

Приведем необходимые определения и обозначения. Пусть задан мультиграф G . Вершины мультиграфа будем обозначать буквами i, j, k с нижними индексами, а дуги — выражением вида (i, n, j) , где i, j — вершины, соединяемые этой дугой, n — номер дуги среди дуг, соединяющих эти вершины. Вершины i и j называются соответственно началом и концом дуги (i, n, j) . Дуги, соединяющие вершины i и j , назовем родственными, а их количество обозначим через $\beta(i, j)$.

Путем $L = [i_1, n_1, i_2, n_2, i_3, \dots, i_{s-1}, n_{s-1}, i_s]$ называется такая последовательность дуг $(i_1, n_1, i_2), (i_2, n_2, i_3), \dots, (i_{s-1}, n_{s-1}, i_s)$, в которой начало следующей дуги является концом предшествующей. Вершины i_1 и i_s являются началом и концом этого пути, а остальные вершины $i_\xi, 2 \leq \xi \leq s-1$ — промежуточными вершинами. Если $i_1 \equiv i_s$, то путь называется замкнутым. Будем рассматривать мультиграфы без замкнутых путей. Каждой дуге (i, n, j) мультиграфа сопоставим число $t(i, n, j) \geq 0$, называемое ее длиной.

Длиною пути $L = [i_1, n_1, i_2, \dots, i_{s-1}, n_{s-1}, i_s]$ называется число

$$d(L) = \sum_{\xi=1}^{s-1} t(i_\xi, n_\xi, i_{\xi+1}).$$

Рассмотрим правильно упорядоченную последовательность всех вершин мультиграфа G : $T = \{k_1, k_2, \dots, k_{N_1}\}$, обладающую тем свойством, что если в G существует путь, соединяющий вершину k_{j_1} с вершиной k_{j_2} , то в T вершина k_{j_1} записана левее вершины k_{j_2} .

Алгоритмы построения последовательности T приведены в [1, 2]. Они позволяют также обнаружить наличие замкнутых путей в G . Пусть заданы последовательность вершин $A = \{a_i\}$, $1 \leq i \leq n$, в которой они расположены аналогично их расположению в T , и множество вершин B , причем $A \cap B = \bigwedge$ ($\bigwedge$ — символ пустого множества).

Для любой вершины k_j определим множества

$$\bar{A}_j = A \cap \{k_1, k_2, \dots, k_j\}, \quad A_j = \bar{A}_j \cup k_j.$$

Число элементов множества $\bar{A}_j$, отличных от k_j , обозначим через ω_j .

Рассмотрим все пути, начинающиеся в a_1 , проходящие последовательно через вершины последовательности $\bar{A}_j$ и оканчивающиеся в k_j , но не проходящие через вершины из B . Наидлиннейший из этих путей назовем путем первого ранга, следующий по величине путь назовем путем второго ранга, и т. д. Путь r -го ранга обозначим через $L_r(A_j)$. Пути $L_{r_1}(A_j)$ и $L_{r_2}(A_j)$ отличаются один от другого или родственными дугами, или промежуточными вершинами, не принадлежащими A . Число таких путей обозначим через ρk_j .

Задача 1. Дан мультиграф G множества A и B . Пусть k_{j_0} — вершина в G и r — произвольное натуральное число. Требуется:

- а) определить, существует ли в G путь $L_r(A_{j_0})$;
- б) если в G имеется путь $L_r(A_{j_0})$, то построить его и определить его длину.

Решение. Пусть $L_r(A_{j_0}) = [a_1, \dots, k_j, n_j, k_{j_0}]$, тогда $[a_1, \dots, k_j] = L_s(A_j)$, где $s \leq r$. Поэтому $L_r(A_{j_0}) = [L_s(A_j), (k_j, n_j, k_{j_0})]$. Это значит, что для нахождения дуг, составляющих путь $L_r(A_{j_0})$, достаточно иметь для любых k_{j_0} и r находить k_j, n_j и s .

Вершину k_j назовем предпоследней, n_j — номером последней дуги в $L_r(A_{j_0})$ и s — рангом пути от a_1 до вершины k_j в $L_r(A_{j_0})$.

Приведем алгоритм $D(A_j, B)$ решения рассматриваемой задачи.

Алгоритм $D(A_j, B)$.

1. Составить последовательность T .
2. Составить для каждой вершины k_j , расположенной в T правее a_1 , множество $H(k_j)$ всех вершин мультиграфа, соединяющихся дугами с k_j .
3. Взять число $M \geq tN_2$, где $t = \max_{(i, n, j) \in G} t(i, n, j)$ и N_2 — количество дуг мультиграфа.

$$4. \text{ Положим } t_1'(k) = \begin{cases} M, & \text{если } k \equiv a_1; \\ 0, & \text{если } k \text{ расположено в } T \text{ левее } a_1. \end{cases}$$

5. Найти такой индекс i_0 , что вершина k_{i_0} из T совпадает с a_1 .
6. Рассмотреть в T следующую вершину.
7. Для этой вершины k_h взять $H(k_h) = \{k_{\gamma_1}, \dots, k_{\gamma_m}\}$.
8. Для каждого k_{γ_ξ} , где $1 \leq \xi \leq m$, взять все ранее составленные числа $t_e'(k_{\gamma_\xi})$, причем $1 \leq e \leq \rho(k_{\gamma_\xi})$.

$$9. \text{ Составить суммы } \sigma(e, \gamma_\xi, \alpha, h) = t_e'(k_{\gamma_\xi}) + t(k_{\gamma_\xi}, n_\alpha, k_h), \quad 1 \leq \xi \leq m, \quad 1 \leq e \leq \rho(k_{\gamma_\xi}),$$

$$1 \leq \alpha \leq \beta(k_{\gamma_\xi}, k_h) \downarrow$$

10. Расположить числа σ в порядке убывания

$$\sigma(e_1, \gamma_{\xi_1}, \alpha_1, h) \geq \sigma(e_2, \gamma_{\xi_2}, \alpha_2, h) \geq \dots \geq \sigma(e_c, \gamma_{\xi_c}, \alpha_c, h),$$

где

$$c = \rho(k_h) = \sum_{\xi=1}^m \rho(k_{\gamma_{\xi}}) \cdot \beta(k_{\gamma_{\xi}} k_h) \quad (1)$$

11. Положим

$$P_1(A_h, l) = \sigma(e_l, \gamma_{\xi_l}, \alpha_l, h), \quad P_2(A_h, l) = k_{\gamma_{\xi_l}}, \quad P_3(A_h, l) = n_{\alpha_l}, \quad P_4(A_h, l) = e_l$$

12. Вычислить

$$t_l'(k_h) = \begin{cases} \sigma(e_l, \gamma_{\xi_l}, \alpha_l, h) + M, & \text{если } k_h \in A, \\ \sigma(e_l, \gamma_{\xi_l}, \alpha_l, h) - M, & \text{если } k_h \in B, \\ \sigma(e_l, \gamma_{\xi_l}, \alpha_l, h), & \text{если } k_h \in \overline{A \cup B}. \end{cases}$$

$$1 \leq l \leq \rho(k_h)$$

13. Если рассмотрены все вершины последовательности A , то окончить вычисления. В противном случае переход к 6.

Теорема. Пусть k_{j_0} — заданная вершина ориентированного мультиграфа G и r — произвольное натуральное число.

1) В G имеется путь $L_r(A_{j_0})$ тогда и только тогда, когда $P_1(A_{j_0}, r) \geq \omega_{j_0} M$.

2) Если $P_1(A_{j_0}, r) \geq \omega_{j_0} M$, то $d(L_r(A_{j_0})) = P_1(A_{j_0}, r) - \omega_{j_0} M$, $P_2(A_{j_0}, r)$ — предпоследняя вершина в $L_r(A_{j_0})$, $P_3(A_{j_0}, r)$ — номер последней дуги в $L_r(A_{j_0})$, а ранг пути к предпоследней вершине в $L_r(A_{j_0})$ равен $P_4(A_{j_0}, r)$.

Доказательство. Докажем необходимость и достаточность первого утверждения теоремы.

Пусть в мультиграфе имеется путь $L_r(A_{j_0})$. Рассмотрим вершину k_{i_0+1} (см. пункт 5 алгоритма). Обозначим через $L_{r_1}(A_{i_0+1})$ часть пути $L_r(A_{j_0})$ между вершинами k_{i_0} и k_{i_0+1}

$$P_1(A_{i_0+1}, r_1) = \sigma(e_{r_1}, \gamma_{\xi_{r_1}}, \alpha_{r_1}, i_0+1) = t_1'(a_1) + t(a_1, n_{\alpha_{r_1}}, k_{i_0+1}) \geq M = \omega_{i_0+1} M.$$

Рассмотрим вершину k_c . Обозначим через r_i , $i_0 \leq i \leq c$ ранг части $L_r(A_{j_0})$ между вершинами k_{i_0} и k_i .

Допустим, что имеет место соотношение

$$P_1(A_j, r_j) \geq \omega_j M, \quad i_0 < j < c,$$

и докажем его справедливость при $j = c$.

Возьмем δ равным единице, если $k_{c-1} \in A$, и равным нулю в противном случае.

Из алгоритма следует, что

$$\begin{aligned} P_1(A_c, r_c) &= \sigma(r_{c-1}, c-1, \alpha_1, c) = t'_{r_{c-1}}(k_{c-1}) + t(k_{c-1}, n_{\alpha_1}, k_c) = \\ &= \sigma(r_{c-2}, c-2, \alpha_2, c-1) + \delta M + t(k_{c-1}, n_{\alpha_1}, k_c) \geq \\ &\geq P_1(A_{c-1}, r_{c-1}) + \delta M \geq \omega_{c-1} M + \delta M = (\omega_{c-1} + \delta) M = \omega_c M. \end{aligned}$$

Таким образом доказана необходимость первого утверждения теоремы.

Для всех вершин, расположенных в T , после a_1 при помощи алгоритма $D(A_j, B)$ определяются числа $P_i(A_j, l)$, $1 \leq i \leq 4$.

Рассмотрим путь

$$L = [k_{j_q}, n_{j_q}, k_{j_{q-1}}, \dots, k_{j_1}, n_{j_1}, k_{j_0}],$$

где

$$r_0 = r, k_{j_{i+1}} = P_2(A_{j_i}, r_i), \quad n_{j_{i+1}} = P_3(A_{j_i}, r_i),$$

$$r_{i+1} = P_4(A_{j_i}, r_i) \quad (0 \leq i \leq q-1).$$

Из алгоритма следует, что $P_1(A_{j_i}, r_i) = t'_{r_{i+1}}(k_{j_{i+1}}) + t(k_{j_{i+1}}, n_{j_{i+1}}, k_{j_i})$,

поэтому
$$P_1(A_{j_0}, r) = \sum_{i=0}^{q-1} t(k_{j_{i+1}}, n_{j_{i+1}}, k_{j_i}) + t'_1(k_{j_q}) + (\rho_1 - \rho_2)M,$$

где ρ_1 и ρ_2 — соответственно количество элементов из A и B в последовательности $k_{j_{q-1}}, \dots, k_{j_0}$.

Пусть $P_1(A_{j_0}, r) \geq \omega_{j_0}M$, так как $t'_1(a_1) = M$, то $\rho_1 = \omega_{j_0} - 1$, $\rho_2 = 0$ и $k_{j_q} = a_1$, таким образом путь L проходит через все вершины последовательности A_{j_0} и не проходит ни через одну из вершин B . Из пункта 10 алгоритма $D(A_j, B)$ следует, что ранг пути L равен r , поэтому $L \equiv L_r(A_{j_0})$. Таким образом, из $P_1(A_{j_0}, r) \geq \omega_{j_0}M$ следует, что в мультиграфе существует путь $L_r(A_{j_0})$.

Докажем второе утверждение теоремы.

Выше было показано, что $L \equiv L_r(A_{j_0})$. Это значит, что $P_2(A_{j_0}, r)$ является предпоследней вершиной, $P_3(A_{j_0}, r)$ является номером последней дуги и ранг пути к вершине k_{j_1} равен $P_4(A_{j_0}, r)$ для пути и $L_r(A_{j_0})$ и $d(L_r(A_{j_0})) = P_1(A_{j_0}, r) - \omega_{j_0}M$. Теорема доказана.

При помощи алгоритма $D(A_j, B)$ можно найти пути любого ранга r (если они существуют в G) до заданной вершины k_{j_0} .

Если же задается некоторое число r_0 и требуется найти все пути $L_q(A_j)$ при $1 \leq q \leq r_0$, то целесообразно после шага 10 выполнить шаг 10-а.

Для этого надо отбросить из (1) последнее $\rho(k_j)$ — $\min(\rho(k_j), r_0)$ членов. При этом можно несколько сократить объем запоминаемых чисел σ .

Алгоритм $D(A_j, B)$ можно легко видоизменить для определения кратчайших путей различных рангов, проходящих через A_j и не проходящих через B .

Составленный алгоритм можно применять только к мультиграфам, не содержащим замкнутых путей. Наличие таких путей определяется в алгоритме составления последовательности T .

В рассмотренном алгоритме составляется последовательность T и множества $H(k_j)$, которые рассматриваются в порядке следования k_j в T . Это позволяет определить числа $P_i(A_j, l)$, $1 \leq i \leq 4$ за один просмотр всех $H(k_j)$ (или дуг). Если не составлять последовательности T , т. е. не упорядочивать вершины и рассматривать все дуги в произвольном порядке, то приходится их просматривать до N_2 раз, определяя при этом каждый раз новые промежуточные значения $P_i(A_j, B)$.

Если $A = \{a_1, a_2\}$, где a_1 и a_2 — начальная и конечная вершины G и $B = \wedge$, то при помощи алгоритма $D(A_j B)$ определяются пути r -го ранга между начальной и заданной вершинами мультиграфа. Алгоритмы для нахождения таких путей в графе с упорядоченной нумерацией вершин приведены в [2], а с неупорядоченной нумерацией в [2, 3].

Рассмотрим еще одну задачу, связанную с нахождением путей различных рангов в мультиграфах.

Задача 2. Дан ориентированный мультиграф G . Пусть k_{j_0} вершина G и $d_1 \leq d_2$ положительные числа. Требуется найти все пути $L(A_{j_0})$, проходящие через A_{j_0} и не проходящие через B , для которых

$$d_1 \leq d(L(A_{j_0})) \leq d_2. \tag{2}$$

Решение. Пусть $r(k_j)$ такой индекс, что для чисел $P_1(A_j l)$ имеют место неравенства $P_1(A_j, r(k_j)) \geq \omega_j M$ и $P_1(A_j, r(k_j) + 1) < \omega_j M$. Из доказанной теоремы следует, что в мультиграфе G имеются ровно $r(k_j)$ путей, проходящих через A_j и не проходящих через вершины B . Очевидно, что $L_r(k_j)(A_j)$ — кратчайший из таких путей.

При решении рассматриваемой задачи для каждой вершины k_j находятся два числа $P_1(A_j, 1)$ и $P_1(A_j, r(k_j))$. Чтобы не определять всех чисел σ в последовательности (1), целесообразно для каждой вершины после составления (1) оставлять только $\sigma(e_1, \gamma_{\xi_1}, \alpha_1, j)$ и $\sigma(e_{r(k_j)}, \gamma_{\xi_r(k_j)}, \alpha_{r(k_j)}, j)$, вычислять только $t_1'(k_j)$ и $t_{r(k_j)}'(k_j)$.

В этом случае, если множество $H(k_j)$ состоит из m элементов, то для вершины k_j вычисляются не более $2 \cdot \sum_{\xi=1}^m \beta(k_{\gamma_{\xi}}, k_j)$ сумм, вместо $\sum_{\xi=1}^m \beta(k_{\gamma_{\xi}}, k_j) \rho(k_{\gamma_{\xi}})$.

Идея предлагаемого алгоритма состоит в следующем. Пусть вершина k_j лежит на искомом пути $L_r(A_{j_0})$ и $L_{r_j}(A_j)$ — часть этого пути между вершинами a_1 и k_j . Если выполняется соотношение (2), то

$$d_1 - \Delta_1(j_0, j) \leq d(L_{r_j}(A_j)) \leq d_2 - \Delta_2(j_0, j), \quad (3)$$

где $\Delta_1(j_0, j) = P_1(A_{j_0}, 1) - P_1(A_j, 1) - (\omega_{j_0} - \omega_j)M$ — разность длин максимальных путей, проходящих через A_{j_0} и A_j и не проходящих через B . $\Delta_2(j_0, j) = P_1(A_{j_0}, r(k_{j_0})) - P_1(A_j, r(k_j)) - (\omega_{j_0} - \omega_j)M$ — разность длин кратчайших путей, проходящих через A_{j_0} и A_j и не проходящих через B .

Последовательно рассматривая все вершины из таблицы T , мы будем для каждой из них находить все пути, длина которых удовлетворяет (3). Если для какой-либо вершины k_γ уже найдены все такие пути, то при их помощи можно построить все пути, проходящие через вершины из $A_{\gamma+1}$, длина которых также удовлетворяет (3) при $j = \gamma + 1$.

Алгоритм $D(A_j, B, d_1, d_2)$.

1. Для вершины k_{j_0} и всех вершин $k_j \in T$, расположенных между a_1 и k_{j_0} , определить $P_1(A_j, 1)$ и $P_1(A_j, r(k_j))$.
2. Найти такой индекс i_0 , что $k_{i_0} \equiv a_1$.
3. Рассмотреть следующую вершину в последовательности T .
4. Для рассматриваемой вершины k_h взять $H(k_h) = \{k_{\gamma_1}, \dots, k_{\gamma_m}\}$.
5. Для каждой вершины k_{γ_ξ} взять числа $t_c'(k_{\gamma_\xi})$, где $t_1'(a_1) = M$ и $t_1'(k_{\gamma_\xi}) = 0$, если k_{γ_ξ} расположено в T левее a_1 .
6. Составить суммы $\sigma(e, \gamma_\xi, \alpha, h) = t_e'(k_{\gamma_\xi}) + t(k_{\gamma_\xi}, n_\alpha, k_h)$.
7. Запомнить числа $\sigma(e^*, \gamma_{\xi^*}, \alpha^*, h)$, для которых выполняется неравенство

$$d_1 - \Delta_1(j_0, j) \leq \sigma(e^*, \gamma_{\xi^*}, \alpha^*, h) \leq d_2 - \Delta_2(j_0, j). \quad (4)$$

Если таких чисел σ нет, то прекратить вычисления: задача 2 не имеет решения.

8. Занумеровать числа σ : $\sigma_1, \dots, \sigma_i, \dots, \sigma_{r(k_h)}$.
9. Для каждого из запомненных чисел $\sigma_{i^*}(e^*, \gamma_{\xi^*}, \alpha^*, h)$ определить $P_1(A_h, i^*) = \sigma_{i^*}$, $P_2(A_h, i^*) = k_{\gamma_{\xi^*}}$, $P_3(A_h, i^*) = n_{\alpha^*}$, $P_4(A_h, i^*) = e^*$.

10. Вычислить для запомненных чисел

$$t_{i^*}(k_h) = \begin{cases} \sigma_{i^*} + M, & \text{если } k_h \in A, \\ \sigma_{i^*} - M, & \text{если } k_h \in B, \\ \sigma_{i^*}, & \text{если } k_h \in \overline{A \cup B}. \end{cases}$$

11. Если вершина k_{j_0} рассмотрена, то прекратить вычисления. В противном случае переход к пункту 3.

В результате применения этого алгоритма для некоторых вершин, расположенных в T между a_1 и k_{j_0} , и для вершины k_{j_0} определяются числа $P_i(A_j, l)$, $1 \leq i \leq 4$, по которым можно построить искомые пути.

Рассмотрим число q_0 , $1 \leq q_0 \leq r(k_{j_0})$ и путь

$$L = [k_{j_1}, n_{j_1}, k_{j_2}, \dots, k_{j_{p-1}}, n_{j_{p-1}}, k_{j_p}],$$

где

$$j_p = j_0, \quad i_p = i_0, \quad k_{j_c} = P_2(A_{j_{c+1}}, i_{c+1}),$$

$$n_{j_c} = P_3(A_{j_{c+1}}, i_{c+1}) \quad \text{и} \quad i_c = P_4(A_{j_{c+1}}, i_{c+1}).$$

Для пути L имеет место (2).

Покажем теперь, что любой путь в мультиграфе, для которого имеет место (2), можно построить при помощи найденных чисел $P_i(A_j, l)$, $1 \leq i \leq 4$.

Пусть путь L проходит через A , не проходит через вершины B и для него выполняется неравенство (2). Возьмем две соседние вершины k_{j_c} и $k_{j_{c+1}}$, через которые проходит этот путь. Пусть L_1, L_2 и L_3 — соответственно части пути L до вершины k_{j_c} , между вершинами k_{j_c} и $k_{j_{c+1}}$ и между вершинами $k_{j_{c+1}}$ и k_{j_0} . Тогда

$$d_1 \leq d(L) \leq d_2, \quad d(L_1) + d(L_2) = d(L) - d(L_3),$$

причем

$$\Delta_2(j_0, j_{c+1}) \leq d(L_3) \leq \Delta_1(j_0, j_{c+1}),$$

поэтому

$$d_1 - \Delta_1(j_0, j_{c+1}) \leq d(L_1) + d(L_2) \leq d_2 - \Delta_2(j_0, j_{c+1}). \quad (5)$$

Таким образом, если для какого-либо пути L справедливо неравенство (2), то для его части $[L_1, L_2]$ справедливо неравенство (4).

Так как $\Delta_2(j_{c+1}, j_c) \leq d(L_2) \leq \Delta_1(j_{c+1}, j_c)$, то из (5) следует, что $d_1 - \Delta_1(j_0, j_c) \leq d(L_1) \leq d_2 - \Delta_2(j_0, j_c)$, т. е. для пути L_1 выполняется неравенство (4). Это показывает, что для определения всех путей, удовлетворяющих (2), достаточно для каждой вершины находить и запоминать числа σ_i , для которых выполняется неравенство (4).

ЛИТЕРАТУРА

1. Г. С. Поспелов, А. И. Тейман. Автоматизация процессов управления работами больших систем или сложных комплексов. Изв. АН СССР, ОТН, Техническая кибернетика, 1963, № 4.
2. Д. И. Голенко, Н. А. Левин, В. С. Михельсон, Ч. Г. Найдов-Железов. Автоматизация планирования и управления разработками. Рига, Латгосиздат, 1965 (в печати).
3. R. Bellman, Kalaba. On K-th best policies. J. Industr. Appl. Math., 1960, vol. 8, No. 4.

Поступила в редакцию
12 X 1964