

НАХОЖДЕНИЕ КРИТИЧЕСКИХ УЗЛОВ ТРАНСПОРТНОЙ СЕТИ НА ОСНОВЕ ПОСТРОЕНИЯ ЗАМКНУТОЙ ОБЛАСТИ

Крыгин А.А.¹, Куприянов Б.В.²
(ФГБУН Институт проблем управления
им. В.А. Трапезникова РАН, Москва)

Рассматривается задача поиска критических узлов транспортной сети, решаемой с помощью максимизации обобщенной стоимости проезда. Известны два метода ее решения: полный перебор и сведение задачи к задаче линейного программирования. Предлагаемый в работе метод заключается в выделении замкнутого множества вершин, декомпозиции и сокращении вычислений при переборе вариантов.

Ключевые слова: транспортные сети, поиск критических узлов.

1. Введение

В научной литературе существует класс задач, связанных с транспортными сетями, описываемыми нагруженными графами. Ряд подобных задач решается с помощью оценки ущерба сети при удалении из графа сети подмножества ребер и нахождения критического подмножества, при котором ущерб достигает оптимума. Таковы, например, задачи оценки надежности и уязвимости транспортных сетей [4, 5, 6, 7, 9, 10, 11]. Количество вариантов удаления q ребер из графа, содержащего m ребер определяется как C_m^q . Трудоемкой частью оценки величины ущерба является вычисление матрицы минимальных стоимостей путей для каждой комбинации удаления ребер из графа.

Рассмотрим выраженный случай, представленный на рис. 1. Как правило, плотность автодорог внутри области существенно больше чем плотность межобластных дорог. На данном графе можно выделить три подграфа, которые являются

¹ Андрей Александрович Крыгин, к.т.н., (andreyakr@yandex.ru).

² Борис Васильевич Куприянов, к.т.н., (kuprianovb@mail.ru).

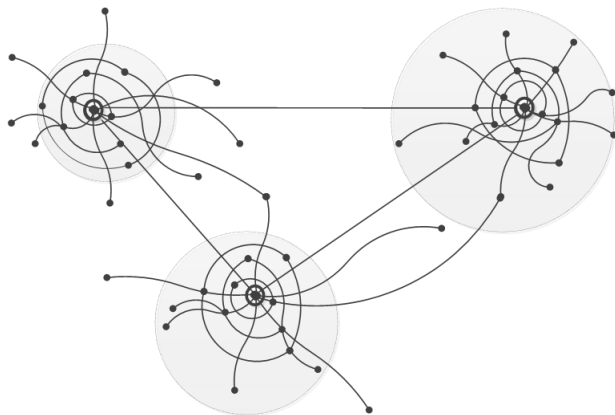


Рис. 1. Пример графа автотранспортной сети с тремя связанными областями.

изолированными в некотором смысле от остальной части графа.

В данной статье рассматривается метод выделения замкнутого множества вершин в графе и связанных с ними ребер. Данное выделение замкнутого множества позволяет декомпозировать и сократить вычисление матрицы путей минимальной стоимости.

2. Определения и постановка задачи

Пусть имеется неориентированный граф $G = (V, E)$, $n = |V|$, $m = |E|$ с взвешенными ребрами. Множеству ребер E соответствуют однородные и неветвящиеся участки дорог: мосты, туннели, автострады. Ребро графа, соединяющее вершины v_i и v_j , обозначим как (i, j) . Множеству вершин V соответствуют населенные пункты, перекрестки и границы смены категории дорог. Каждой паре вершин v_i и v_j ставится в соответствие два числа: $c_{i,j}$ и $d_{i,j}$. Величина $c_{i,j}$ характеризует стоимость поездки от узла v_i к узлу v_j , а $d_{i,j}$ характеризует потребность в движении (спрос на поездку), т.е. количество поездок от узла v_i к узлу v_j за единицу времени. В ра-

боте [1] приведена методика определения $c_{i,j}, d_{i,j}$. Величина $d_{i,j}$ не меняется при удалении ребер из графа; будем считать ее заданной. Величина $c_{i,j}$ определяется следующим образом. Для каждого ребра графа (i, j) задается его вес $w_{i,j}$, характеризующий длину соответствующего участка дороги. Путем в графе, соединяющим вершины v_i и v_j является последовательность ребер $(i, l_1), (l_1, l_2), \dots, (l_{q-1}, l_q), (l_q, j)$. Длиной пути является сумма весов всех его ребер. Путем минимальной стоимости, соединяющим вершины v_i и v_j является такой путь, что длина любого другого пути, соединяющего эти вершины, не меньше пути минимальной стоимости. Значение $c_{i,j}$ определяется как длина пути минимальной стоимости. Существуют эффективные алгоритмы вычисления минимальных стоимостей для всех пар вершин графа, например [2].

Рассматривается задача нахождения подмножества критических ребер Q для заданного $q = |Q|$. Будем считать, что задача решается методом полного перебора следующим образом. Для каждой из C_m^q комбинаций q ребер, составляющих множество Q , вычисляется стоимость

$$S(G(V, E \setminus Q)) = \sum_{i=1}^n \sum_{j=1}^{i-1} d_{i,j} c_{i,j}.$$

Где $c_{i,j}$ – минимальная стоимость пути между вершинами v_i, v_j при удалении из графа комбинации, $d_{i,j}$ – потребность в движении. Находится комбинация Q_0 , при которой $S(G(V, E \setminus Q_0))$ достигает своего максимума. Основная часть этой операции – вычисление матрицы C для графа $G(V, E \setminus Q)$ выполняется C_m^q раз. Задачей настоящего исследования является сокращение числа операций при вычислениях матрицы C .

3. Декомпозиция вычислений на основе замкнутого множества вершин

Выделим в графе подмножество вершин $S \subset V$. Определим подмножество Z , состоящее из внутреннего множества S и граничного множества P т.е. $Z = S \cup P$, следующим образом. Для любого ребра (i, j) , где $v_i \in S \rightarrow v_j \in Z$. Z таково, что все вершины S смежны только вершинам из Z . Соответственно, $P = Z \setminus S$ и обозначим $T = V \setminus S$.

Пример графа показан на рис.2. Некоторое подмноже-

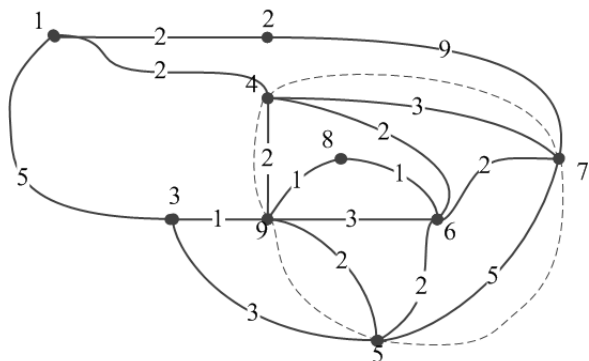


Рис. 2. Пример графа и некоторой области вершин Z .

ство вершин $Z = \{4, 5, 6, 7, 8, 9\}$ выделено пунктиром. Сама пунктирная линия проходит по граничным вершинам Z . Таким образом вершины множества $S = \{6, 8\}$ образуют внутреннее множество, а $P = \{4, 9, 5, 7\}$ образует граничное множество в Z . Пусть $n_z = |Z|$, $n_p = |P|$, $n_s = |S|$, в примере $n_z = 6$, $n_p = 4$ и $n_s = 2$. Очевидно, что $n_z = n_s + n_p$.

Переупорядочим номера вершин таким образом, что вершины из $V \setminus Z$ будут пронумерованы от 1 до $(n - n_z)$, далее будут идти вершины из P с номерами вершин от $(n - n_z + 1)$ до $(n - n_z + n_p)$, а в конце вершины из S с нумерацией от $(n - n_s + 1)$ до n . Таким образом, будет следующее разбиение $1, \dots, (n - n_z), (n - n_z + 1), \dots, (n - n_z + n_p), (n - n_s + 1), \dots, n$. С

учетом данной нумерации построим матрицу смежности W графа G

$$W_{i,j} = \begin{cases} \text{вес ребра } (i,j), & \text{если ребро } (i,j) \in E; \\ 0, & \text{если ребро } (i,j) \notin E. \end{cases}$$

Например, для графа на рис. 2 матрица смежности W представлена в таб. 1. Матрица W будет иметь структуру пока-

Таблица 1. Матрица смежности W

	1	2	3	4	5	6	7	8	9
1	0	2	5	2	0	0	0	0	0
2	2	0	0	0	0	0	9	0	0
3	5	0	0	0	3	0	0	0	1
4	2	0	0	0	0	2	3	0	2
5	0	0	3	0	0	2	5	0	2
6	0	0	0	2	2	0	2	1	3
7	0	9	0	3	5	2	0	0	0
8	0	0	0	0	0	1	0	0	1
9	0	0	1	2	2	3	0	1	0

занную в двух разрезах на рис. 3. В соответствии с дан-

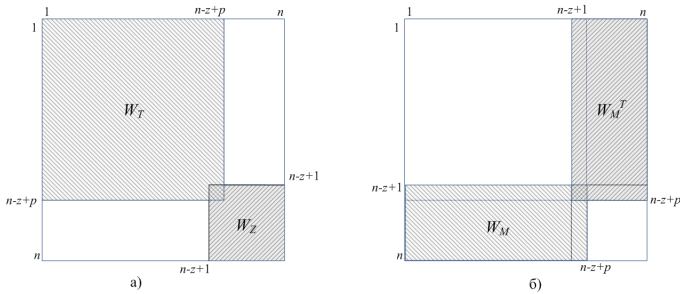


Рис. 3. Структура матрицы W в двух разрезах.

ным разбиением будем рассматривать подматрицы матрицы W : W_T , W_M , W_S , W_M^T . Каждая подматрица будет включать

строки и столбцы, соответствующие граничным вершинам. Квадратная матрица W_T будет иметь размерность $n - n_z + n_p$, матрица W_M будет иметь размерность $(n - n_z + n_p, n_z)$ и квадратная матрица W_Z будет иметь размерность n_z . Матрица W симметрична, поэтому в дальнейшем матрицу W_M^T рассматривать не будем, т.к. она является результатом транспонирования матрицы W_M .

Рассматривая веса ребер как стоимости, вычислим минимальные стоимости путей (далее просто минимальные стоимости) для всех пар вершин графа на основе матрицы W , например алгоритмом Флойда-Уоршалла [2]. Пусть C – матрица минимальных стоимостей путей, вычисленная на основе матрицы W . Матрице C будет соответствовать 4 "вырезанные из нее" подматрицы C_T , C_M , C_M^T , C_S . В виду симметрии матрицы C подматрицу C_M^T рассматривать не будем. В дальнейшем в статье будут рассматриваться граф $G = (V, E)$ и соответствующие ему подмножества вершин T, Z, P, S в описанном выше смысле.

Определение 1. Пусть граф $F = (V_F, E_F)$ является подграфом G т.е. $V_F \subset V$ и $E_F \subset E$. В этом случае назовем W_F квадратной вырезкой из матрицы W , если номера вершин V_F являются индексами строк и столбцов матрицы W . Т.е. $W_F = ||W_{i,j}||$ если $v_i, v_j \in V_F$. Определим функцию вырезки $Cut_F(W)$, т.е. $W_F = Cut_F(W)$.

Определение 2. Пусть для графа G определена матрица смежностей с весами ребер. Определим функцию $Ms(W)$ вычисляющую матрицу минимальных стоимостей для матрицы W .

Определение 3. Будем называть в графе $G = (V, E)$ подмножество вершин $Z \subset V$ замкнутым, если для каждой пары вершин множества Z существует путь минимальной стоимости, такой что все его вершины принадлежат множеству Z .

Рассмотрим матрицу $C_Z = Cut_Z(Ms(W))$, т.е. вырезку внутренней области Z из матрицы C (вычисленной по матрице W). Матрица минимальных стоимостей C представлена в

таб. 2, а матрица C_Z – в таб. 3.

Таблица 2. Матрица минимальных стоимостей C

	1	2	3	4	5	6	7	8	9
1	0	2	5	2	6	4	6	5	4
2	2	0	7	4	8	6	7	7	6
3	5	7	0	3	3	3	5	2	1
4	2	4	3	0	4	2	3	3	2
5	6	8	3	4	0	2	4	3	2
6	4	6	3	2	2	0	2	1	2
7	6	7	5	3	4	2	0	3	4
8	5	7	2	3	3	1	3	0	1
9	4	6	1	2	2	2	4	1	0

Таблица 3. Вырезка C_Z из матрицы минимальных стоимостей C

	4	5	6	7	8	9
4	0	4	2	3	3	2
5	4	0	2	4	3	2
6	2	2	0	2	1	2
7	3	4	2	0	3	4
8	3	3	1	3	0	1
9	2	2	2	4	1	0

Если на основании матрицы $W_Z = Cut_Z(W)$ представленной в таб. 4 вычислить матрицу C , то она совпадет с матрицей C_Z . Это справедливо в общем случае, благодаря следующему утверждению.

Утверждение 1. Пусть дан граф $G = (V, E)$ и подмножество вершин $Z \subset \bar{V}$. Чтобы множество Z было замкнутым необходимо и достаточно, чтобы

$$Ms(Cut_Z(W)) = Cut_Z(Ms(W)).$$

Таблица 4. Матрица W_Z

	4	5	6	7	8	9
4	0	0	2	3	0	2
5	0	0	2	5	0	2
6	2	2	0	2	1	3
7	3	5	2	0	0	0
8	0	0	1	0	0	1
9	2	2	3	0	1	0

Доказательство. Заметим, что при поэлементном сравнении,

$$\mathcal{M}s(\text{Cut}_Z(W)) \geq \text{Cut}_Z(\mathcal{M}s(W)),$$

так как в левой части минимальная стоимость вычисляется по подграфу. Докажем утверждение необходимости. Пусть Z замкнутое подмножество. Предположим, что

$$\mathcal{M}s(\text{Cut}_Z(W)) \neq \text{Cut}_Z(\mathcal{M}s(W)),$$

т.е. для некоторых $v_i, v_j \in Z$ соответствующие элементы матриц $\mathcal{M}s(\text{Cut}_Z(W))$ и $\text{Cut}_Z(\mathcal{M}s(W))$ различны. В этом случае в графе G между вершинами v_i, v_j найдется путь меньшей стоимости, чем в подграфе $G_Z = (V_Z, E_Z)$, т.е. Z не замкнутое множество.

Аналогично доказывается достаточность.

Утверждение доказано.

Матрица C однозначно определяется через матрицы C_T, C_M, C_Z . Покажем как сократить вычисления при нахождении каждой из этих матриц.

Так как Z замкнутое подмножество, то

$$C_Z = \mathcal{M}s(\text{Cut}_Z(W)),$$

т.е. вычисления проводятся по матрице меньшего размера.

Модифицируем матрицу W_T , заменив ее элементы, относящиеся к области P , т.е. $w_{i,j}$ такие, что $v_i, v_j \in P$ на соответствующие элементы матрицы C_Z , как показано на рис. 4.

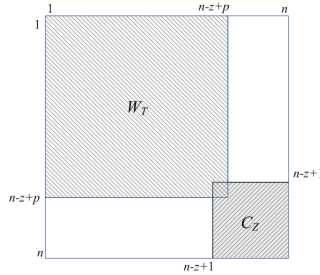


Рис. 4. Модификация матрицы W_T .

Утверждение 2. Для модифицированной описанным образом матрицы W'_T выполняется равенство:

$$\mathcal{M}s(W'_T) = \text{Cut}_T(\mathcal{M}s(W)).$$

Доказательство. Если кратчайший путь между вершинами $v_i, v_j \in V \setminus Z$ не проходит через вершины множества Z , то соответствующие элементы матриц $\mathcal{M}s(W'_T)$ и $\text{Cut}_T(\mathcal{M}s(W))$ совпадут. Рассмотрим случай, когда все кратчайшие пути между этими вершинами проходят через вершины множества Z . Преобразуем граф G так, что все вершины множества S

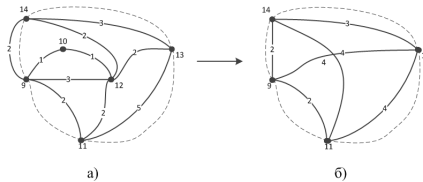


Рис. 5. Преобразование замкнутого множества.

и смежные с ними ребра удаляются из графа, а граничные

вершины дополняются ребрами до полного графа со стоимостями ребер, равными соответствующим минимальным стоимостям путей исходного графа, как на примере рис. 5. Пусть путь минимальной стоимости в исходном графе G проходит в замкнутой области через вершины $(a, b, \dots, c, d)$, где a и d граничные вершины. В преобразованном графе этот путь проходит по ребру (a, d) , поэтому его минимальная стоимость не изменится. С другой стороны матрица смежности преобразованного графа совпадает с W'_T . Утверждение доказано.

Это позволяет рассчитать матрицу C_T по матрице W'_T меньшей размерности.

Покажем как уменьшить вычисления при нахождении матрицы C_M : матрица кратчайших путей между вершиной $v_i \in (V \setminus Z)$ и вершиной $v_j \in S$. Определим функцию $\min p(i, j)$ вычисляющую минимальную стоимость пути между вершинами v_i и v_j . Любой путь между этими вершинами проходит через вершины из P , поэтому

$$(1) \quad \min p(v_i, v_j) = \min_{w \in P} (\min p(v_i, w) + \min p(w, v_j)).$$

Все значения $\min p(v_i, w)$ и $\min p(w, v_j)$ вычислены и имеются в матрицах C_T и C_S .

4. Алгоритм построения замкнутого множества

Алгоритм 1 (Построения замкнутого множества).

Исходные данные алгоритма: матрица смежностей W и начальное множество S_0 (внутренних вершин). Алгоритм строит замкнутое множество Z такое, что $S_0 \subseteq S$.

1) Построить множество P_0 – множество смежных с S_0 вершин, такое что $P_0 \cap S_0 = \emptyset$. Определим $Z = S_0 \cup P_0$.

2) Вычислить матрицу минимальных стоимостей $C = Ms(W)$ и матрицу $C_Z = Cut_Z(C)$.

3) Вычислить матрицу $C'_Z = Ms(Cut_Z(W))$.

4) Если $C_Z = C'_Z$ то множество Z является замкнутым и перейти к п. 6.

5) Пусть $C'_Z(i, j) \neq C_Z(i, j)$. На основании матрицы C вычислим множество вершин пути минимальной стоимости (см. алгоритм [2]) $\{v_i, \tilde{v}_1, \tilde{v}_2, \dots, \tilde{v}_k, v_j\}$. Включим множество вершин $\{\tilde{v}_1, \tilde{v}_2, \dots, \tilde{v}_k\}$ в множество Z . Перейти к п. 3.

6) Для найденного замкнутого множества Z определяется внутреннее множество S . Перенумеруем номера вершин так, чтобы индексы вершины Z превышали индексы остальных вершин. Вершина $v_k \in Z$ является внутренней, если $W(k, i) = 0$ для любого $1 \leq i \leq (n - n_z)$.

Конец алгоритма.

Так как по построению множество S_0 окружено границей P_0 , то $S_0 \subseteq S$.

Данный алгоритм находит решение не более, чем за $|V|$ итераций, однако в худшем случае множество S может совпадать с множеством V .

Пример 1. Рассмотрим в качестве исходных данных граф на рис. 2. Матрица смежностей представлена в таб. 1. Соответствующая матрица минимальных стоимостей представлена в таб. 2. Выберем в качестве начального внутреннего множества $S_0 = \{8\}$. Построим множество $P_0 = \{6, 9\}$ – граничных вершин, $Z = \{6, 8, 9\}$. Матрицы $Ms(Cut_Z(W))$ и $Cut_Z(Ms(W))$ совпадают и равны

	6	8	9
6	0	1	2
8	1	0	1
9	2	1	0

Поэтому Z – замкнутое множество и после перенумерации вершин (п. 6) вершина 8 определяется как внутренняя. •

5. Алгоритм нахождения критических ребер

Будем считать, что вычисление матрицы путей минимальной стоимости (кратчайших путей) C осуществляется с помощью алгоритма Флойда-Уоршелла [2]. Сложность этого

алгоритма зависит только от количества вершин, поэтому в дальнейшем фразу "вычисление матрицы C по матрице W размера n " будем понимать в смысле применения алгоритма Флойда-Уоршелла для матрицы смежности W размера $n \times n$ и получение в результате его работы матрицы C того же размера.

Алгоритм 2 (Нахождение критических ребер). Исходными данными алгоритма являются матрица весов W , матрица потребности в движении D и количество удаляемых ребер q .

1) Выполнить алгоритм вычисления замкнутого множества вершин, в частности алгоритм 1 вычислит первоначальную матрицу C для графа $G(V, E)$.

2) Выполнить в графе преобразование замкнутого множества и получить матрицу W'_T .

3) Для каждой из C_m^q комбинаций q ребер, составляющих множество Q , вычисляется обобщенная стоимость поездок

$$S(G(V, E \setminus Q)) = \sum_{i=1}^n \sum_{j=1}^{i-1} d_{i,j} c_{i,j}.$$

Основная часть этой операции – вычисление матрицы C для графа $G(V, E \setminus Q)$, распадается на два случая.

3.1) Если хотя бы одно из q ребер является ребром из замкнутого множества. В этом случае матрица C вычисляется обычным способом по матрице W размера n .

3.2) Если ни одно из q ребер не принадлежит замкнутому множеству. В этом случае матрица C для графа $G(V, E \setminus Q)$ однозначно определяется с помощью уже вычисленной матрицы C для графа $G(V, E)$ и вычислением матрицы C_T по матрице $W'_T(V, E \setminus Q)$, размера $n - n_s$.

4) Среди всех комбинаций определяются комбинации, имеющие максимальное значение обобщенной стоимости поездок. Ребра, составляющие эти комбинации, являются критическими.

Конец алгоритма.

Докажем справедливость этого алгоритма. При удалении хотя бы одного ребра из замкнутого множества, уже нельзя гарантировать, что это множество останется замкнутым, т.е. что все пути минимальной стоимости лежат внутри замкнутой области. Именно поэтому в п. 3.1 матрица C вычисляется обычным способом. Остается показать справедливость п. 3.2. В зависимости от того, к какому множеству относятся вершины i и j , существует 6 вариантов:

1. $i \in V \setminus Z, j \in V \setminus Z$,
2. $i \in V \setminus Z, j \in P$,
3. $i \in V \setminus Z, j \in S$,
4. $i \in P, j \in P$,
5. $i \in P, j \in S$,
6. $i \in S, j \in S$.

Значения элементов $c_{i,j}$ матрицы C в вариантах 4, 5, 6 уже получены при вычислении первоначальной матрицы C в п.1. Варианты 1,2,3 соответствуют элементам матриц C_T и C_M , которые вычисляются описанным выше способом.

6. Оптимизация алгоритма

Отметим, что единственный параметр, на который можно влиять при реализации предложенного алгоритма, является мощность множества S – внутренних вершин замкнутого множества. Действительно, можно построить замкнутое множество, добавить к внутреннему множеству еще одну вершину (например, одну из граничных) и построить следующее замкнутое множество большего размера, пользуясь тем же алгоритмом. Также понятно, что чем больше будет мощность множества S , тем меньшего размера будет матрица W_T и меньшее число операций потребуется для вычисления матрицы C . А с другой стороны, тем меньше будет количество комбинаций ребер, не входящих в замкнутое множество, то есть тем чаще придется вычислять матрицу C обычным способом по п. 3.1. Эти соображения позволяют предположить, что существует оптимальное значение n_s . Логично искать это оптимальное

значение, оценивая количество операций алгоритма и минимизируя ее по n_s . В общем виде это сделать невозможно по следующей причине. Для конкретного графа и заданного множества S , множество P определяется однозначно (если оно существует), при этом его мощность невозможно оценить, т.к. она зависит и от графа и от S . В тоже время, как будет видно ниже, сложность алгоритма является функцией от мощностей реберных и вершинных, внутренних и граничных множеств, а сами эти величины не являются независимыми и их зависимость неизвестна. Тем не менее, некоторый анализ функции сложности оказывается возможным.

Определим ребра замкнутого множества как ребра, соединяющие вершины из замкнутого множества и, по аналогии, ребра внутреннего и граничного множества. Введем следующие обозначения:

m_s – количество ребер внутреннего множества,

m_p – количество ребер граничного множества,

m_z – количество ребер множества Z .

При этом $m_z > m_s + m_p$, так как еще учитываются ребра, соединяющие вершины из S с вершинами из P .

В большинстве современных процессоров операции сложения и умножения занимают одинаковое количество тактов, поэтому будем оценивать только общее количество этих операций. Переменные m_s , n_p , m_p , n_z , m_z и переменная n_s , относительно которой ищется минимум сложности всего алгоритма, «участвуют» в пунктах 1, 3.1 и 3.2, поэтому будем оценивать сложность только этих пунктов. Ниже будет часто встречаться выражение сложности вычисления матрицы C , поэтому определим для него отдельную функцию. Обозначим $M(l)$, $l \in N$ – количество операций сложения и умножения, необходимых для нахождения матрицы кратчайших путей размера $l \times l$.

В алгоритме Флойда-Уоршалла элементы матрицы C определяются с помощью операции

$$\tilde{c}_{i,j} = \min_{k=1..l} (c_{i,j}, c_{i,k} + c_{k,j}),$$

которая проводится в тройном цикле, то есть l^3 раз. Вся операция состоит из одной операции сложения, поэтому $M(l) = l^3$.

Оценим сложность алгоритма вычисления замкнутого множества. Будем рассматривать наихудший случай и считать, что вершины в п. 5 добавляются по одной и первоначальное множество состоит из одной внутренней и одной граничной вершины. По окончании выполнения алгоритма мощность множества Z равняется n_z , то есть количество операций для вычисления матрицы C'_Z равно $\sum_{i=2}^{n_z} M(i)$. Необходимо оценить сложность операции в п. 5 по определению множества вершин пути минимальной стоимости между вершинами (x, y) . Так как мы рассматриваем наихудший случай и вершины добавляются по одной, то количество ребер в этом пути равно двум. Стоимость (длина) этого пути известна из исходной матрицы C (п. 1) и в среднем необходимо перебрать половину вершин, соседних и с x и с y , чтобы найти добавляемую вершину. Это вершина w такая, что сумма длин ребер $c_{x,w}$ и $c_{w,y}$ равняется стоимости кратчайшего пути. Тогда среднее количество операций сложения можно оценить как половину средней степени вершины. И общая сложность алгоритма вычисления замкнутого множества (обозначим ее Σ_1) оценивается как

$$\Sigma_1 = \sum_{i=2}^{n_z} \left(M(i) + \frac{m}{n} \right) = \frac{m(n_z - 1)}{n} + \sum_{i=2}^{n_z} M(i).$$

Рассмотрим пункты 3.1 и 3.2. Количество комбинаций ребер, соответствующее пункту 3.2 равно $C_{m-m_z}^q$, а пункту 3.1 – $(C_m^q - C_{m-m_z}^q)$. Тогда общая сложность п. 3.1 (обозначим ее Σ_2) оценивается как $\Sigma_2 = (C_m^q - C_{m-m_z}^q)M(n)$. Сложность п. 3.2 состоит из вычисления матрицы C_T по матрице $W'_T(V, E \setminus Q)$ размера $n - n_s$ и кроме одной группы случаев все элементы матрицы C определяются без дополнительных расчетов. Эта группа соответствует комбинациям $i \in V \setminus Z$, $j \in S$ и количество таких комбинаций вершин равно $(n - n_z)n_s$. Для каждой

комбинации необходимо найти

$$\min_{v_1 \in P} (\min p(i, v_1) + \min p(v_1, j))$$

по всем граничным вершинам. То есть необходимо выполнить n_p операций сложения. Общую сложность п. 3.2 (обозначим ее Σ_3) можно оценить так: $\Sigma_3 = C_{m-m_z}^q (M(n-n_s) + (n-n_z)n_sn_p)$. Суммируя результаты, получим, что количество операций сложения и умножения в рассматриваемых пунктах алгоритма нахождения критических ребер, можно оценить так. $\Sigma = \Sigma_1 + \Sigma_2 + \Sigma_3$, где

$$\begin{aligned} \Sigma_1 &= \frac{m(n_z - 1)}{n} + \sum_{i=2}^{n_z} M(i), \\ \Sigma_2 &= (C_m^q - C_{m-m_z}^q)M(n), \\ \Sigma_3 &= C_{m-m_z}^q (M(n - n_s) + (n - n_z)n_sn_p), \quad M(l) = l^3. \end{aligned}$$

Также отметим, что сложность алгоритма полного перебора (обозначим ее $\hat{\Sigma}$) можно по аналогии оценить как $\hat{\Sigma} = C_m^q M(n)$.

Выражение для Σ зависит от 4-х переменных, которые, в свою очередь, являются зависимыми друг от друга и эта зависимость неизвестна. Понятно, что методы математического анализа в такой ситуации не применимы.

Построим график Σ , приняв следующее допущение. Будем считать, что исходный граф транспортной сети регулярен с некоторой степенью приближения, то есть отношения количества ребер к количеству вершин всего графа и его достаточно большого подграфа примерно равны. Обозначим это отношение $\xi = m/n = m_s/n_s = m_p/n_p$ и перепишем, используя также равенство, выражение для Σ относительно n_s и n_p - количества вершин во внутренней и граничной области. При этом необходимо через n_s и n_p оценить величину $m - m_z$. Это множество ребер состоит из ребер, соединяющих внешние вершины, количество которых оценивается как $\xi(n - n_s - n_p)$ и множества ребер, соединяющих внешнюю и

граничную вершины. Оценим второе множество в наихудшем случае. С одной стороны, чем меньше будет размер внешней области, тем меньше будет количество комбинаций q ребер, лежащих во внешней области. С другой стороны, каждая граничная вершина по определению имеет хотя бы одно ребро, связанное с внешней вершиной. Поэтому оценим количество таких ребер величиной n_p . В результате получаем оценку $m - m_z \approx \xi(n - n_s - n_p) + n_p$.

После подстановки Σ_1 остается без изменений,

$$\begin{aligned}\Sigma_2 &= (C_m^q - C_{\xi(n-n_s-n_p)+n_p}^q)M(n), \\ \Sigma_3 &= C_{\xi(n-n_s-n_p)+n_p}^q (M(n - n_s) + (n - n_s - n_p)n_s n_p).\end{aligned}$$

В итоге получаем функцию $\Sigma(n_s, n_p)$ с параметрами n, q и ξ .

На рисунках 6 и 7 показаны графики этой функции в двух ракурсах, при различных значениях параметров.

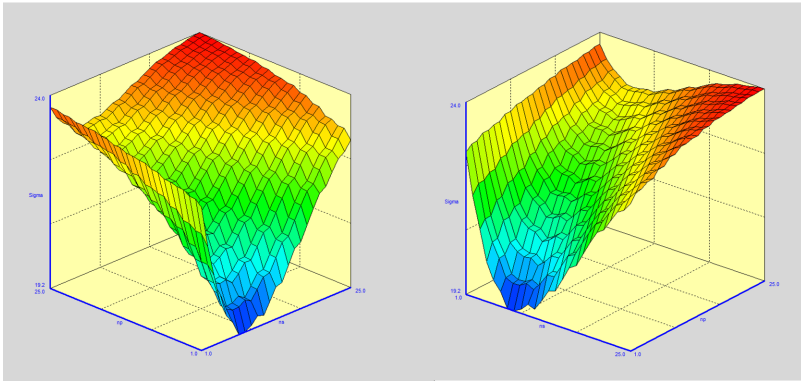


Рис. 6. График $\Sigma(n_s, n_p)$ при $n = 50$, $q = 4$, $\xi = 1.7$ в двух ракурсах.

Показанные на рисунках графики – типичные, то есть и при других значениях параметров наблюдается схожая картина: наличие «оврага», «дно» которого проходит примерно параллельно плоскости (Σ, n_p) . Примем это обстоятельство в

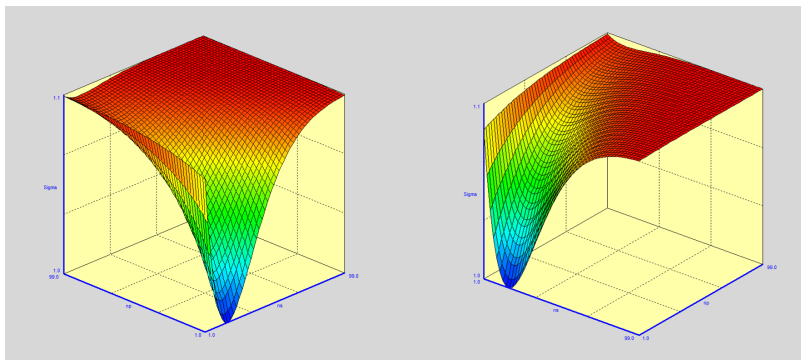


Рис. 7. График $\Sigma(n_s, n_p)$ при $n = 200$, $q = 10$, $\xi = 1.5$ в двух ракурсах.

качестве еще одного допущения. С его помощью становится возможным решить проблему зависимости переменных n_s и n_p . Обозначим n_s^0 - значение n_s , зависящее от n , q и ξ , в котором $\Sigma(n_s, n_p)$ достигает своего минимума при $n_p = 1$. В силу принятого допущения, для других фиксированных значений n_p , минимум $\Sigma(n_s, n_p)$ также будет достигаться при $n_s = n_s^0$. Тогда можно утверждать, что алгоритм нахождения критических ребер будет работать наиболее быстро, если замкнутое подмножество вершин удовлетворяет следующим условиям:

1. Мощность внутреннего множества примерно равна n_s^0 .
2. Мощность граничного множества минимальна.

Рассмотрим, как можно добиться выполнения каждого условия.

6.1. Мощность внутреннего множества

По заданному графу и величине q несложно определить значения n и ξ и численными методами найти n_s^0 . Чтобы ускорить эту операцию, был проведен анализ интервала значений n_s^0 при различных параметрах n , q и ξ с целью его уменьшения. Для этого фиксировался один из параметров и рассматривался трехмерный график относительно двух других. Было установлено, что интервал значений n_s^0 практически не

зависит от ξ . Диапазон изменения параметра ξ можно оценить интервалом (1,2): если граф сети представляет собой дерево, то ξ близко к единице; для наиболее плотных дорожных сетей (например, район Манхэттен в Нью-Йорке), граф которых практически является регулярным графом степени 4, $\xi = 2$.

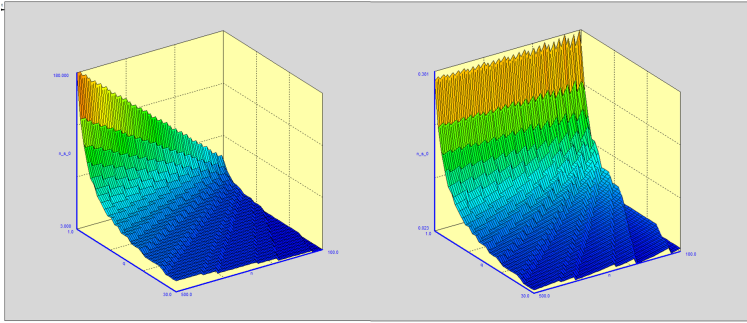


Рис. 8. График зависимости n_s^0 и $\frac{n_s^0}{n}$ при $\xi = 1.6$, $n = 100..500$, $q = 1..30$.

На рисунке 8 показан типичный график зависимости n_s^0 от n и q и, для наглядности, зависимость относительного значения $\frac{n_s^0}{n}$ от этих же параметров.

Анализ этих и схожих графиков привел к следующим выводам:

1) При любых значениях n , q и ξ , величина n_s^0 не превосходит $n/2$, причем максимальное значение достигается при $q = 1$. Этот вывод был также подтвержден полным перебором значений $\Sigma(n_s, n_p)$ при различных значениях параметров.

2) График зависимости $\Sigma(n_s, 1)$ от n_s при фиксированных n , q , ξ и $n_p = 1$ имеет локальные минимумы. На рис. 9 показан пример такого графика. Для наглядности ось ординат имеет относительную шкалу: вместо абсолютного значения Σ используются значения $\frac{\Sigma - \Sigma_{\min}}{\Sigma_{\min}}$.

Поэтому логично определять n_s^0 численными методами, вычисляя значения $\Sigma(n_s, 1)$, начиная с $n_s = 1$ до $n_s = n/2$.

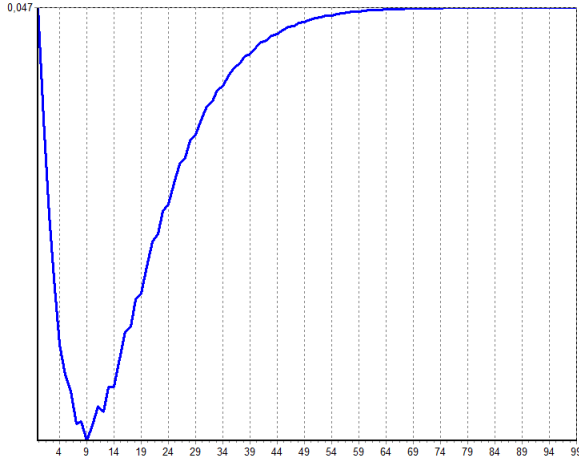


Рис. 9. График зависимости $\Sigma(n_s, 1)$ от n_s при $\xi = 1.6$, $n = 200..500$, $q = 20$.

6.2. Мощность граничного множества

В начале был проведен анализ области применимости предлагаемого метода. Для этого фиксировались параметры n , q , ξ и рассматривалась плоскость, на которой оси абсцисс соответствует значение n_s , а оси ординат - n_p . На этой плоскости отмечались точки (синим цветом), для которых выгоднее использовать метод полного перебора, то есть $\hat{\Sigma}(n_s, n_p) < \Sigma(n_s, n_p)$. На рис. 10 показан пример такой раскраски. Коричневой линией отмечено значение n_s^0 . Во всех

подобных графиках синяя область представляла собой прямоугольную трапецию, хотя во многих случаях она отсутствовала. То есть значение $\hat{n}_p$ такое, что для всех $n_p > \hat{n}_p$ выгоднее использовать метод полного перебора, превосходило n . Несложно показать, что $\Sigma_1 = O(n_z^4)$, что гораздо меньше $\hat{\Sigma} = O(m^q n^3)$. Поэтому логично перед решением конкретной задачи нахождения критических ребер определить n_s^0 , $\hat{n}_p$ и внутренне множество, так как это существенно не скажется на общем времени решения. Внутреннее множество необходи-

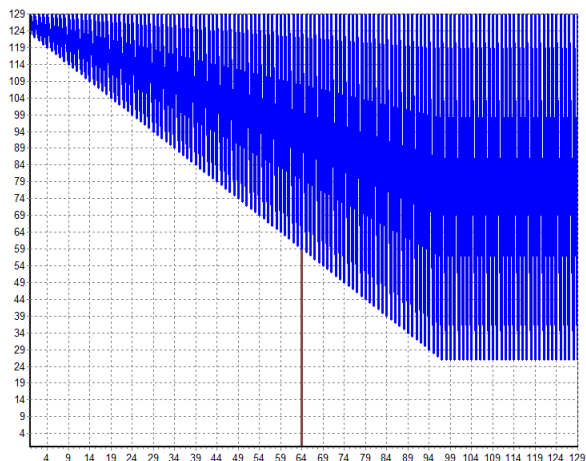


Рис. 10. Область применимости метода при $n = 130$, $q = 11$, $\xi = 1.6$.

мо построить так, чтобы его мощность приблизительно равнялась n_s^0 , а мощность граничного множества была как можно меньше, во всяком случае, не превосходила $\hat{n}_p$. Каких-либо общих рекомендаций по выбору начального множества S_0 для произвольного графа дать невозможно. Если рассматривать графы транспортных сетей, содержащие несколько административных областей (см. рис. 1), то для такого класса графов выбор внутреннего множества очевиден. Необходимо найти область с высокой плотностью автодорог, размера примерно n_s^0 .

7. Заключение

В работе рассматривается задача нахождения критических узлов транспортной сети и метод ее решения с помощью вычисления аргументов при которых функция обобщенной стоимости поездок достигает своего максимума. Наиболее трудоемкая операция в этом методе заключается в многократном вычислении матрицы длин кратчайших путей между все-

ми вершинами. Введено понятие замкнутой области и предложен метод сокращения вычислений этой матрицы. Метод основан на декомпозиции и последующих вычислений подматриц. Была приведена оценка количества операций и предложен метод построения замкнутой области, минимизирующей количество вычислений при решении задачи нахождения критических узлов.

Литература

1. КРЫГИН А.А., КУПРИЯНОВ Б.В. Определение критических узлов транспортной сети. – М.: Изд-во «УБС», 2023. – 24 с.
2. ЛИПСКИЙ В. Комбинаторика для программистов –М., "Мир", –1988.
3. ПАПАДИМИТРИУ Х.Х., СТАЙГЛИЦ К. Комбинаторная оптимизация. Алгоритмы и сложность МИР, Москва, –1982, –512.
4. BELL M.G.H., CASSIR C. Reliability of Transport Networks Baldock, Herts: Research Studies Press. –2000.
5. BELL M.G.H., IIDA Y. Transportation Network Analysis Chichester: John Wiley and Sons. –1997.
6. D'ESTE G.M., TAYLOR M.A.P. Modelling network vulnerability at the level of the national strategic transport network Journal of the Eastern Asia Society for Transportation Studies 4 (2), –2001. 1–14.
7. D'ESTE G.M., TAYLOR M.A.P. Network vulnerability: an approach to reliability analysis at the level of national strategic transport networks In The Network Reliability of Transport. Edited by Y. Iida and M.G.H. Bell. Oxford: Pergamon-Elsevier, –2003. 23–44.
8. GAOA L., LIUA X., LIUB Y., WANGC P. et al. Measuring Road Network Topology Vulnerability by Ricci Curvature // Physica A: Statistical Mechanics and its Applications. – 2019. – URL: <https://arxiv.org/abs/1811.05743>.
9. GUZE S. Graph Theory Approach to the Vulnerability of Transportation Networks Algorithms –2019, 12(12), 270.
10. JENELIUS E., PETERSEN T., MATTSSON L-G. Importance and exposure in road network vulnerability analysis Transportation Research Part A Policy and Practice. February 2006. 40(7):537–560

11. NEUMANN T., BEHRISCH M. Terminal reliability of road networks with multiple destination options // Int. J. of Safety and Security Eng. – 2018. – Vol. 8, No. 3. – P. 426–437.

ARTICLE TITLE

Andrey Krygin, Institute of Control Sciences of RAS, Moscow, Cand.Sc., (andreyakr@yandex.ru).

Boris Kupriyanov, Institute of Control Sciences of RAS, Moscow, Cand.Sc., (Moscow, Profsoyuznaya st., 65, (495)198-17-20 +1509).

Abstract: The problem of finding critical nodes in a transportation network, solved by maximizing the generalized travel cost, is considered. There are two known methods for solving it: exhaustive search and reducing the problem to a linear programming problem. The proposed method in this study involves identifying a closed set of vertices, decomposing and reducing calculations when considering different options.

Keywords: transport networks, critical object search.

УДК 338.49

ББК 39.311

Статья представлена к публикации
членом редакционной коллегии ...

Поступила в редакцию ...

Дата опубликования ...