

The algorithms for the Eulerian cycle and Eulerian trail problems for a multiple graph

A. V. Smirnov¹DOI: [10.18255/1818-1015-2023-3-264-282](https://doi.org/10.18255/1818-1015-2023-3-264-282)¹P.G. Demidov Yaroslavl State University, Sovetskaya str., 14, Yaroslavl, 150003, Russia.

MSC2020: 05C45, 05C65

Research article

Full text in Russian

Received August 13, 2023

After revision August 26, 2023

Accepted August 30, 2023

In this paper, we study undirected multiple graphs of any natural multiplicity $k > 1$. There are edges of three types: ordinary edges, multiple edges and multi-edges. Each edge of the last two types is a union of k linked edges, which connect 2 or $(k + 1)$ vertices, correspondingly. The linked edges should be used simultaneously. If a vertex is incident to a multiple edge, it can be also incident to other multiple edges and it can be the common end of k linked edges of some multi-edge. If a vertex is the common end of some multi-edge, it cannot be the common end of another multi-edge.

We set the problem of finding the eulerian walk (the cycle or the trail) in a multiple graph, which generalizes the classical problem for an ordinary graph. We formulate the necessary conditions for existence of an eulerian walk in a multiple graph and show that these conditions are not sufficient. Besides that, we show that the necessary conditions of existence of an eulerian cycle and eulerian trail are not mutually exclusive for an arbitrary multiple graph, that is why it is possible to construct a multiple graph where two types of eulerian walks exist simultaneously. Any multiple graph can be juxtaposed to the ordinary graph with quasi-vertices, which represents the structure of the initial graph in a simpler form. In particular, each eulerian walk in the multiple graph corresponds to the eulerian walk in the graph with quasi-vertices. The algorithm for getting such a graph is formulated. Also, the auxiliary problem of finding the covering trails with given endpoints in an ordinary graph is studied. Two algorithms are obtained for this problem. We elaborate the algorithm for finding the eulerian walk in a multiple graph, which has the exponential complexity. We suggest the polynomial algorithm for the special case of a multiple graph and show that the necessary conditions are sufficient for existence of an eulerian walk in this special case.

Keywords: multiple graph; multiple path; divisible graph; reachability set; covering trails; eulerian trail; eulerian cycle; graph with quasi-vertices

INFORMATION ABOUT THE AUTHORS

Alexander V. Smirnov	orcid.org/0000-0002-0980-2507 . E-mail: alexander_sm@mail.ru
corresponding author	PhD, Associate Professor, Department of Theoretical Computer Science.

Funding: Yaroslavl State University (project VIP-016).

For citation: A. V. Smirnov, "The algorithms for the Eulerian cycle and Eulerian trail problems for a multiple graph", *Modeling and analysis of information systems*, vol. 30, no. 3, pp. 264-282, 2023.

Алгоритмы для задач об эйлеровом цикле и эйлеровой цепи в кратном графе

А. В. Смирнов¹

DOI: [10.18255/1818-1015-2023-3-264-282](https://doi.org/10.18255/1818-1015-2023-3-264-282)

¹Ярославский государственный университет им. П.Г. Демидова, ул. Советская, 14, Ярославль, 150003, Россия.

УДК 519.17

Научная статья

Полный текст на русском языке

Получена 13 августа 2023 г.

После доработки 26 августа 2023 г.

Принята к публикации 30 августа 2023 г.

В статье рассматриваются неориентированные кратные графы произвольной натуральной кратности $k > 1$. Кратный граф содержит ребра трех типов: обычные, кратные и мультиребра. Ребра последних двух типов представляют собой объединение k связанных ребер, которые соединяют 2 или $(k + 1)$ вершину соответственно. Связанные ребра могут использоваться только согласованно. Если вершина инцидентна кратному ребру, то она может быть инцидентна другим кратным ребрам, а также она может быть общим концом k связанных ребер мультиребра. Если вершина является общим концом мультиребра, то она не может быть общим концом никакого другого мультиребра.

Ставится задача об эйлеровом маршруте (цикле или цепи) в кратном графе, которая обобщает классическую задачу для обычного графа. Сформулированы необходимые условия существования эйлерова маршрута в кратном графе, показано, что эти условия не являются достаточными. Кроме того, показано, что для произвольного кратного графа необходимые условия существования эйлерова цикла и эйлеровой цепи не являются взаимоисключающими, поэтому можно построить кратный граф, в котором одновременно существуют два вида эйлеровых маршрутов. Кратному графу сопоставляется обычный граф с квазивершинами, в упрощенном виде представляющий структуру исходного графа. В частности, каждому эйлерову маршруту в кратном графе соответствует эйлеров маршрут в графе с квазивершинами. Формулируется алгоритм построения такого графа. Также рассмотрена вспомогательная задача о покрывающих цепях с заданными концами в обычном графе, получены два алгоритма ее решения. Разработан алгоритм поиска эйлерова маршрута в кратном графе экспоненциальной трудоемкости. Для частного случая кратного графа предложен полиномиальный алгоритм, показано, что в этом частном случае необходимые условия существования эйлерова маршрута являются достаточными.

Ключевые слова: кратный граф; кратный путь; делимый граф; множество достижимости; покрывающие цепи; эйлерова цепь; эйлеров цикл; граф с квазивершинами

ИНФОРМАЦИЯ ОБ АВТОРАХ

Александр Валерьевич Смирнов
автор для корреспонденции

orcid.org/0000-0002-0980-2507. E-mail: alexander_sm@mail.ru
канд. физ.-мат. наук, доцент, кафедра теоретической информатики.

Финансирование: ЯрГУ (проект VIP-016).

Для цитирования: A. V. Smirnov, "The algorithms for the Eulerian cycle and Eulerian trail problems for a multiple graph", *Modeling and analysis of information systems*, vol. 30, no. 3, pp. 264-282, 2023.

Введение

В данной статье мы рассмотрим задачу об *эйлеровом маршруте* (цикле или цепи) в кратном графе. Кратные графы содержат три типа ребер (обычные, кратные и мультиребра) и являются обобщением обычных графов — по сути, обычный граф имеет кратность $k = 1$. Определения кратного графа кратности $k > 1$ и делимого кратного графа были сформулированы в статье [1].

Среди других известных обобщений графов наиболее близкими нам концепциями являются мультиграфы, гиперграфы (см., например, [2, 3]), а также метаграфы (см. [4, 5]). Действительно, как и в мультиграфах, в кратных графах допускается наличие нескольких ребер между парой вершин (набор таких ребер мы будем в дальнейшем называть *кратным ребром*), однако в случае кратного графа количество таких ребер должно быть строго равным k . В кратных графах присутствуют *мультиребра*, соединяющие между собой $(k + 1)$ вершину. Но в отличие от гиперребер гиперграфа, мультиребро представляется в виде k связанных ребер, имеющих один общий конец, причем все эти k ребер должны использоваться согласованно. По сути, понятие мультиребра близко понятию ребра между вершиной и метавершиной в метаграфе. При этом в метаграфе, напомним, метапуть между двумя метавершинами фактически моделирует причинно-следственные связи в некоторой предметной области. Однако в кратном графе используется принципиально иной подход к определению пути: *кратный путь* должен состоять ровно из k обычных путей, проходящих по обычным ребрам, а также по связанным ребрам кратных и мультиребер; при этом пути должны быть согласованы (одинаковы) на кратных и мультиребрах. Поэтому кратный граф нельзя считать частным случаем метаграфа.

Отметим также, что частным случаем кратного графа является кратная сеть (см. [6, 7]). Задача о наибольшем потоке в кратной сети обобщает классическую задачу (см. [8]) и имеет ряд приложений в сфере экономики, управления, финансов. В частности, кратные сети и потоки используются для поиска решения NP-трудной задачи целочисленного сбалансирования трех- и четырехмерной матрицы (см., например, [9, 10]).

Ранее мы обобщили для случая кратных графов задачи о кратчайшем пути между двумя вершинами и о минимальном остовном дереве (см., например, [1, 11]). При этом задача о кратчайшем кратном пути полиномиальна (см. [12]), а задача о минимальном остовном дереве является NP-трудной, по крайней мере, для графов кратности $k \geq 3$ (см. [13]).

В данной статье мы рассмотрим обобщение для кратных графов еще одной классической задачи теории графов — задачи об *эйлеровом маршруте* (см. [14]). Будут приведены необходимые условия существования такого маршрута в кратном графе. Отметим, что для обычного графа задача полиномиальна и разработан ряд полиномиальных алгоритмов (см., например, [15, 16]), однако для кратного графа мы получим алгоритм нахождения эйлерова маршрута, который в общем случае потребует экспоненциального количества шагов. Тем не менее, особенности построения кратного графа и эйлерова маршрута в нем позволяют предположить NP-трудность задачи. Также мы рассмотрим подкласс кратных графов, для которого можно построить полиномиальный алгоритм поиска эйлерова маршрута.

1. Кратные графы и деревья: необходимые определения

Напомним несколько определений, связанных с кратными графами и путями, которые ранее были сформулированы в статьях [1, 11].

Определение 1. *Кратный граф* G произвольной натуральной кратности $k > 1$ — это граф, вершины которого могут соединяться ребрами одного из 3 видов:

1. *Обычное ребро* e^o ; множество обычных ребер обозначим через E^o .

2. *Кратное ребро* e^k между двумя вершинами, которое состоит из k одинаковых связанных ребер; связанные ребра кратного ребра могут использоваться только согласованно; множество кратных ребер обозначим через E^k .
3. *Связанное ребро* e между двумя вершинами, имеющее один общий конец с другим $(k - 1)$ ребром (у любых двух из k связанных ребер только один конец является общим); множество связанных общей вершиной ребер будем называть *мультиребром* e^m ; связанные ребра мультиребра могут использоваться только согласованно; множество мультиребер обозначим через E^m .

Если вершина инцидентна какому-либо кратному ребру, то она может быть инцидентна другим кратным ребрам, а также она может быть общим концом какого-либо мультиребра.

Если вершина является общим концом какого-либо мультиребра, то она не может быть общим концом никакого другого мультиребра.

Если вершина является отдельным концом мультиребра или инцидентна обычному ребру, то она не может быть общим концом мультиребра и не может быть инцидентна кратному ребру.

Множества вершин и ребер графа G обозначим через V и E соответственно. Заметим, что $E = E^o \cup E^k \cup E^m$.

В данной статье рассматриваются только неориентированные кратные графы.

Рис. 1 и 2 иллюстрируют определение 1. В левой части рис. 1 кратное ребро представлено в виде объединения k одинаковых ребер между двумя вершинами, что показано штрихами. Равенство (или согласованность) связанных ребер предполагает, что все характеристики этих ребер (например, длина) одинаковы, и эти ребра могут использоваться только одновременно. Так, если осуществляется проход в определенном направлении по одному из связанных ребер, то одновременно с этим все остальные ребра проходятся в том же самом направлении. Кратное ребро может включаться в какие-либо новые структуры только целиком. В дальнейшем мы будем обозначать кратные ребра жирными линиями, как в правой части рис. 1.

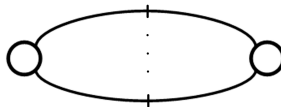


Fig. 1. Multiple edge



Рис. 1. Кратное ребро

В левой части рис. 2 мультиребро $\{x_0, \{x_1, \dots, x_k\}\}$ представлено в виде объединения k одинаковых ребер, связывающих общую вершину x_0 с k разными вершинами x_1, \dots, x_k . Как и на рис. 1, равенство ребер показано штрихами. Согласованность связанных ребер имеет тот же смысл, что и для кратных ребер. В дальнейшем мультиребра мы будем изображать при помощи расцепляющихся на k частей линий, как в правой части рис. 2.

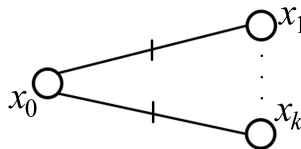


Fig. 2. Multi-edge

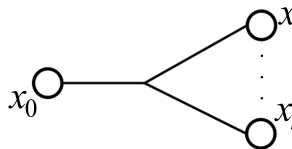


Рис. 2. Мультиребро

Определение 2. *Обычной вершиной* назовем вершину, которая инцидентна обычному ребру или является отдельным концом мультиребра.

Кратной вершиной назовем вершину, которая инцидентна кратному ребру или является общим концом мультиребра.

Из определения 1 следует, что множества обычных и кратных вершин не пересекаются. При этом кратная вершина может быть соединена с обычными только посредством мультиребра.

Определение 3. *Делимым кратным графом* назовем такой граф, в котором между двумя концами одного мультиребра не существует пути, проходящего только по обычным ребрам.

При удалении всех мультиребер делимый граф распадется на n компонент связности (связность здесь понимается в том же смысле, что и для обычных графов), каждая из которых содержит только кратные ребра либо только обычные ребра. При этом связанные ребра каждого мультиребра можно пронумеровать от 1 до k таким образом, что каждой компоненте связности, содержащей только обычные ребра, будут инцидентны связанные ребра мультиребер с одинаковыми номерами.

Определение 4. *Частью G_i ($i \in \overline{1, k}$) делимого графа $G(V, E)$ назовем подграф, содержащий связанные ребра с номером i всех кратных и мультиребер, а также компоненты связности, состоящие из обычных ребер и инцидентные i -ым связанным ребрам всех мультиребер.*

Каждая часть G_i является обычным графом. При этом возможность выделения частей G_i является особенностью делимых графов. В общем случае получить части G_i не удастся.

Пример 1. Рассмотрим представленный на рис. 3 кратный граф G кратности 2 со следующими множествами обычных, кратных и мультиребер (вершины будем обозначать их номерами):

$$\begin{aligned} E^k &= \{\{1, 3\}, \{1, 4\}, \{2, 3\}, \{2, 4\}, \{3, 4\}, \{15, 16\}\}; \\ E^m &= \{\{3, \{5, 13\}\}, \{4, \{8, 10\}\}, \{15, \{6, 14\}\}, \{16, \{9, 11\}\}\}; \\ E^o &= \{\{5, 6\}, \{5, 7\}, \{5, 8\}, \{6, 7\}, \{6, 9\}, \{7, 8\}, \{7, 9\}, \{8, 9\}, \{10, 11\}, \\ &\quad \{10, 12\}, \{10, 13\}, \{11, 12\}, \{11, 14\}, \{12, 13\}, \{12, 14\}, \{13, 14\}\}. \end{aligned}$$

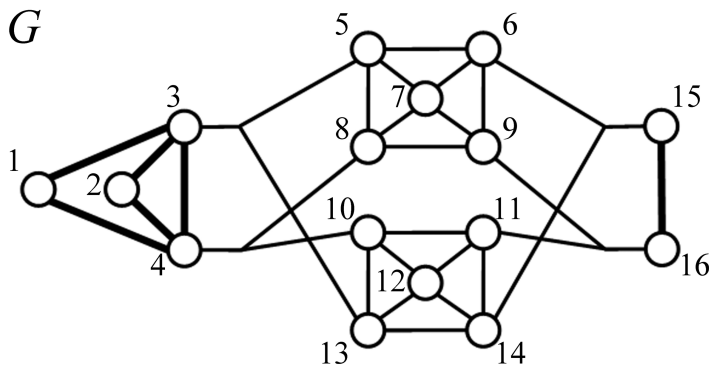


Fig. 3. Divisible graph of multiplicity 2

Рис. 3. Делимый граф кратности 2

Граф G является делимым. Части G_1 и G_2 этого графа показаны на рис. 4, связанные ребра всех кратных и мультиребер изображены пунктирными линиями.

Заметим, что граф перестанет быть делимым, если добавит в него обычное ребро между любой парой вершин из множеств $\{5, 6, 7, 8, 9\}$ и $\{10, 11, 12, 13, 14\}$.

Определим теперь путь в кратном графе.

Определение 5. $S(x, y) = \cup_{i=1}^k S^i(x, y)$ является *кратным путем* из вершины x в вершину y в графе $G(V, E)$, если выполнены следующие условия:

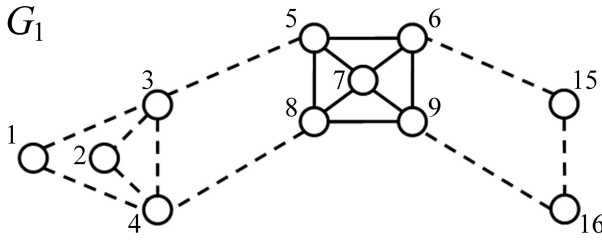


Fig. 4. Partition of a divisible graph

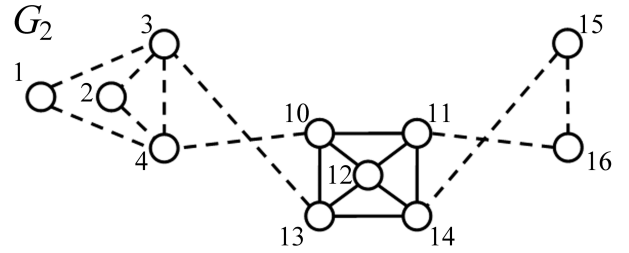


Рис. 4. Части делимого графа

1. $S^i(x, y) = (\{x, v_1^i\}, \{v_1^i, v_2^i\}, \dots, \{v_{l_i-1}^i, v_{l_i}^i\}, \{v_{l_i}^i, y\})$, где $l_i \geq 0$, — последовательность ребер, представляющая собой обычный (некратный) путь из x в y , где каждое ребро $\{a, b\}$ является либо обычным ребром в графе $G(V, E)$, либо i -ым связанным ребром кратного или мультиребра. Значения l_i и l_j ($i \neq j$) не согласовываются и могут быть как равными, так и различными. Если в путь $S(x, y)$ не входит ни одного кратного или мультиребра, то $S^2(x, y) = S^3(x, y) = \dots = S^k(x, y) = \emptyset$.
2. Любая обычная вершина может встретиться в $S^i(x, y)$ несколько раз, то есть $S^i(x, y)$ может содержать циклы.
3. Никакая кратная вершина не может встретиться в $S^i(x, y)$ дважды.
4. Любое обычное ребро может встречаться в $S^i(x, y)$ несколько раз, причем направления, в которых оно проходится в разных вхождениях, могут не совпадать.
5. Обычное ребро, входящее в $S^i(x, y)$, может также входить в любой $S^j(x, y)$, $j \neq i$.
6. Все пути $S^i(x, y)$ согласованы (одинаковы) на общей части. Это условие означает, что если связанное ребро какого-то кратного или мультиребра входит в некоторый путь $S^i(x, y)$, то остальные связанные ребра должны входить во все $S^j(x, y)$, $j \neq i$ (по одному связанному ребру в каждый $S^j(x, y)$). При этом порядок вхождения всех кратных и мультиребер во все $S^i(x, y)$ одинаков.
Фактически это значит, что если e_1 и e_2 — это два ребра пути $S(x, y)$, каждое из которых либо кратное, либо мультиребро, и в проекции $S^i(x, y)$ связанное ребро из e_1 проходится раньше связанного ребра из e_2 , то во всех остальных проекциях $S^j(x, y)$ связанные ребра из e_2 могут проходиться только после связанных ребер из e_1 .
7. Если $S(x, y)$ содержит мультиребро $\{x_0, \{x_1, \dots, x_k\}\}$, проходимое в направлении от общего конца, то он не может содержать никакого другого мультиребра $\{y_0, \{x_1, \dots, x_k\}\}$, проходимого в том же направлении. Аналогичное условие должно выполняться и в случае движения к общему концу.

Определение 6. Кратный путь $S(x, y)$ является *кратным циклом*, если $x = y$ и $S(x, y) \neq \emptyset$.

Пример 2. Проиллюстрируем определение кратного пути. Для этого рассмотрим граф, показанный на рис. 3, и построим в нем один из многочисленных возможных кратных путей $S(1, 2)$ из вершины 1 в вершину 2. Он состоит из двух обычных путей:

$$S^1(1, 2) = (\{1, 3\}, \{3, 5\}, \{5, 6\}, \{6, 9\}, \{9, 16\}, \{16, 15\}, \{15, 6\}, \{6, 5\}, \{5, 7\}, \{7, 8\}, \{8, 4\}, \{4, 2\});$$

$$S^2(1, 2) = (\{1, 3\}, \{3, 13\}, \{13, 14\}, \{14, 11\}, \{11, 16\}, \{16, 15\}, \{15, 14\}, \{14, 11\}, \{11, 12\}, \{12, 13\}, \{13, 10\}, \{10, 4\}, \{4, 2\}).$$

Связанные ребра кратных и мультиребер отмечены подчеркиванием. Двойным подчеркиванием в пути $S^1(1, 2)$ отмечено обычное ребро $\{5, 6\}$, которое в этом пути проходится дважды, но в противоположных направлениях. Соответственно, в пути $S^2(1, 2)$ двойным подчеркиванием отмечено обычное ребро $\{14, 11\}$, которое также проходится дважды, но в одном и том же направлении. Таким образом, в пути $S^1(1, 2)$ содержится обычный цикл $(6, 9, 16, 15, 6)$, а в пути $S^2(1, 2)$ содержатся обычные циклы $(14, 11, 16, 15, 14)$ и $(13, 14, 11, 16, 15, 14, 11, 12, 13)$, а вершины 5, 6, 11, 13, 14 проходятся дважды. Однако кратный путь $S(1, 2)$ не содержит в себе кратных циклов, как и должно быть (ни одна кратная вершина не проходится дважды). Полученный кратный путь $S(1, 2)$ показан на рис. 5. Части $S^1(1, 2)$ и $S^2(1, 2)$ этого пути представлены на рис. 6.

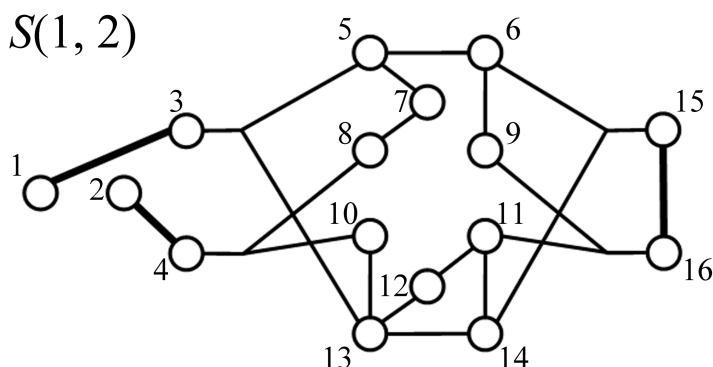


Fig. 5. Multiple path in the multiple graph

Рис. 5. Кратный путь в делимом графе

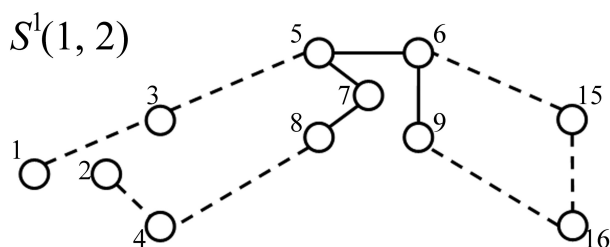


Fig. 6. Partition of the multiple path

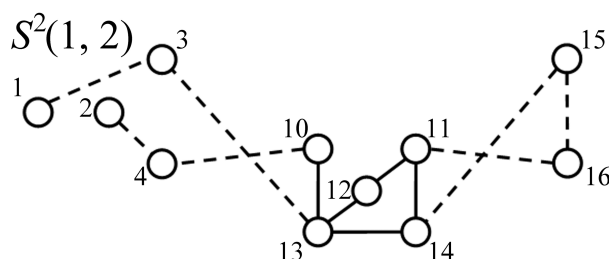


Рис. 6. Части кратного пути

Отметим, что при замене в кратном пути $S(1, 2)$ ребра $\{4, 2\}$ на ребро $\{4, 1\}$ мы получим кратный цикл.

Определение 7. Кратный граф $G(V, E)$ является *связным*, если одновременно выполнены два условия:

1. Кратный путь $S(x, y)$ существует для любых двух кратных вершин $x \in V, y \in V$.
2. Невозможно выделить такой подграф $G' \subset G$, который будет содержать только обычные ребра, и при этом подграфы G' и $G \setminus G'$ не будут соединены ни одним ребром (обычным ребром или связанным ребром мультиребра).

В отличие от обычных графов, связность кратного графа не предполагает наличие кратных путей из каждой вершины в каждую. Фактически в связном кратном графе между каждой парой вершин должен существовать обычный (некратный) путь, использующий связанные ребра кратных и мультиребер несогласованно, а кратные пути обязательно должны существовать только для пар кратных вершин.

Для делимого кратного графа определение связности может быть переписано в более простой форме, что обусловлено структурой графа.

Определение 8. Делимый кратный граф $G(V, E)$ является *связным*, если одновременно выполнены два условия:

1. Кратный путь $S(x, y)$ существует для любых двух кратных вершин $x \in V, y \in V$.
2. Каждая из частей G_i является связным (некратным) графом.

Определение 9. Множеством *достижимости по кратным ребрам* для некоторой кратной вершины x назовем множество R_x^k всех вершин y таких, что существует путь из x в y , проходящий только по кратным ребрам.

Будем обозначать через G_x^k подграф, образованный всеми вершинами из R_x^k и всеми кратными ребрами $e \in E^k$, которые соединяют вершины из R_x^k .

Определение 10. Множеством *достижимости по обычным ребрам* для некоторой обычной вершины x назовем множество R_x^o всех вершин y таких, что существует путь из x в y , проходящий только по обычным ребрам.

Будем обозначать через G_x^o подграф, образованный всеми вершинами из R_x^o и всеми обычными ребрами $e \in E^o$, которые соединяют вершины из R_x^o .

Очевидно, что $x \in R_x^k, x \in R_x^o$. Если $y \in R_x^k$, то $R_y^k = R_x^k$. Если $y \in R_x^o$, то $R_y^o = R_x^o$.

Отметим, что все множества *достижимости* R_x^k и R_x^o могут быть найдены с помощью быстрых полиномиальных алгоритмов, приведенных в статье [1]. Там же сформулированы полиномиальные алгоритмы проверки связности кратного графа.

Определение 11. Для любой вершины $x \in V$ определена ее *степень* $\deg x$ — количество обычных или связанных ребер инцидентных x .

Таким образом, каждое обычное ребро вида $\{x, y\}$ добавляет 1 к $\deg x$, каждое кратное ребро вида $\{x, y\}$ добавляет k к $\deg x$, а каждое мультиребро вида $\{x, \{y_1, \dots, y_k\}\}$ добавляет k к $\deg x$ и 1 к каждому из $\deg y_i$. Очевидно, что степень любой кратной вершины будет кратна k .

2. Постановка задачи об эйлеровом маршруте в кратном графе

Пусть имеется связный кратный граф $G(V, E)$ кратности $k > 1$. Так же, как и для обычного графа, для него можно определить понятие эйлерова маршрута.

Определение 12. *Эйлеровым маршрутом* μ в кратном графе $G(V, E)$ назовем такой обход графа $G(V, E)$, в котором каждое ребро из E встречается ровно один раз, а связанные ребра каждого кратного и мультиребра из $E^k \cup E^m$ используются только согласованно (одновременно).

Как и в определении 5, для эйлерова маршрута выполнено: $\mu = \bigcup_{i=1}^k \mu_i$, то есть каждый эйлеров маршрут μ в кратном графе представляет собой объединение k обычных маршрутов μ_i ($i \in \overline{1, k}$), в каждом из которых присутствует ровно одно связанное ребро каждого кратного и мультиребра, причем порядок обхода связанных ребер одинаков во всех μ_i .

Определение 13. Замкнутый эйлеров маршрут в кратном графе (начальная вершина равна конечной) называется *эйлеровым циклом*.

Незамкнутый эйлеров маршрут в кратном графе называется *эйлеровой цепью*.

Кратный граф назовем *эйлеровым*, если в нем существует эйлеров маршрут (цикл или цепь).

Если сравнить определение кратного пути и эйлеровой цепи в кратном графе, можно заметить, что эйлерова цепь — это, по сути, кратный путь, в котором допускается использование нескольких

мультиребер с одинаковыми множествами конечных вершин, проходимых в одном направлении, а также повторное использование кратных вершин, однако не допускается повтор встречавшихся ранее обычных ребер. Точно те же допущения относятся к эйлерову циклу в кратном графе в сравнении с кратным циклом.

Построение эйлерова цикла всегда можно начинать с кратной вершины. Эйлерову цепь в большинстве случаев также можно начинать строить с кратной вершины, однако если граф не будет делимым, эйлерова цепь может начинаться и в обычной вершине. Для этого необходимо, чтобы в графе было мультиребро $e^m = \{x_0, \{x_1, \dots, x_k\}\}$, все концы которого принадлежат одному множеству достижимости: $x_i \in R_{x_1}^o$ ($i \in 1, k$).

Пример 3. Снова рассмотрим кратный граф из примера 1 (рис. 3). Нетрудно убедиться, что в этом графе существует эйлеров цикл C , представленный на рис. 7.

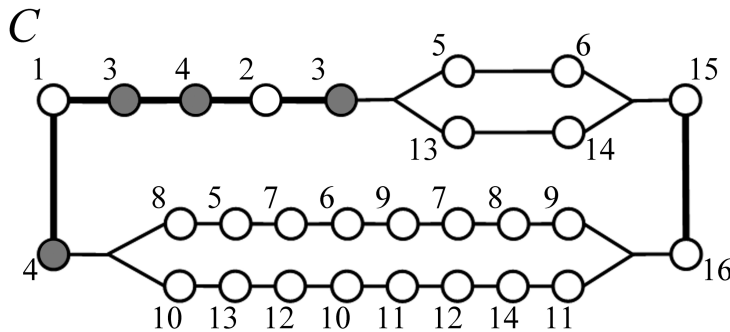


Fig. 7. Eulerian cycle in the multiple graph

Рис. 7. Эйлеров цикл в кратном графе

Следует отметить, что в эйлеровом цикле C две кратные вершины 3 и 4 проходятся дважды (на рисунке отмечены серым), поэтому цикл C является эйлеровым маршрутом, но не является кратным циклом согласно определению 6.

Задача 1 (эйлеров маршрут в кратном графе). Для данного кратного графа $G(V, E)$ установить, существует ли в нем эйлеров маршрут (цикл или цепь), и найти этот маршрут.

3. Необходимые условия существования эйлерова маршрута в кратном графе

Сформулируем необходимые условия существования эйлерова маршрута в кратном графе, которые будут во многом похожи на аналогичные условия для обычных графов (но с поправкой на особенности построения кратных графов). Однако, в отличие от последних, эти условия не будут достаточными. Напомним, что в обычном графе эйлеров цикл существует тогда и только тогда, когда все вершины графа имеют четную степень, а эйлерова цепь существует тогда и только тогда, когда в графе есть ровно две вершины нечетной степени (см., например, [17]).

Теорема 1. Если в кратном графе $G(V, E)$ существует эйлеров цикл, то все его обычные вершины имеют четную степень, а степень каждой кратной вершины v может быть представлена как $\deg v = k \cdot a_v$, где k — это кратность графа, a_v — четное число.

Справедливость утверждения теоремы следует из того, что, обходя эйлеров цикл, мы зайдём в каждую вершину столько же раз, сколько и выйдем, при этом для кратных вершин каждая операция захода или выхода будет выполняться одновременно для k связанных ребер кратного или мультиребра.

Теорема 2. Если в делимом кратном графе $G(V, E)$ существует эйлерова цепь, то одновременно выполнены следующие условия:

- все обычные вершины имеют четную степень;
- степень двух кратных вершин x, y представляется в виде $\deg x = k \cdot a, \deg y = k \cdot b$, где k — это кратность графа, a и b — нечетные числа;
- степень всех остальных кратных вершин v представляется в виде $\deg v = k \cdot c_v$, где k — это кратность графа, c_v — четное число.

Теорема 3. Если в неделимом кратном графе $G(V, E)$ существует эйлерова цепь, то найдутся две вершины x, y , для которых выполняется одно из условий:

1. x, y — кратные вершины; $\deg x = k \cdot a, \deg y = k \cdot b$, где k — это кратность графа, a и b — нечетные числа.
2. x — кратная вершина, y — обычная вершина и найдется мультиребро $\{y_0, \{y_1, \dots, y_k\}\} \in E^m$ такое, что $y_i \in R_y^o$ ($i \in \overline{1, k}$); $\deg x = k \cdot a, \deg y = k + b$, где k — это кратность графа, a — нечетное число, b — четное неотрицательное число.
3. x, y — обычные вершины и найдутся два мультиребра $\{x_0, \{x_1, \dots, x_k\}\}, \{y_0, \{y_1, \dots, y_k\}\} \in E^m$ такие, что $x_i \in R_x^o, y_i \in R_y^o$ ($i \in \overline{1, k}$); $\deg x = k + a, \deg y = k + b$, где k — это кратность графа, a и b — четные неотрицательные числа.

Степень всех остальных обычных вершин четна, а степень всех остальных кратных вершин v представляется в виде $\deg v = k \cdot c_v$, где k — это кратность графа, c_v — четное число.

Суммарное количество операций захода и выхода для начальной и конечной вершины эйлеровой цепи при ее обходе будет нечетно, для остальных же вершин это значение четно (как в теореме 1), отсюда следует справедливость теорем 2 и 3.

Более сложное условие теоремы 3 обусловлено тем, что в делимом графе кратная цепь обязательно начинается и заканчивается в кратной вершине, а вот для неделимого графа возможна ситуация, когда кратная цепь начинается или заканчивается в обычной вершине, причем в такой ситуации мы должны взять одновременно k ребер инцидентных этой вершине.

Следствие 1. Если делимый кратный граф $G(V, E)$ является эйлеровым, то эйлеровой является и каждая его часть G_i ($i \in \overline{1, k}$).

Сведем кратный граф $G(V, E)$ к обычному графу $G_{ord}(V_{ord}, E_{ord})$, структура которого будет отражать структуру исходного графа. Часть вершин будут соответствовать множествам подграфов G_x^o . Такие вершины мы будем называть квазивершинами, а граф G_{ord} — графом с квазивершинами.

Алгоритм 1 (построение графа с квазивершинами).

1. Для каждой кратной вершины $x \in V$ создадим обычную вершину $x \in V_{ord}$.
2. Для каждого кратного ребра $\{x, y\} \in E^k$ создадим обычное ребро $\{x, y\} \in E_{ord}$.
3. Найдем все множества достижимости R_x^o , в качестве x будем указывать минимальный номер вершины из этого множества.
4. Рассмотрим поочередно все мультиребра $e^m = \{x_0, \{x_1, \dots, x_k\}\} \in E^m$. Для каждой обычной вершины x_i ($i \in \overline{1, k}$) определим множество достижимости $R_{y_i}^o$, которому она принадлежит: $x_i \in R_{y_i}^o$. Создадим квазивершину $q_{\{y_1, \dots, y_k\}} \in V_{ord}$ (если таковая не была создана раньше) и добавим обычное ребро $\{x_0, q_{\{y_1, \dots, y_k\}}\} \in E_{ord}$.

Следствие 2. Если кратный граф $G(V, E)$ является эйлеровым, то эйлеровым является и граф с квазивершинами $G_{ord}(V_{ord}, E_{ord})$, причем каждому эйлерову маршруту в исходном кратном графе соответствует эйлеров маршрут в графе с квазивершинами.

Обратное, вообще говоря, неверно: граф с квазивершинами может быть эйлеровым, тогда как исходный граф не эйлеров (см. пример 5 ниже); также в графе с квазивершинами может существовать эйлеров цикл или цепь, которым не соответствует эйлеров цикл или цепь в кратном графе.

Отметим, что следствия 1 и 2 останутся верными, если в их условиях мы заменим требование эйлеровости кратного графа $G(V, E)$ на требование выполнения необходимых условий его эйлеровости.

Пример 4. Рассмотрим пример построения графа с квазивершинами согласно алгоритму 1. На рис. 8 слева представлен кратный делимый граф G кратности 2. Заменим кратные вершины и ребра на обычные. Далее определим множества достижимости по обычным ребрам:

$$R_7^o = \{7, 8, 9, 10\}, \quad R_{11}^o = \{11\}, \quad R_{12}^o = \{12, 13\}.$$

Нетрудно заметить, что в кратном графе G есть четыре мультиребра с концами в R_7^o и R_{11}^o , а также два мультиребра с концами в R_{11}^o и R_{12}^o . Поэтому нужно создать две квазивершины — $q_{7,11}$ и $q_{11,12}$. Остается соединить эти квазивершины с обычными вершинами ребрами $\{1, q_{7,11}\}$, $\{2, q_{7,11}\}$, $\{14, q_{7,11}\}$, $\{15, q_{7,11}\}$, $\{5, q_{11,12}\}$ и $\{6, q_{11,12}\}$. В итоге получим граф с квазивершинами G_{ord} , представленный на рис. 8 справа.

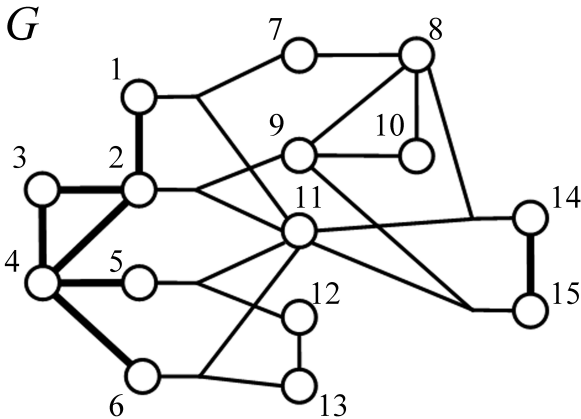


Fig. 8. Multiple graph G and the corresponding ordinary graph G_{ord} with quasi-vertices

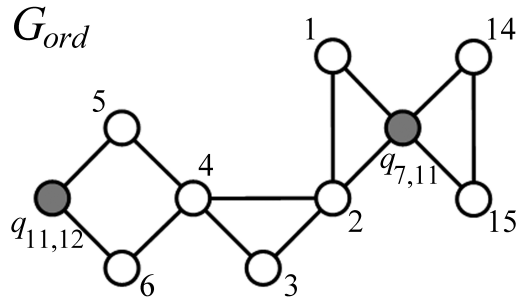


Рис. 8. Кратный граф G и соответствующий ему обычный граф G_{ord} с квазивершинами

Заметим, что все степени обычных вершин графа G четны, а степени кратных вершин кратны 4. Поскольку граф имеет кратность 2, мы видим, что для него выполнены необходимые условия существования эйлерова цикла. Более того, эйлеров цикл C легко найти. $C = C^1 \cup C^2$, где C^1 и C^2 — обычные циклы следующей структуры:

$$C^1 = (1, 7, 8, 14, 15, 9, 10, 8, 9, 2, 3, 4, 5, 12, 13, 6, 4, 2, 1), \quad C^2 = (1, 11, 14, 15, 11, 2, 3, 4, 5, 11, 6, 4, 2, 1).$$

Этому циклу соответствует эйлеров цикл C_{ord} в графе с квазивершинами G_{ord} :

$$C_{ord} = (1, q_{7,11}, 14, 15, q_{7,11}, 2, 3, 4, 5, q_{11,12}, 6, 4, 2, 1).$$

Пример 5. Покажем теперь, что необходимые условия существования эйлерового маршрута в кратном графе не всегда являются достаточными даже в случае делимого графа.

Рассмотрим делимый граф G , представленный в левой части рис. 9. Кратность графа равна 2, степень всех обычных вершин четна, а степень всех кратных вершин равна 4. Следовательно, для этого графа выполняется необходимое условие существования эйлерова цикла.

Нетрудно убедиться, что в каждой из проекций G_1 и G_2 этого графа существует эйлеров цикл (центральная и правая часть рис. 9), однако эти циклы невозможно согласовать на связанных ребрах кратных и мультиребер. Действительно, часть G_1 представляет собой простой цикл $C^1 = (1, 7, 2, 3, 8, 4, 5, 9, 6, 1)$, значит, соответствующий цикл в части G_2 должен быть представлен в форме $C^2 = (1, 10, \dots, 13, 2, 3, 11, \dots, 14, 4, 5, 12, \dots, 15, 6, 1)$, но это невозможно сделать, не пройдя трижды по ребру $\{16, 17\}$, что противоречит определению эйлерова цикла.

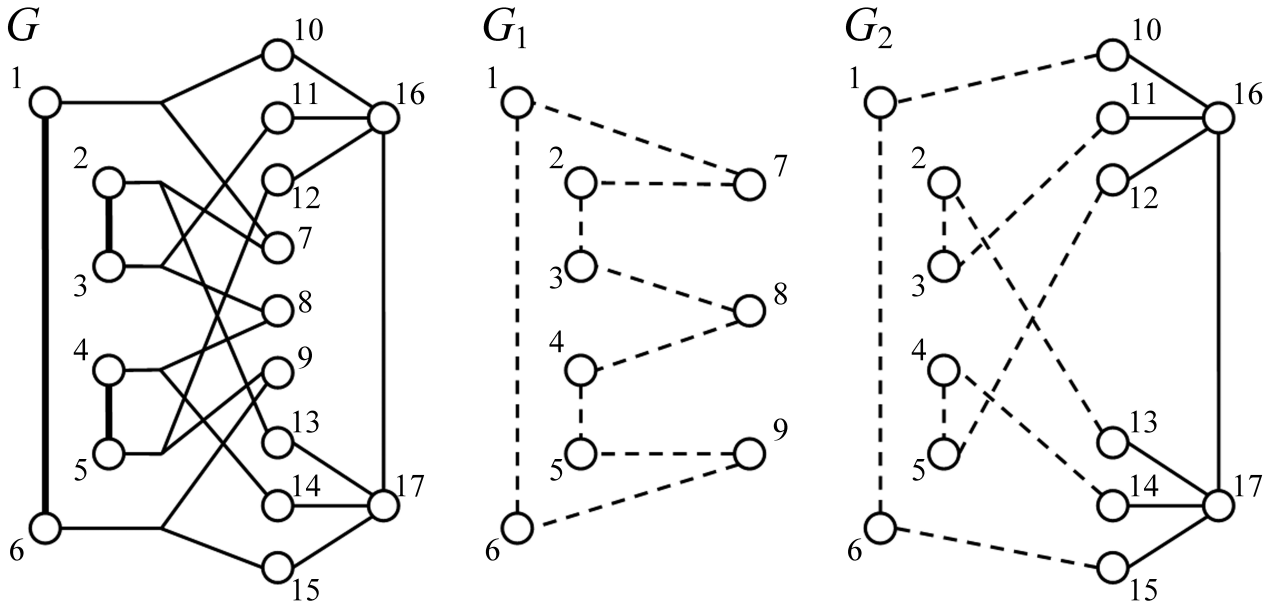


Fig. 9. Non-eulerian divisible multiple graph

Рис. 9. Делимый кратный граф, не являющийся эйлеровым

Отметим, что граф с квазивершинами G_{ord} , построенный для рассматриваемого графа G , также будет представлять собой простой цикл $C_{ord} = (1, q_{7,10}, 2, 3, q_{8,10}, 4, 5, q_{9,10}, 6, 1)$.

Пример 6. Рассмотрим еще одну особенность кратных графов: если граф не является делимым, в нем может одновременно существовать и эйлеров цикл, и эйлерова цепь благодаря тому, что необходимые условия существования эйлерова цикла и цепи не являются взаимоисключающими для таких графов. Действительно, пусть имеется граф G , показанный на рис. 10.

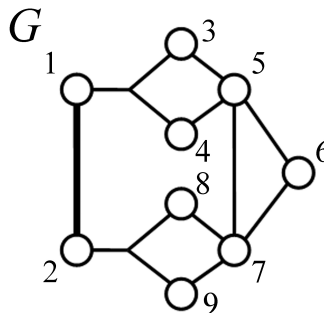


Fig. 10. Non-divisible multiple graph

Рис. 10. Неделимый кратный граф

Этот граф имеет кратность 2 и не является делимым. При этом все обычные вершины имеют четную степень, а степень кратных вершин равна 4. Следовательно, выполнены необходимые условия существования эйлерова цикла. Пример такого цикла C показан на рис. 11 слева.

В то же время, для графа выполнено и необходимое условие существования эйлеровой цепи: $\deg 5 = k + 2$, $\deg 7 = k + 2$ и есть два мультиребра $\{1, \{3, 4\}\}$ и $\{2, \{8, 9\}\}$ таких, что $3, 4 \in R_5^o$ и $8, 9 \in R_7^o$. Следовательно, можно попытаться построить эйлерову цепь $T(5, 7)$. Эта цепь существует (справа на рис. 11).

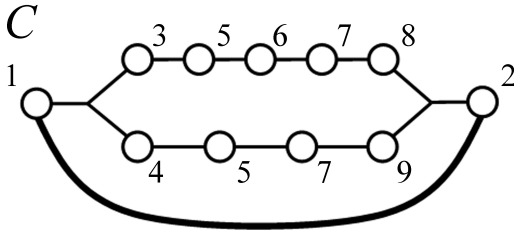


Fig. 11. Eulerian cycle and eulerian trail in the multiple graph G

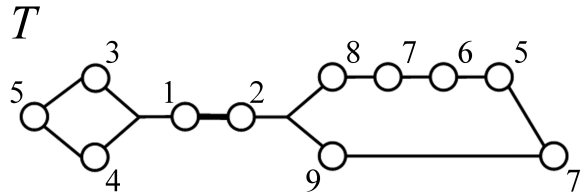


Рис. 11. Эйлеров цикл и эйлерова цепь в кратном графе G

Отметим еще два любопытных факта. Во-первых, обоим эйлеровым маршрутам C и T будет соответствовать один и тот же эйлеров цикл $C_{ord} = (q_{3,3}, 1, 2, q_{3,3})$ в графе с квазивершинами G_{ord} . Во-вторых, вместо вершины 5 или 7 мы могли бы взять в качестве конца эйлеровой цепи вершину 6 ($\deg 6 = k + 0$), и в этом случае построение эйлеровой цепи тоже было бы возможно.

4. Покрывающие цепи в обычном графе

Рассмотренные в предыдущем разделе примеры приводят к следующей идее алгоритма построения эйлерова маршрута в кратном графе G :

1. Находим эйлеров маршрут C_{ord} в графе с квазивершинами G_{ord} .
2. В каждом подграфе G_x^o находим множество из s покрывающих цепей, где число s , а также начальная и конечная вершина каждой цепи определяется по мультиребрам, соответствующим ребрам цикла C_{ord} .

Под *покрывающими цепями* в обычном графе мы понимаем множество цепей, не пересекающихся по ребрам и не содержащих повторяющихся ребер; при этом множество покрывающих цепей содержит все ребра графа.

Известно, что если начала и концы таких цепей не фиксированы, для наличия s покрывающих цепей достаточно, чтобы ровно $2s$ вершин графа имели нечетную степень. Чтобы найти эти цепи, нужно соединить каждую из $2s$ вершин нечетной степени с дополнительной вершиной v^* , а затем найти эйлеров цикл в получившемся графе (см., например, [17]).

В нашем случае может потребоваться начать или закончить сразу несколько из покрывающих цепей в одной и той же вершине. Отмеченный выше результат останется справедливым: для построения покрывающих цепей нам достаточно будет соединить каждую такую вершину сразу несколькими ребрами с дополнительной вершиной v^* и построить эйлеров цикл в получившемся мультиграфе.

Таким образом, если для кратного графа выполнены необходимые условия существования эйлерова маршрута, в каждом подграфе G_x^o существует s покрывающих цепей, где $2s$ — это количество концов мультиребер в R_x^o . Если какая-то вершина является концом нескольких мультиребер, она учитывается несколько раз. Если граф неделимый и для эйлеровой цепи начальная или конечная вершина y находится в R_x^o , эта вершина k раз учитывается в множестве указанных $2s$ вершин.

Однако для построения эйлерова маршрута в кратном графе недостаточно просто существования s покрывающих цепей в G_x^o — эти цепи должны соединять определенные пары вершин, чтобы

обеспечить согласованный порядок прохода связанных ребер каждого мультиребра в эйлеровом маршруте. Указанное обстоятельство приводит к следующей задаче.

Задача 2 (покрывающие цепи с заданными концами в обычном графе). В обычном графе $G_x^o(R_x^o, E_x^o)$ $s \geq 1$ вершин x_1, \dots, x_s отмечены как начальные и s вершин y_1, \dots, y_s отмечены как конечные (множества вершин x_i и y_j могут содержать повторяющиеся вершины и могут пересекаться между собой).

Требуется найти s покрывающих цепей $\mu_1(x_1, y_1), \dots, \mu_s(x_s, y_s)$ с заданными начальными и конечными вершинами.

Отметим, что при $s = 1$ задача 2 сводится к построению эйлеровой цепи в графе G_x^o и разрешима, если для исходного кратного графа выполнены необходимые условия существования эйлерова маршрута. Если же $s > 1$, решения может не быть, даже если условия существования эйлерова маршрута выполнены в исходном кратном графе.

Пример 7. Рассмотрим граф, представленный на рис. 12. В этом графе 10 вершин нечетной степени и 2 — четной, следовательно, в нем можно построить множество из 5 покрывающих цепей.

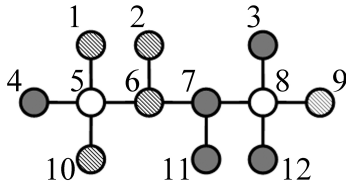


Fig. 12. The problem of finding the covering trails with given endpoints

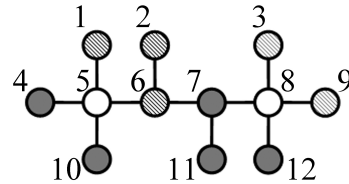


Рис. 12. Проблема получения покрывающих цепей с заданными концами

Однако при фиксации концов этих цепей мы не всегда сможем их получить. Такая ситуация показана слева на рис. 12 (начальные вершины закрашены серым, конечные вершины заштрихованы). При этом не важно, какие именно пары вершин из множеств $\{3, 4, 7, 11, 12\}$ и $\{1, 2, 6, 10, 12\}$ нужно соединить между собой, в любом случае ребро $\{6, 7\}$ придется использовать несколько раз.

Справа на рис. 12 показана ситуация, когда вершина 10 становится начальной для какой-то цепи, а вершина 3 — конечной. В этом случае можно построить, например, такое множество из 5 покрывающих цепей: $\mu_1(4, 1)$, $\mu_2(10, 6)$, $\mu_3(11, 2)$, $\mu_4(7, 9)$, $\mu_5(12, 3)$.

Тем не менее, при другой фиксации концов покрывающие цепи построить не удастся, например: $\mu_1(4, 1)$, $\mu_2(10, 6)$, $\mu_3(11, 3)$, $\mu_4(7, 9)$, $\mu_5(12, 2)$.

Рассмотрим два алгоритма решения задачи 2. Оба они будут в худшем случае выполняться за экспоненциальное число шагов, однако можно предположить, что рассматриваемая задача является NP-трудной.

Идея первого алгоритма состоит в том, что мы соединим каждую из вершин x_i, y_i ($i \in \overline{1, s}$) с дополнительной вершиной v^* ребром (если какая-то вершина встречается в наборе несколько раз, для нее будет создано несколько ребер). В получившемся мультиграфе будем искать эйлеров цикл, в котором ребра инцидентные v^* будут проходиться в заданном порядке и в нужном нам направлении (таком, чтобы при удалении вершины v^* из цикла у нас остались искомые s цепей).

Отметим, что задача построения эйлерова цикла, в котором некоторые ребра проходятся в заданном порядке и в заданном направлении рассматривалась в статьях [18, 19]. В [18] доказано, в частности, что если требуется пройти m ребер и в заданном порядке, и в заданном направлении, задача будет гарантированно иметь решение, если граф $2m$ -связен, и в этом случае решение можно найти с помощью полиномиального алгоритма, описанного в [19]. Однако в нашем графе $m = 2s$ и граф в лучшем случае m -связен, поэтому воспользоваться указанным алгоритмом не удастся (более того, в примере 7 мы рассмотрели ситуацию, когда решения может не быть вовсе).

В нашем алгоритме мы будем использовать описанную в работе [20] операцию *каппа-трансформации* (*κ -трансформации*) эйлерова цикла. Данная операция состоит в изменении на противоположное направления обхода ребер на выбранном участке эйлерова маршрута, заключенном между двумя вхождениями одной вершины (фактически меняется направление обхода какого-то подцикла). В указанной статье доказано, что с помощью последовательного применения операций κ -трансформации можно перебрать все возможные для данного графа эйлеровы маршруты.

Алгоритм 2 (поиск покрывающих цепей с заданными концами перебором эйлеровых циклов).

1. Соединим каждую из вершин x_i, y_i ($i \in \overline{1, s}$) с дополнительной вершиной v^* ребром (если какая-то вершина встречается в наборе несколько раз, для нее будет создано несколько ребер).
2. С помощью полиномиального алгоритма Хиргольцера (см. [15]) найдем эйлеров цикл C в получившемся мультиграфе.
3. Если удалить из цикла C вершины v^* вместе с инцидентными ребрами, у нас останется ровно s покрывающих цепей. Если при этом указанные цепи будут иметь вид: $\mu_1(x_1, y_1), \dots, \mu_s(x_s, y_s)$, эти цепи являются искомыми, выход. Иначе переходим на шаг 4.
4. Рекурсивно перестраиваем цикл C , используя все возможные κ -трансформации. После применения каждой отдельной операции переходим на шаг 3 для проверки. Если вариантов κ -трансформации не осталось, задача 2 для данного графа не имеет решения, выход.

Альтернативный подход к решению задачи 2 состоит в том, что мы будем последовательно находить цепи $\mu_i(x_i, y_i)$, включая в них ребра, не являющиеся *перешейками* (*мостами*) и исключая эти ребра из графа. Похожим образом работает алгоритм Флэри поиска эйлерова маршрута (см. [16]), однако нам нужно будет предусмотреть возможность возврата из-за того, что требуется найти сразу s цепей, которые не должны пересекаться по ребрам.

Понятие перешейка также придется трактовать иначе, чем в классическом алгоритме.

Определение 14. Пусть на текущем шаге выполняется построение цепи $\mu_i(x_i, y_i)$ ($i \in \overline{1, s}$), и уже получена цепь от x_i до v . Ребро $\{v, w\}$ является *перешейком*, если после его удаления будет выполнено одно из условий:

- не существует цепи $\mu(w, y_i)$ (перешеек *первого типа*);
- не существует цепи $\mu(x_j, y_j)$ ($j > i$) (перешеек *второго типа*);
- в графе появляется компонента связности, не содержащая ни x_j , ни y_j ($j > i$), ни y_i (перешеек *третьего типа*).

Отметим, что одно и то же ребро может одновременно быть перешейком и первого, и второго типа, но перешеек третьего типа не может одновременно иметь первый или второй тип. Если какое-то ребро $\{v, w\}$ — перешеек третьего типа, остальные ребра инцидентные v не являются перешейками (v будет частью компоненты связности, «отсекаемой» этим перешейком).

Алгоритм 3 (поиск с возвратами покрывающих цепей с заданными концами).

1. $i = 1$.
2. Если $i < s$, переходим на шаг 3. Иначе $i = s$, ищем эйлерову цепь $\mu_s(x_s, y_s)$ с помощью алгоритма Флэри, выход.
3. $v = x_i$.
4. Если $v = y_i$, полагаем $i = i + 1$ и переходим на шаг 2. Иначе переходим на шаг 5.
5. Если какое-то ребро $\{v, w\}$ является перешейком третьего типа, определяем компоненту связности, «отсекаемую» этим перешейком, и находим в ней эйлеров цикл с помощью алгоритма Флэри. Присоединяем найденный цикл к $\mu_i(x_i, y_i)$ и удаляем его из графа. Включаем ребро $\{v, w\}$

в $\mu_i(x_i, y_i)$ и удаляем его из графа вместе с вершиной v . Полагаем $v = w$ и переходим на шаг 4. Иначе переходим на шаг 6.

6. Поочередно рассматриваем все ребра $\{v, w\}$ (вершины w перебираем в порядке возрастания номеров).

Если ребро $\{v, w\}$ не является перешейком первого или второго типа, включаем его в $\mu_i(x_i, y_i)$ и удаляем из графа (если для вершины v это ребро последнее, она также удаляется из графа). Полагаем $v = w$ и переходим на шаг 4.

Если все ребра инцидентные v являются перешейками, переходим на шаг 7.

7 (возврат). Отменяем действия предыдущих шагов до тех пор, пока не дойдем до шага 6, где для какой-то вершины v осталось нерассмотренным хотя бы одно ребро $\{v, w\}$. После этого возобновляем шаг 6 с рассмотрением указанного ребра. Если в процессе возврата мы дошли до вершины x_1 и для нее не осталось нерассмотренных ребер, задача 2 не имеет решения для данного графа, выход.

Пример 8. Проиллюстрируем работу алгоритмов 2 и 3 на примере графа, представленного на рис. 13. Здесь требуется найти покрывающие цепи с фиксированными концами вида: $\mu_1(1, 6)$, $\mu_2(2, 5)$, $\mu_3(3, 4)$.

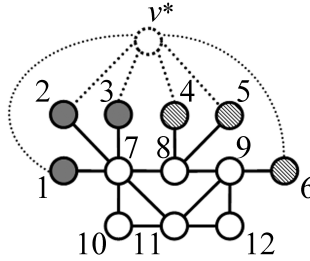


Fig. 13. Search for the covering trails with given endpoints

Рис. 13. Поиск покрывающих цепей с заданными концами

Сначала применим алгоритм 2. Добавим в граф дополнительную вершину v^* и ребра, соединяющие ее с вершинами 1, 2, 3, 4, 5, 6 (показаны на рис. 13 пунктиром). Найдем эйлеров цикл C в графе:

$$C = (v^*, 1, 7, 2, v^*, 3, 7, 8, 4, v^*, 5, 8, 9, 11, 7, 10, 11, 12, 9, 6, v^*).$$

Этот цикл не удовлетворяет условию задачи, поэтому будем рекурсивно выполнять κ -трансформацию цикла (трансформируемые подциклы отмечены подчеркиванием):

$$C = (v^*, 2, 7, 1, v^*, 3, 7, 8, 4, v^*, 5, 8, 9, 11, 7, 10, 11, 12, 9, 6, v^*),$$

$$C = (v^*, 2, 7, 3, v^*, 1, 7, 8, 4, v^*, 5, 8, 9, 11, 7, 10, 11, 12, 9, 6, v^*), \quad \dots,$$

пока не придем к искомому циклу:

$$C = (v^*, 2, 7, 8, 5, v^*, 1, 7, 11, 9, 6, v^*, 3, 7, 10, 11, 12, 9, 8, 4, v^*).$$

Теперь продемонстрируем работу алгоритма 3 для того же графа (но без дополнительной вершины v^*).

Будем строить цепь $\mu_1(1, 6)$. Сначала включим в нее ребро $\{1, 7\}$. На следующем шаге ребра $\{7, 2\}$ и $\{7, 3\}$ будут перешейками одновременно и первого и второго типа, поэтому добавим в цепь следующее по номеру ребро $\{7, 8\}$. Дальше все три ребра оказываются перешейками: $\{8, 9\}$ — второго типа, а $\{8, 4\}$ и $\{8, 5\}$ — одновременно и первого и второго типа. Поэтому выполняем возврат,

исключая из μ_1 ребро $\{7, 8\}$ и добавляя следующее по номеру ребро $\{7, 10\}$. Продолжая действовать подобным образом, мы получим цепь $\mu_1(1, 6) = (1, 7, 10, 11, 9, 6)$, а затем цепи $\mu_2(2, 5) = (2, 7, 8, 5)$ и $\mu_3(3, 4) = (3, 7, 11, 12, 9, 8, 4)$.

В заключение данного раздела отметим, что задачу 2 можно рассмотреть и для подграфов G_x^k кратного графа $G(V, E)$, если кратные ребра заменить на обычные, а в качестве начальных и конечных вершин покрывающих цепей рассматривать кратные вершины, являющиеся началами мультиребер.

5. Алгоритм поиска эйлерова маршрута в кратном графе

Сформулируем теперь алгоритм поиска эйлерова маршрута в делимом кратном графе $G(V, E)$, который будет использовать алгоритмы 1–3 на отдельных своих шагах.

Алгоритм 4 (поиск эйлерова маршрута в делимом кратном графе).

1. Проверяем выполнение необходимых условий существования эйлерова цикла или цепи (теоремы 1–2). Если они не выполнены, эйлерова маршрута не существует, выход.
2. Находим все R_x^o , с помощью алгоритма 1 строим граф с квазивершинами G_{ord} .
3. Находим эйлеров маршрут C_{ord} в графе G_{ord} с помощью алгоритма Хиргольцера.
4. По очередности и направлению прохождения в маршруте C_{ord} ребер инцидентных квазивершинам определяем для каждого G_x^o множества начальных и конечных вершин и их индексы.
5. В каждом из G_x^o с помощью алгоритма 2 или алгоритма 3 ищем покрывающие цепи с заданными концами. Если для всех G_x^o эта задача решена успешно, переходим на шаг 6, иначе — на шаг 7.
6. Находим искомый эйлеров маршрут C в кратном графе $G(V, E)$, перенося в кратный граф маршрут C_{ord} и объединяя его со всеми найденными покрывающими цепями в нужном порядке. Выход.
7. Рекурсивно перестраиваем цикл C_{ord} , используя все возможные κ -трансформации. После применения каждой отдельной операции переходим на шаг 4. Если вариантов κ -трансформации не осталось, задача 1 для данного графа не имеет решения, выход.

Отметим, что алгоритм 4 выполнится в худшем случае за экспоненциальное количество шагов, однако можно предположить, что задача 1 является NP-трудной (из-за необходимости согласования всех частей маршрута на связанных ребрах кратных и мультиребер).

Алгоритм 4 сформулирован для делимых графов. Однако его можно перенести и на случай неделимого графа с некоторыми дополнениями (в общем случае увеличивающими перебор). При этом на шаге 1 мы будем проверять условия теорем 1 и 3.

Так, для квазивершин, у которых совпадают два и более индексов, набор начальных и конечных вершин при поиске покрывающих цепей в G_x^o может определяться неоднозначно. Тогда в худшем случае придется перебирать все возможные варианты сопоставления. К примеру, в эйлеровом маршруте сначала проходит мультиребро $\{x_0, \{x_1, x_2\}\}$ в направлении от общего конца, а потом — мультиребро $\{y_0, \{y_1, y_2\}\}$ в направлении к общему концу, и все вершины $x_1, x_2, y_1, y_2 \in R_x^o$. Тогда при поиске покрывающих цепей надо сначала будет рассмотреть вариант $\mu_1(x_1, y_1), \mu_2(x_2, y_2)$, а если таких цепей найти не удастся, то вариант $\mu_1(x_1, y_2), \mu_2(x_2, y_1)$.

Дополнительная сложность появляется, если для графа выполняется условие 2 или 3 из теоремы 3. В этом случае у нас может возникнуть несколько вариантов выбора начальной или конечной обычной вершины эйлеровой цепи (как в примере 6), и в худшем случае придется просмотреть их все. Кроме того, при построении покрывающих цепей нужно будет k раз включить выбранную обычную вершину в множество начальных или конечных вершин для какого-то G_x^o .

6. Полиномиальный алгоритм для частного случая

Пусть теперь $G(V, E)$ — делимый кратный граф. Пусть в графе с квазивершинами G_{ord} степень всех квазивершин $\deg q_{\{i_1, \dots, i_k\}} = 2$ и никакие два множества индексов квазивершин не пересекаются. По сути, это означает, что каждый подграф G_x^o инцидентен ровно двум связанным ребрам разных мультиребер.

Тогда эйлеров маршрут в этом графе можно найти с помощью следующего алгоритма.

Алгоритм 5 (поиск эйлерова маршрута, частный случай делимого кратного графа).

1. Проверяем выполнение необходимых условий существования эйлерова цикла или цепи (теоремы 1–2). Если они не выполнены, эйлерова маршрута не существует, выход.
2. Находим все R_x^o , с помощью алгоритма 1 строим граф с квазивершинами G_{ord} .
3. Находим эйлеров маршрут C_{ord} в графе G_{ord} с помощью алгоритма Хиргольцера.
4. В каждом G_x^o находим эйлерову цепь с помощью алгоритма Хиргольцера. В силу вышесказанного, эта цепь будет соединять два конца связанных ребер мультиребер (степень всех обычных вершин кратного графа четна; убирая два мультиребра, мы получаем две вершины нечетной степени в соответствующем G_x^o).
5. Находим искомым эйлеров маршрут C в кратном графе $G(V, E)$, перенося в кратный граф маршрут C_{ord} и объединяя его со всеми найденными в подграфах G_x^o эйлеровыми цепями (при необходимости обращая эти цепи).

Теорема 4. В делимом кратном графе с указанными выше ограничениями на квазивершины необходимые условия существования эйлерова маршрута являются достаточными; эйлеров маршрут может быть найден с помощью алгоритма 5 за полиномиальное время.

Доказательство. Достаточность условий и применимость алгоритма 5 следуют из вышеизложенных рассуждений.

Обоснуем полиномиальность алгоритма 5:

- проверка условий на шаге 1 представляет собой простую проверку степеней вершин;
- множества R_x^o на шаге 2 ищутся с помощью полиномиальных алгоритмов из статьи [1]; построение графа с квазивершинами линейно относительно количества вершин и ребер в исходном кратном графе;
- на шаге 3 применяется полиномиальный алгоритм Хиргольцера;
- на шаге 4 этот же алгоритм запускается $|E^m|/2$ раз;
- восстановление искомого маршрута на шаге 5 линейно относительно количества ребер кратного графа.

□

Отметим, что если мы ослабим условие и допустим пересечение множеств индексов квазивершин, то необходимые условия существования эйлерова маршрута в кратном графе перестанут быть достаточными (такому ослабленному условию соответствует граф из примера 5).

References

- [1] A. V. Smirnov, “The shortest path problem for a multiple graph”, *Automatic Control and Computer Sciences*, vol. 52, no. 7, pp. 625–633, 2018.
- [2] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*, 3rd. The MIT Press, McGraw-Hill Book Company, 2009, 1292 pp.
- [3] C. Berge, *Graphs and Hypergraphs*. North-Holland Publishing Company, 1973, 528 pp.

- [4] A. Basu and R. W. Blanning, “Metagraphs in workflow support systems”, *Decision Support Systems*, vol. 25, no. 3, pp. 199–208, 1999.
- [5] A. Basu and R. W. Blanning, *Metagraphs and Their Applications*, ser. Integrated Series in Information Systems. Springer US, 2007, vol. 15, 172 pp.
- [6] V. S. Rublev and A. V. Smirnov, “Flows in multiple networks”, *Yaroslavsky Pedagogichesky Vestnik*, vol. 3, no. 2, pp. 60–68, 2011, in Russian.
- [7] A. V. Smirnov, “The problem of finding the maximum multiple flow in the divisible network and its special cases”, *Automatic Control and Computer Sciences*, vol. 50, no. 7, pp. 527–535, 2016.
- [8] L. R. Ford and D. R. Fulkerson, *Flows in Networks*. Princeton University Press, 1962, 210 pp.
- [9] V. S. Rublev and A. V. Smirnov, “The problem of integer-valued balancing of a three-dimensional matrix and algorithms of its solution”, *Modeling and Analysis of Information Systems*, vol. 17, no. 2, pp. 72–98, 2010, in Russian.
- [10] A. V. Smirnov, “Network model for the problem of integer balancing of a four-dimensional matrix”, *Automatic Control and Computer Sciences*, vol. 51, no. 7, pp. 558–566, 2017.
- [11] A. V. Smirnov, “Spanning tree of a multiple graph”, *Journal of Combinatorial Optimization*, vol. 43, no. 4, pp. 850–869, 2022.
- [12] A. V. Smirnov, “The optimized algorithm of finding the shortest path in a multiple graph”, *Modeling and Analysis of Information Systems*, vol. 30, no. 1, pp. 6–15, 2023, in Russian.
- [13] A. V. Smirnov, “NP-completeness of the minimum spanning tree problem of a multiple graph of multiplicity $k \geq 3$ ”, *Automatic Control and Computer Sciences*, vol. 56, no. 7, pp. 788–799, 2022.
- [14] L. Euler, “Solutio problematis ad geometriam situs pertinentis”, *Commentarii Academiae Petropolitanae*, vol. 8, pp. 128–140, 1741, in Latin.
- [15] C. Hierholzer, “Über die Möglichkeit, einen Linienzug ohne Wiederholung und ohne Unterbrechung zu umfahren”, *Mathematische Annalen*, vol. 6, pp. 30–32, 1873, in German.
- [16] M. Fleury, “Deux problèmes de géométrie de situation”, *Journal de mathématiques élémentaires*, vol. 2, pp. 257–261, 1883, in French.
- [17] F. Harary, *Graph theory*. Addison-Wesley Pub. Co., 1969, 274 pp.
- [18] M.-C. Cai and H. Fleischner, “An eulerian trail traversing specified edges in given order”, *Journal of Graph Theory*, vol. 19, no. 2, pp. 137–144, 1995.
- [19] M.-C. Cai, “An algorithm for an eulerian trail traversing specified edges in given order”, *Discrete Applied Mathematics*, vol. 55, no. 3, pp. 233–239, 1994.
- [20] J. Abraham and A. Kotzig, “Transformations of euler tours”, *Annals of Discrete Mathematics*, vol. 8, pp. 65–69, 1980.