

ОСНОВНЫЕ ПОНЯТИЯ

1 ОСНОВНЫЕ ПОНЯТИЯ И ОПРЕДЕЛЕНИЯ ТЕОРИИ ГРАФОВ

1.1. Начнем данный параграф с определения неориентированных и ориентированных графов.

1.1.1. Формальное и достаточно общее определение неориентированного графа таково.

Определение 1.1. Неориентированным графом G называется тройка

$$G = (V, E, I),$$

состоящая из

(1) (конечного) множества вершин $V = V(G)$, например,

$$V = \{1, 2, 3, 4\},$$

(2) (конечного) множества ребер $E = E(G)$, например,

$$E = \{a, b, c, d, e, f, g, h\},$$

(3) а также отображения $I: E \rightarrow V_2$, сопоставляющего любому ребру $e \in E$ неупорядоченную пару вершин $\{x, y\} \in V_2$, которую это ребро соединяет.

Вершины x и y называются *концевыми вершинами* ребра e . При этом говорят, что ребро e *инцидентно* своим концевым вершинам. Про любую из двух вершин x, y , в свою очередь, говорят, что она инцидентна ребру e .

В принципе, возможен случай $x = y$. Ребро $e \in E$, соответствующее паре $\{x, x\}$, называется обычно *петлей*. Кроме того, в общем случае у нас могут быть несколько различных ребер, соединяющих одну и ту же пару вершин $\{x, y\}$. Такие ребра называют кратными ребрами и говорят, что они образуют *мультиребро* графа G . Если пара вершин $\{x, y\}$ соединена между собой единственным ребром, то такое ребро иногда называют простым ребром графа.

Пример 1.2. Зададим отображение I в виде следующей таблицы:

E	a	b	c	d	e	f	g	h
V_2	$\{1, 3\}$	$\{2, 4\}$	$\{1, 3\}$	$\{3, 4\}$	$\{3, 4\}$	$\{1, 2\}$	$\{3, 4\}$	$\{2, 2\}$

Ей соответствует граф G , изображенный на рис.1. В этом графе ребра f и b являются простыми, кратные ребра a и c образуют мультиребро, соединяющее вершины 1 и 3, а ребро h представляет собой петлю.

1.1.2. С понятием инцидентности тесно связано важное понятие степени вершины графа G .

Определение 1.3. В неориентированном графе G *степенью* $\deg(x)$ или *валентностью* вершины x называется количество ребер, инцидентных x . Считается, что петля дает вклад, равный двум, в степень любой вершины. Вершина, степень которой равна нулю, называется *изолированной*. Если все вершины в графе G имеют одинаковую степень, то граф G называют *регулярным*.

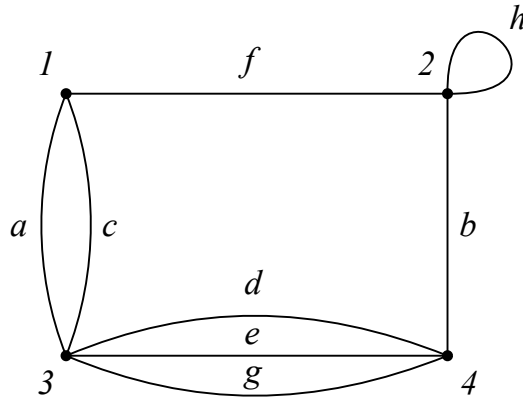


Рис. 1: Пример графа на четырех вершинах

Так, вершина 1 на рис.1 имеет степень, равную трем, а вершина 2 — степень, равную четырем. Следующее утверждение часто называют первой теоремой теории графов.

Теорема 1.4. *В неориентированном графе G сумма степеней всех вершин равна удвоенному количеству всех ребер графа:*

$$\sum_{x \in V(G)} \deg(x) = 2|E(G)|. \quad (1)$$

Доказательство практически очевидно — любое ребро дает вклад, равный двум, в стоящую слева сумму.

Следствие 1.5. *Количество вершин в графе G , имеющих нечетную степень, четно.*

1.1.3. Несмотря на достаточную очевидность доказательства теоремы 1.4, оно использует чрезвычайно важный прием, очень часто встречающийся и в теории графов, и в комбинаторике — так называемый двойной подсчет (double counting).

Неформально его можно себе представлять следующим образом. Предположим, что у нас имеется какое-то множество C элементарных объектов C_i , любой из которых мы можем рассматривать как составную часть двух различных, более сложно устроенных объектов A_j и B_k . Тогда количество $c = |C|$ всех элементарных объектов C_i мы всегда сможем сосчитать двумя способами. Именно, для любого сложного объекта первого типа A_j мы можем сосчитать количество a_j элементарных объектов, из которых состоит A_j , а затем просуммировать полученные числа a_j по j от 1 до n , где n есть количество всех сложных объектов первого типа. С другой стороны, мы можем для любого B_k сосчитать количество b_k элементарных объектов, из которых состоит сложный объект второго типа B_k , а затем просуммировать b_k по всем k от 1 до m , где m — количество объектов второго типа. Так как общее количество c всех элементарных объектов C_i , от способа подсчета не зависит, то мы в итоге получаем равенство вида

$$\sum_{j=1}^n a_j = c = \sum_{k=1}^m b_k, \quad (2)$$

которое и отражает некоторый конечный результат процедуры double counting.

При доказательстве первой теоремы теории графов мы в качестве элементарного объекта C_i можем взять так называемое *полуребро* графа G (semi-edges или darts), то есть половинку любого ребра, инцидентного вершине x (см.рис.2, на котором полуребра изображены синим цветом).

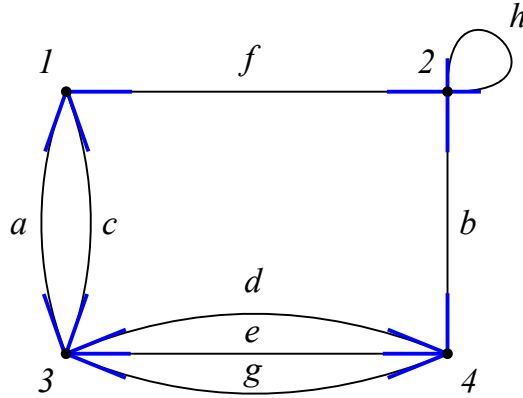


Рис. 2

Каждое ребро состоит ровно из двух полуредер, поэтому количество c всех полуредер равно удвоенному количеству $2m$, $m = |E(G)|$, всех ребер. С другой стороны, любую вершину x_j мы можем представить в виде “ежа”, то есть вершины вместе с инцидентными ей полуредрами. Количество a_j таких полуредер совпадает со степенью $\deg(x_j)$ вершины x_j . Суммируя эти числа a_j по всем $j = 1, \dots, n$, где n — количество всех вершин в графе, мы вновь получаем общее количество полуредер c . Формула (1) при этом есть не что иное, как соотношение (2), переписанное для рассматриваемого частного случая.

1.1.4. Описанную выше процедуру double counting можно формализовать с помощью так называемой матрицы инцидентности M_i . Именно, рассмотрим матрицу M_i размерами $n \times m$, строки которой отвечают объектам A_j , $j = 1, \dots, n$ первого типа, а столбцы — объектам B_k , $k = 1, \dots, m$ второго типа. Назовем объекты A_j и B_k *инцидентными* друг другу, если у нас существует хотя бы один элементарный объект, являющийся частью как A_j , так и B_k . Любой элемент $m_{j,k}$ матрицы M_i равен количеству элементарных объектов, входящих как в A_j , так и в B_k . При таком подходе левая часть равенства (2) подсчитывает количество элементарных объектов суммированием по строкам матрицы M_i , а правая часть (2) — суммированием по столбцам.

В случае неориентированного графа G строки матрицы M_i инцидентности графа G отвечают вершинам этого графа, а столбцы — его ребрам. Элемент $m_{j,k}$ матрицы M_i равен нулю, если вершина x_j и ребро e_k не инцидентны друг другу, $m_{j,k} = 2$ в случае, если e_k есть инцидентная вершине x_j петля, и $m_{j,k} = 1$ в остальных случаях.

Рассмотрим, к примеру, граф G , показанный на рисунках 1 и 2. Для такого графа матрица инцидентности имеет следующий вид:

$$M_i = \left(\begin{array}{c|cccccccc} V \setminus E & a & b & c & d & e & f & g & h \\ \hline 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 2 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 2 \\ 3 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 0 \\ 4 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 \end{array} \right).$$

Видно, что сумма элементов в любом столбце этой матрицы равна двум, а сумма элементов в любой строке совпадает со степенью вершины x этого графа. Суммирование по строкам и суммирование по столбцам снова приводит нас к равенству (1).

1.2. Чаще всего на практике встречаются так называемые простые графы, к описанию которых мы и перейдем.

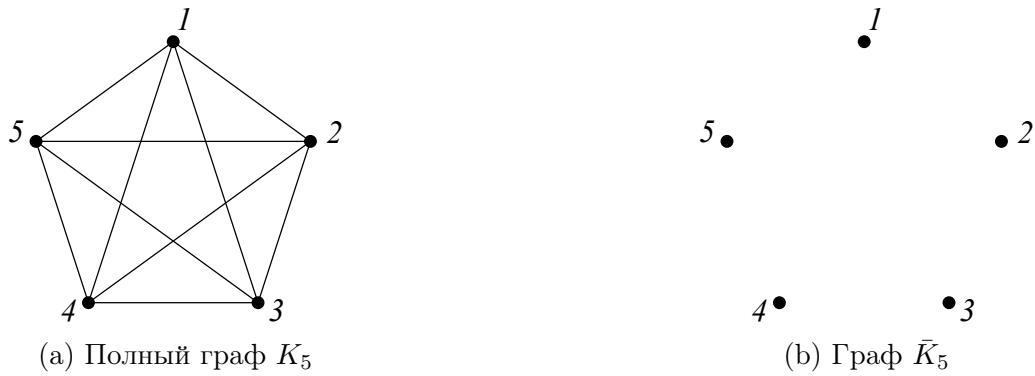


Рис. 4: Полный граф и пустой граф

Определение 1.7. Граф \bar{G} называется *дополнением* к графу G , если множества вершин этих двух графов совпадают, а множество ребер графа \bar{G} дополняет множество ребер $E(G)$ исходного графа G до множества ребер полного графа K_n .

На рис.3 в качестве примера приведены два графа, построенные на четырех вершинах — граф G и граф \bar{G} . Граф \bar{G} из графа G можно получить, например, так: взять полный граф K_4 , построенный на том же количестве вершин, что и графы G и \bar{G} , и удалить из него ребра, принадлежащие графу G . Полученный в результате этой операции граф \bar{G} и будет являться дополнением к графу G в смысле данного выше определения.

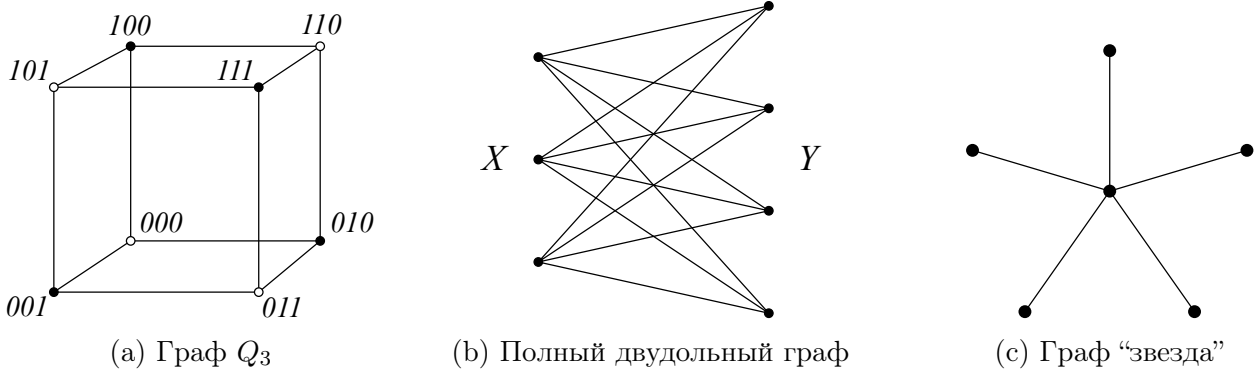


Рис. 5: Двудольные графы

1.2.3. При изучении теории графов нам часто будут встречаться еще несколько важных подклассов простых графов.

Определение 1.8. Граф G называется *двудольным* (рис.5), если множество $V(G)$ его вершин можно разбить на два блока X и Y так, что концы x и y любого ребра $e = \{x, y\} \in E(G)$ лежат в разных блоках этого разбиения. Обозначается двудольный граф с разбиением (X, Y) через $G[X, Y]$.

Простой граф $G[X, Y]$, в котором любая вершина из X соединена ребром с каждой вершиной из блока Y и наоборот, называется *полным двудольным графом* (рис.5,b). Часто такой граф обозначается через $K_{n,m}$, где $n = |X|$, а $m = |Y|$. Граф $K_{n,1}$ называется *звездой* (рис.5,c).

Еще два важных подкласса простого графа, называемые путем P_n и циклом C_n , показаны на рис.6,a и рис.6,b. Количество ребер в графах P_n и C_n характеризует длину таких графов.

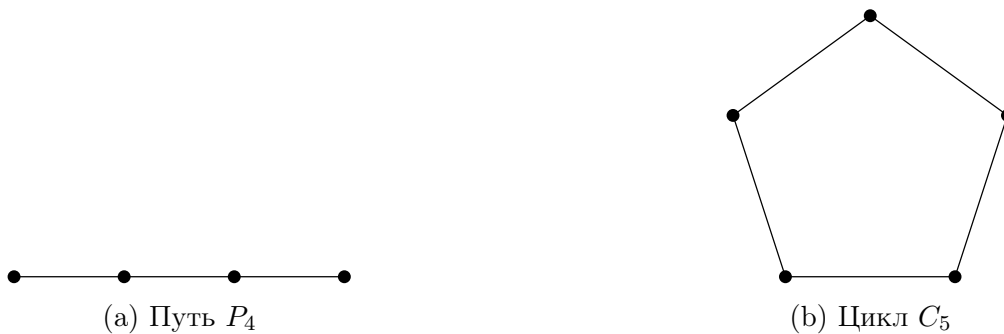


Рис. 6: Путь и цикл

Также часто нам будет встречаться граф W_n , называемый колесом (см.рис.7,а). Индекс n в обозначении W_n такого графа обозначает либо количество вершин в его внешнем цикле, либо количество всех вершин. Мы будем считать, что n — это количество вершин во внешнем цикле графа W_n .



Рис. 7: Колесо и граф Петерсена

Важный класс простых подграфов представляют собой так называемые *регулярные графы*, то есть графы, у которых все вершины имеют одну и ту же степень k . Также их называют k -регулярными графами или регулярными графами степени k . В случае $k = 3$ такие графы называются кубическими. На рис.7,б показан пример кубического графа, который достаточно часто будет встречаться у нас в приложениях — так называемый граф Петерсена (Petersen graph). Графы C_n и K_n также являются регулярными графами степени 2 и $n - 1$ соответственно.

Наконец, имеется важный класс графов, являющихся одновременно k -регулярными и двудольными — так называемые k -кубы Q_k . Вершины графа Q_k можно пометить бинарными строками длины k (см.рис.5,с). Ребра же в таком графе проводятся только между теми вершинами, бинарные последовательности которых отличаются только в одной из k позиций. Пример графа Q_3 показан на рис.5,с.

1.3. Наряду с неориентированными, в теории графов также изучаются и так называемые ориентированные графы (или орграфы).

1.3.1. Начнем с мы вновь с достаточно формального и общего определения ориентированного мультиграфа D .

Определение 1.9. Если в тройке

$$D = (V, E, I),$$

в которой V есть множество вершин, а E — множество ребер, отображение I ставит в соответствие любому ребру e упорядоченную пару вершин $(x, y) \in V \times V$, то такая тройка называется *ориентированным графом* (или *орграфом*). В таком случае говорят, что ребро e *выходит* из вершины x и *входит* в вершину y . На рисунке такое ребро помечается стрелкой, указывающей направление данного ребра.

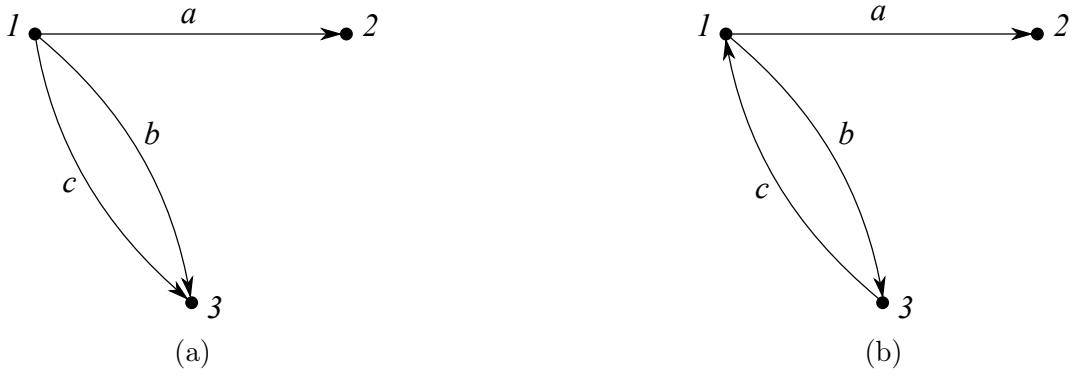


Рис. 8: Примеры ориентированных графов

В качестве примера на рис.8,а показан ориентированный граф D , построенный на множестве вершин $V = \{1, 2, 3\}$ и имеющий три ребра. Ребро a выходит из вершины 1 и входит в вершину 2 (то есть отвечает упорядоченной паре $(1, 2)$), а ребра b и c исходят из вершины 2 и приходят в вершину 3 (то есть им соответствует упорядоченная пара $(2, 3)$ вершин графа D). Таким образом, орграф D на рис. задается тройкой (V, E, I) , в которой множество V вершин и множество E ребер имеют вид

$$V = \{1, 2, 3\}, \quad E = \{a, b, c\},$$

а отображение $I: E \rightarrow V \times V$ задается таблицей вида

E	a	b	c
$V \times V$	$(1, 2)$	$(2, 3)$	$(2, 3)$

1.3.2. В орграфе различают исходящую ($\text{outdeg}(x)$) и входящую ($\text{indeg}(x)$) степень любой вершины $x \in V(D)$, а также просто степень, равную сумме входящей и исходящей степеней. Так, в графе, изображенном на рис.8,а, вершина 1 имеет входящую степень, равную единице, и исходящую степень, равную двойке.

Так как любое ориентированное ребро в орграфе D вносит вклад, равный единице, в сумму всех исходящих степеней вершин орграфа D , а также вклад, равный единице, в сумму всех входящих степеней вершин орграфа G , то для орграфа справедливо равенство вида

$$\sum_{x \in V(D)} \text{indeg}(x) = |E(D)| = \sum_{x \in V(D)} \text{outdeg}(x),$$

являющееся аналогом первой теоремы теории графов (1.4).

1.3.3. Как и для неориентированного графа, важным частным случаем орграфа является простой орграф.

Определение 1.10. Орграф D называется *простым*, если он не содержит петель, а также кратных *упорядоченных* ребер, то есть ребер, отвечающих одинаковым упорядоченным парам вершин.

Упорядоченность вершин в этом определении важна. Так, граф из рис.8,а простым не является — в нем ребра b и c отвечают одной и той же упорядоченной паре $(2, 3)$ вершин. Изображенный же на рис.8,б граф D' является простым. Несмотря на то, что в этом графе по-прежнему имеются два ребра, соединяющих вершины 2 и 3, направлены эти ребра в разные стороны. Иными словами, эти ребра отвечают различным упорядоченным парам вершин — ребро b отвечает упорядоченной паре $(2, 3)$, а ребро c — упорядоченной паре $(3, 2)$.

1.4. Следующим важным понятием в теории графов является понятие смежности вершин.

1.4.1. Начнем с определения смежных вершин в неориентированном графе G .

Определение 1.11. Говорят, что в неориентированном графе G вершина y смежна с вершиной x , если в этом графе существует ребро $\{x, y\}$.

На множестве вершин V смежность задает некоторое отношение. Для неориентированного графа G это отношение является симметричным: если вершина x смежна с вершиной y , то и вершина y смежна с вершиной x .

1.4.2. Для ориентированного графа ситуация несколько сложнее.

Определение 1.12. Говорят, что в ориентированном графе D вершина y смежна с вершиной x , если в этом графе существует ребро (x, y) , исходящее из вершины x и входящее в вершину y . (см.рис.8,а).

Для изображенного на рис.8,а орграфа G вершина 2 смежна с вершиной 1.

Сразу заметим, что если существует ребро (x, y) и не существует ребро (y, x) , то вершина y смежна с вершиной x , а вот x вершиной, смежной с y , уже не является. Так, для изображенного на рис.8,а орграфа вершина 1 смежной с вершиной 2 не является. Вершина x будет смежной с y только в случае, когда в орграфе D существует ребро (y, x) (см. вершины 1 и 2 на рис.8,б). Иными словами, в ориентированном графе D отношение смежности на множестве V вершин симметричным не является.

1.5. Для хранения графа в памяти компьютера, как правило, используются две структуры, тесно связанные с понятием смежности — матрица смежности и список смежности.

1.5.1. Матрица смежности — это матрица M_a размерами $n \times n$, любой элемент a_{ij} которой описывает количество ребер, идущих из вершины i в вершину j . Так, для примера 1.2 соответствующая графу G матрица смежности имеет следующий вид:

$$M_a = \begin{bmatrix} 0 & 1 & 2 & 0 \\ 1 & 1 & 0 & 1 \\ 2 & 0 & 0 & 3 \\ 0 & 1 & 3 & 0 \end{bmatrix}$$

Для ориентированного графа D , показанного на рис.8,а, матрица смежности равна

$$M_a = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \end{bmatrix}$$

Заметим, что для неориентированного графа матрица смежности всегда симметрична. Как следствие, все собственные значения этой матрицы являются вещественными числами.

В случае простого графа или орграфа все диагональные элементы $a_{ii} = 0$, так что сумма собственных значений такой матрицы, совпадающая со следом матрицы M_a , также равняется нулю. Элементы, не лежащие на диагонали, равны единице в случае, если существует ребро, идущее из вершины i в вершину j , и нулю в случае, если такового ребра не существует.

1.5.2. Список смежности — это линейный массив L_a размера n , каждый элемент a_i которого содержит список (мультимножество) вершин, смежных с вершиной i . Для примера 1.2 соответствующий список имеет следующий вид:

1 : смежна с вершинами 2, 3, 3;
 2 : смежна с вершинами 1, 2, 4;
 3 : смежна с вершинами 1, 1, 4, 4, 4;
 4 : смежна с вершинами 2, 3, 3, 3.

Для ориентированного графа, показанного на рис.8,а, список смежности записывается так:

1 : список смежности пуст;
 2 : смежна с вершиной 1;
 3 : смежна с вершинами 2, 2.

2 Маршруты, пути, циклы в графе. Связные графы и орграфы

2.1. Следующая важная группа понятий теории графов связана с обходом графа вдоль некоторой последовательности его вершин и ребер. В этой связи нам понадобятся такие понятия, как маршруты, пути, циклы в графе, а также некоторые связанные с ними характеристики графа.

2.1.1. Начнем с определения маршрута в мультиграфе G .

Определение 2.1. *Маршрутом* (walk) в графе G из вершины x_0 в вершину x_k называется чередующаяся последовательность

$$W := x_0, e_1, x_1, e_2, x_2, \dots, x_{k-1}, e_k, x_k$$

вершин $x_i \in V$ и ребер $e_i \in E$, соединяющих вершины x_{i-1} и x_i . И вершины, и ребра в такой последовательности могут повторяться. Количество ребер в маршруте W называется *длиной* k этого маршрута.

В случае простого графа любой маршрут W полностью определяется последовательностью

$$x_0, x_1, x_2, \dots, x_{k-1}, x_k,$$

вершин $x_i \in V(G)$, любые два последовательных элемента x_{i-1}, x_i которой являются смежными вершинами (т.е. соединены между собой ребром $e_i = \{x_{i-1}, x_i\} \in E(G)$).

Вершины x_0 и x_k называют часто начальной и конечной вершинами маршрута W , а остальные вершины — внутренними его вершинами. Говорят также, что вершины x_0 и x_k связаны маршрутом W , а сам маршрут называют x_0x_k -маршрутом.

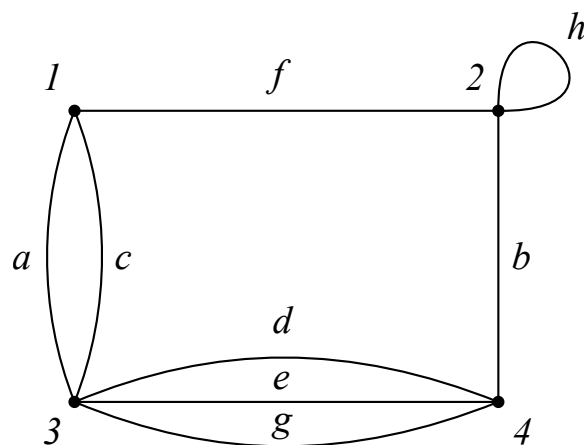


Рис. 9

Определение 2.2. Если все ребра e_1, \dots, e_k в маршруте различны, то такой маршрут называется *путем* (в английской литературе — trail) из вершины x_0 в вершину x_k . Если также и все вершины в данном пути различны, то такой путь называется *простым* (в английской литературе — path).

В качестве примера рассмотрим разные маршруты в графе, показанном на рис.9. Маршрут

$$1, a, 3, d, 4, e, 3, d, 4, b, 2, h, 2$$

путем не является — в нем повторяется ребро d . Маршрут

$$1, a, 3, d, 4, e, 3, g, 4, b, 2, h, 2$$

представляет собой путь (trail), который не является простым — в нем повторяются вершины 3, 4, 2. Наконец, маршрут

$$1, a, 3, e, 4, b, 2$$

является простым путем (path) из вершины 1 в вершину 2.

2.1.2. Следующая серия определений связана с замкнутыми маршрутами в графах, то есть маршрутами, в которых начальная x_0 и конечная x_k вершины совпадают.

Определение 2.3. Замкнутым путем (closed trail) или составным циклом (circuit) в графе G называется путь, в котором $x_0 = x_k$. Замкнутый простой путь называется простым циклом (cycle). Часто слово “простой” опускают, понимая под циклом замкнутый простой путь.

Так, в графе G , показанном на рис.9, путь $1, a, 3, e, 4, d, 3, c, 1$ является составным циклом, а путь вида $1, a, 3, e, 4, b, 2, f, 1$ — простым циклом. Замкнутый маршрут вида $1, a, 3, e, 4, e, 3, c, 1$ составным циклом не является — в нем повторяется ребро e .

В простом графе невозможны циклы длины меньшей, чем три. Циклы длины три часто называются треугольниками. Графы, в которых такие циклы отсутствуют, называются графами, свободными от треугольников (triangle-free graphs).

Определение 2.4. Обхватом графа (girth) называется длина наименьшего цикла в нем. Если в графе циклы отсутствуют, то обхват такого графа считают равным бесконечности.

С учетом данного определения простые графы, свободные от треугольников — это графы, обхват которых больше трех.

В случае мультиграфов возможны и циклы длины 2, и циклы длины 1. Циклы длины 2 — это циклы типа $3, d, 4, e, 3$ на рис.9, появляющиеся при обходе двух ребер одного и того же мультиребра. Циклы длины 1 — это петли графа G . Так, петлю h на рис.9 можно рассматривать как цикл $2, h, 2$ длины 1.

Изолированная вершина по определению является замкнутым путем длины 0, однако циклом она не является.

2.1.3. Теперь мы готовы определить понятие связных графов.

Определение 2.5. Если вершины $x, y \in V$ графа G соединены хотя бы одним путем, то такие вершины называются *связанными*.

Несложно проверить, что связность задает на множестве V вершин графа G отношение эквивалентности. Это отношение делит граф на классы эквивалентности, называемые *компонентами связности графа*.

Определение 2.6. В случае, когда в графе G существует лишь одна компонента связности, то есть в случае, когда любые две вершины x, y графа соединены хотя бы одним путем, граф называется *связным*. В противном случае граф называется *несвязным*.

2.1.4. С понятием пути в графе тесно связаны и такие важные понятия, как расстояние между вершинами в графе, эксцентриситет вершины, диаметр и радиус графа.

Определение 2.7. *Расстоянием* $d(x, y)$ между двумя связанными вершинами $x, y \in V(G)$ называется длина наименьшего пути между ними.

Понятно, что такой путь обязательно является простым. В случае, когда вершины не являются связанными, полагают по определению, что $d(x, y) = \infty$.

Определение 2.8. *Диаметром* графа называется максимальное расстояние между его вершинами:

$$\text{diam}(G) := \max_{x, y \in V(G)} d(x, y).$$

В случае несвязного графа считается, что $\text{diam}(G) = \infty$.

Определение 2.9. *Эксцентриситетом* $\varepsilon(x)$ вершины $x \in V(G)$ называется максимальное расстояние от x до любой другой вершины графа G :

$$\varepsilon(x) := \max_{y \in V(G)} d(x, y).$$

Определение 2.10. *Радиусом* $r(G)$ графа G называется минимальный из эксцентриситетов вершин графа G . Вершины, на которых этот минимум достигается, называются *центральными вершинами* графа G . Множество всех центральных вершин называется *центром* графа.

2.2. Рассмотрим теперь понятие связности в ориентированном графе D .

2.2.1. Начнем с определения связанных вершин в орграфе.

Определение 2.11. Вершины x и y орграфа D называются *связанными*, если в D существуют хотя бы один путь из x в y и хотя бы один путь из y в x .

Определение 2.12. Орграф D называется *сильно связным*, если любые две его вершины являются связанными.

Иногда наряду с этим понятием для орграфа вводят понятие слабой связности.

Определение 2.13. Орграф D называется *слабо связным*, если соответствующий ему неориентированный граф G , получающийся заменой всех ориентированных ребер на неориентированные, является связным.

2.2.2. Как и в случае неориентированного графа, в орграфе D отношение связности является отношением эквивалентности. Как следствие, множество всех вершин $V(D)$ орграфа D разбивается с помощью этого отношения эквивалентности на классы попарно связанных вершин, которые называются *компонентами сильной связности*.

Сразу заметим, что в неориентированном графе ребер между различными компонентами связности не существует. В орграфе такие ребра могут существовать, однако направлены все они будут лишь от одной компоненты связности к другой. Именно, справедливо следующее достаточно очевидное утверждение.

Лемма 2.14. Пусть H_1, H_2 есть две различные компоненты сильной связности графа D , и пусть существует ребро $e \in E(D)$ из H_1 в H_2 . Тогда ребра из H_2 в H_1 отсутствуют.

Доказательство. Действительно, если ребро из H_2 в H_1 существует, то любые две вершины в множестве $H_1 \cup H_2$ вершин оказываются связанными. Иными словами, $H_1 \cup H_2$ представляет собой компоненту сильной связности графа D , что противоречит предположению о том, что H_1 и H_2 есть две различные компоненты сильной связности. \square

Как следствие, по любому орграфу D можно построить так называемый граф $C(D)$ компонент сильной связности графа D , вершинами которого будут компоненты сильной связности графа D , а ребрами — ребра графа D , направленные из одной компоненты сильной связности D к другой. Основное свойство такого орграфа $C(D)$ состоит в том, что в таком графе нет циклов.

Теорема 2.15. В орграфе $C(D)$ циклы отсутствуют, то есть он, как еще говорят, представляет собой *ациклический орграф* (*DAG* — *directed acyclic graph*).

Доказательство. Если бы в таком графе существовал цикл вида $H_1 \rightarrow H_2 \rightarrow \dots \rightarrow H_n \rightarrow H_1$, то любые две вершины в объединении $H_1 \cup H_2 \cup \dots \cup H_k$ оказались бы связанными. Действительно, внутри каждой компоненты H_i мы, по определению сильной связности, можем попасть из любой вершины в любую вершину H_i . Вершины же из разных компонент H_i и H_j мы также можем всегда связать с помощью пути, идущего из H_i в H_j , а также пути, соединяющего компоненты H_j и H_i . \square

3 Подграф графа G . Основные операции над графами

3.1. Вернемся к основным определениям теории графов и введем очень важное понятие подграфа графа G . Заодно мы введем две основные операции над графами — операцию удаления ребра и операцию удаления вершины в графе G .

3.1.1. Начнем с формального определения подграфа H графа G .

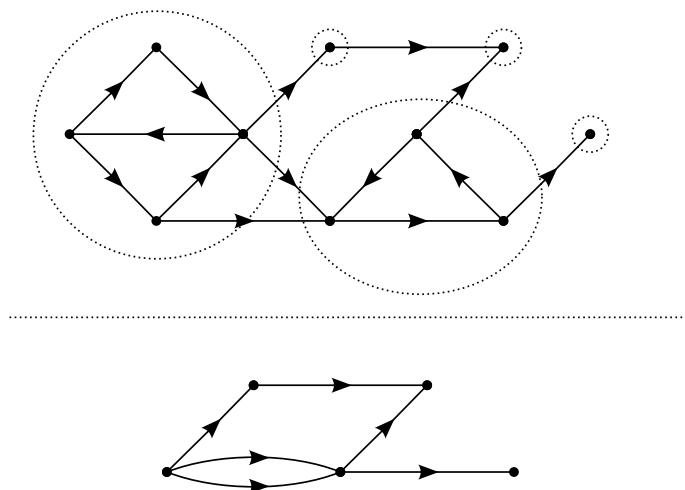


Рис. 10: Орграф D и граф компонент сильной связности графа D

Определение 3.1. Подграфом графа G называется граф H , для которого выполнены следующие три условия:

1. $V(H) \subseteq V(G)$;
2. $E(H) \subseteq E(G)$;
3. любое ребро $e \in E(H)$, соединяющее пару вершин x и y в H , должно соединять ту же самую пару вершин в графе G .

По отношению к графу H граф G иногда называют *надграфом* или *суперграфом*.



Рис. 11

В качестве примера рассмотрим графы G и H , изображенные на рис.11,а,б. Утверждается, что граф H является подграфом графа G . Действительно, множество вершин графа H есть подмножество вершин графа G , а множество ребер $E(H)$ — подмножество множества $E(G)$. Рассмотрим теперь любое ребро графа H , например, ребро $\{a, e\}$ графа H . Это же ребро соединяет те же самые вершины a и e и в графе G . Так как это верно для всех ребер $e \in E(H)$, то граф H действительно является подграфом графа G .

3.1.2. Данное выше определение можно сделать несколько более конструктивным, если ввести две основные операции над графами — операцию удаления ребра и операцию удаления вершины. Оказывается, любой подграф H графа G — это граф, полученный из исходного графа G с помощью этих двух операций.

Начнем с более простой операции удаления ребра e . При такой операции множество вершин графа G не меняется, а из множества ребер удаляется элемент $e \in E(G)$. Полученный в результате этой операции граф обозначается $G - e$. Очевидно, что он является подграфом графа G . Так, в приведенном на рис.11 примере мы в графе G удалили ребро, соединяющее вершины a и b .

В более общем случае мы таким образом можем удалить сразу несколько ребер, принадлежащих некоторому подмножеству S множества $E(G)$ ребер графа G . Полученный в результате этих операций подграф обозначается $G - S$.

Перейдем теперь к чуть более сложной операции удаления вершины. Предположим, что мы хотим удалить в графе G вершину x . Если эта вершина является изолированной, то нам ничего больше делать не нужно. Если же этой вершине инцидентны какие-то ребра, то мы обязаны будем также вместе с вершиной x удалить и их. Действительно, ранее эти ребра вели в вершину x . Вести в никуда ребра не могут, так что нам вместе с вершиной x приходится удалять и все ребра, инцидентные данной вершине. Так, в примере мы удалили в графе G вершину c . Вместе с ней мы вынуждены были удалить и все три инцидентных ей ребра.

Полученный в результате удаления вершины x граф обозначается обычно $G - x$. Очевидно, что граф $G - x$ также является подграфом исходного графа G .

В более общем случае $S \in V(G)$ мы с помощью операции удаления вершин из подмножества S получаем подграф $G - S$, в котором по сравнению с исходным графом G удалены все вершины подмножества S вместе со всеми ребрами, инцидентными этим вершинам.

3.1.3. Как мы уже заметили, любой подграф графа G получается из G последовательным выполнением двух операций — удаления вершин и удаления ребер. Естественным кажется рассмотреть два частных случая этой ситуации. Первый — это случай, когда мы в графе G удаляем только ребра, а второй — когда в G мы удаляем только вершины.

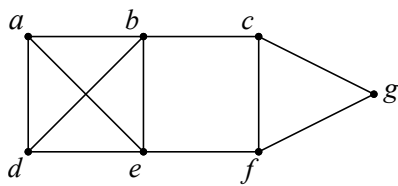


Рис. 12

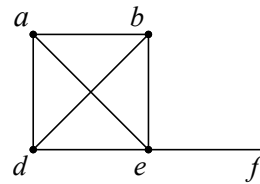
Если мы при получении из графа G подграфа H используем лишь операцию удаления ребер, то мы получаем подграф, множество вершин которого совпадает с множеством $V(G)$ вершин исходного графа. Такой подграф называется *остовным* подграфом (spanning subgraph) графа G (смотри рис.12). В частности, любой простой граф G , построенный на n вершинах, является остовным подграфом полного графа K_n .

Второй частный случай — когда мы в графе G удаляем одну или несколько вершин. В результате такой операции мы получаем подграф H , *индуцированным подмножеством оставшихся вершин графа G* . Иными словами, подграфом H графа G , индуцированным подмножеством вершин S , называется граф, полученный из G удалением всех вершин, не принадлежащих множеству S , вместе со всеми инцидентными этим вершинам ребрами (смотри рис.13).

3.1.4. Достаточно часто в приложениях встречаются важные с практической точки зрения под-



(a) Граф G



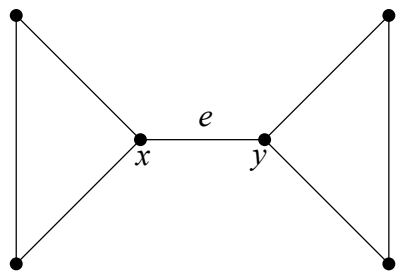
(b) Подграф H графа G , индуцированный подмножеством вершин a, b, d, e и f

Рис. 13

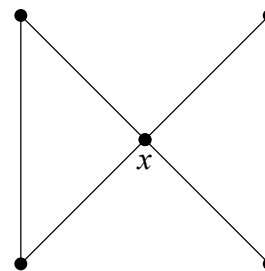
классы остовных подграфов. По историческим причинам некоторые из них имеют специальные названия. Так, остовный 1-регулярный подграф носит название *совершенного паросочетания* в графе G , а остовный k -регулярный подграф для произвольного натурального $k \geq 1$ называется k -*фактором* графа G . Остовный путь в графе носит название гамильтонова пути в графе G , а остовный цикл — гамильтонова цикла в графе G .

В случае мультиграфа G очень часто рассматривают простой граф, получающийся из исходного графа G удалением всех петель и заменой мультиребер на простые ребра. Такой граф, очевидно, является остовным подграфом исходного графа.

3.2. Вернемся к понятию связности в графах. Операции удаления ребра и вершины позволяют нам ввести важные понятия моста и точки сочленения в графе.



(a) Граф G_1



(b) Граф G_2

Рис. 14

3.2.1. Рассмотрим графы, показанные на рис.14. Видно, что связный граф G_1 (рис.14,а) перестает быть таковым после удаления ребра e , связывающего вершины x и y .

Определение 3.2. Ребро $e \in E(G)$ в связном графе G называется *мостом*, если получающийся после его удаления граф $G - e$ становится несвязным. В случае несвязного графа G мостом называется ребро, после удаления которого количество компонент связности увеличивается на единицу.

Теперь рассмотрим связный граф G , показанный на рис.14,б. У этого графа мостов нет, однако имеется вершина x , после удаления которой граф $G - x$ становится несвязным.

Определение 3.3. Вершина x называется *точкой сочленения* графа G , если после ее удаления количество компонент связности графа $G - x$ увеличивается по сравнению с количеством компонент связности исходного графа G .

Заметим, что даже в случае связного графа G удаление в нем точки сочленения может приводить к появлению как двух, так и большего количества связных компонент.

3.2.2. Докажем простое утверждение, характеризующее точку сочленения в связном графе G .

Утверждение 3.4. Вершина x в связном графе G , построенном на $n \geq 3$ вершинах, есть точка сочленения графа G тогда и только тогда, когда в G существуют отличные от x вершины y и z , такие, что x содержится в каждом пути из y в z .

Доказательство. Предположим вначале, что x есть точка сочленения в связном графе G . После удаления x в графе $G - x$ найдутся по крайней мере две различные связные компоненты G_1 и G_2 . Выбрав произвольную вершину y в G_1 и произвольную вершину z в G_2 , мы получим искомую пару вершин. Действительно, если бы в G нашелся путь, не проходящий через вершину x , то в графе $G - x$ вершины y и z оказались бы связанными, что невозможно.

Теперь предположим, что в G существуют описанная в утверждении пара вершин y и z . Удаление вершины x в таком графе уничтожает любой путь из y в z , то есть превращает эти вершины в несвязанные. Это, в свою очередь, означает, что вершины y и z , лежащие ранее в одной компоненте связности графа G , стали принадлежать разным компонентам связности графа $G - x$. Иными словами, удаление x увеличивает количество связных компонент графа, то есть x есть точка сочленения в G . \square

3.2.3. Теперь докажем наиболее важное утверждение, характеризующее мосты в графе G .

Утверждение 3.5. Ребро $e = \{x, y\}$ в простом связном графе G является мостом тогда и только тогда, когда e не принадлежит ни одному из циклов графа G .

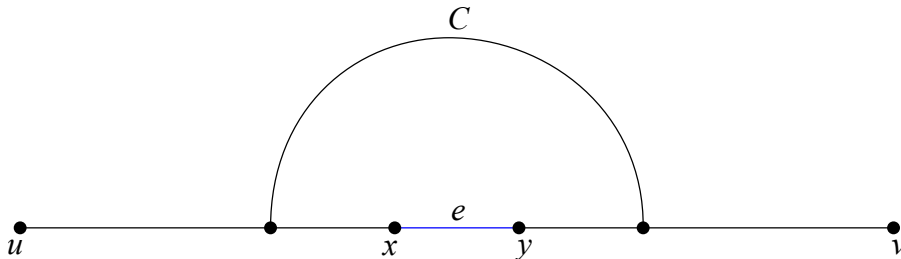


Рис. 15

Доказательство. Пусть ребро e является мостом в графе G . Удаление e приводит к появлению вместо G двух связных компонент G_1 и G_2 , таких, что $x \in G_1$ и $y \in G_2$. Выберем произвольную пару вершин $u \in V(G_1)$ и $v \in V(G_2)$ в этих компонентах (см.рис.15). Предположим, что ребро e принадлежит какому-то циклу C в графе G . Как и любые вершины в связном графе, вершины u и x в G_1 связаны между собой некоторым ux -путем. Аналогично, вершины y и v связаны в G_2 yv -путем. После удаления ребра e вершины x и y остаются связанными между собой некоторым xy -путем, проходящим через оставшиеся ребра цикла C . В силу транзитивности отношения связности отсюда следует, что произвольные вершины $u \in G_1$ и $v \in G_2$ соединены между собой некоторым путем, что противоречит определению моста в G . Следовательно, e никакому циклу принадлежать не может.

Обратно, предположим, что e не является мостом в графе G , то есть его удаление оставляет граф $G - e$ связным. Но тогда концы x и y ребра e в графе $G - e$ оказываются связанными некоторым xy -путем. Добавление к этому пути ребра e превращает этот путь в цикл. \square

Замечание 3.6. Доказанные выше утверждения являются, по сути, аналогами вершинной и реберной теорем Менгера, описывающих k -связные графы. Изучению таких графов будет посвящена отдельная глава.

3.3. Мы ввели операции удаления ребра и вершины. Нам понадобятся еще несколько операций над графами.

3.3.1. Начнем с операций объединения и пересечения графов. Пусть G и H есть пара простых графов.

Определение 3.7. Объединением графов G и H называется граф $F = G \cup H$, множество вершин и множество ребер которого представляют собой объединение соответствующих множеств графов G и H :

$$V(F) = V(G) \cup V(H), \quad E(F) = E(G) \cup E(H).$$

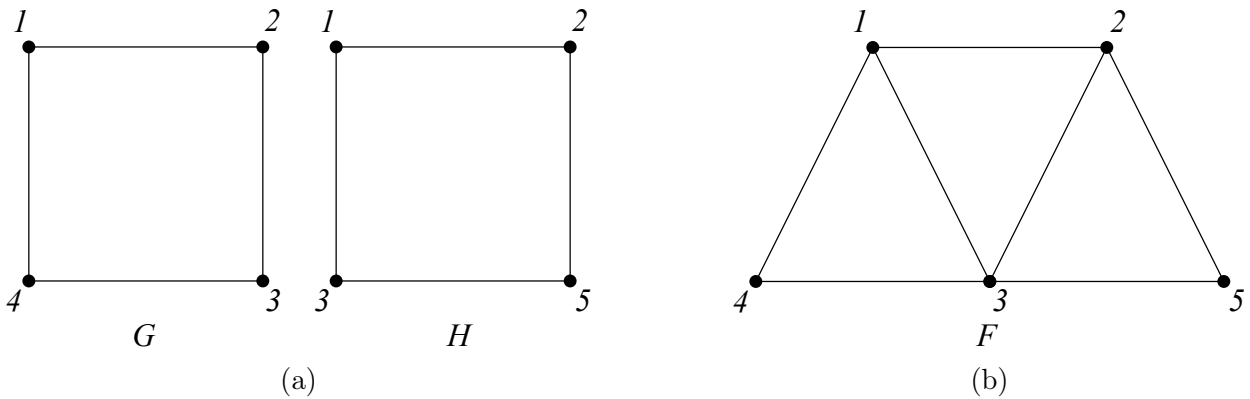


Рис. 16: Операция объединения графов G и H

На рис.16,b в качестве примера показан граф F , полученный в результате объединения графов G и H , показанных на рис.16,a.

В случае, если множества вершин (а следовательно, и ребер) графов G и H не пересекаются, то мы получаем так называемое несвязное объединение $G + H$ таких графов. По сути, любой несвязный граф представляет собой несвязное объединение двух или более связных компонент.

Определение 3.8. Пересечением графов G и H называется граф $F = G \cap H$, множество вершин и множество ребер которого представляют собой пересечение соответствующих множеств графов G и H :

$$V(F) = V(G) \cap V(H), \quad E(F) = E(G) \cap E(H).$$

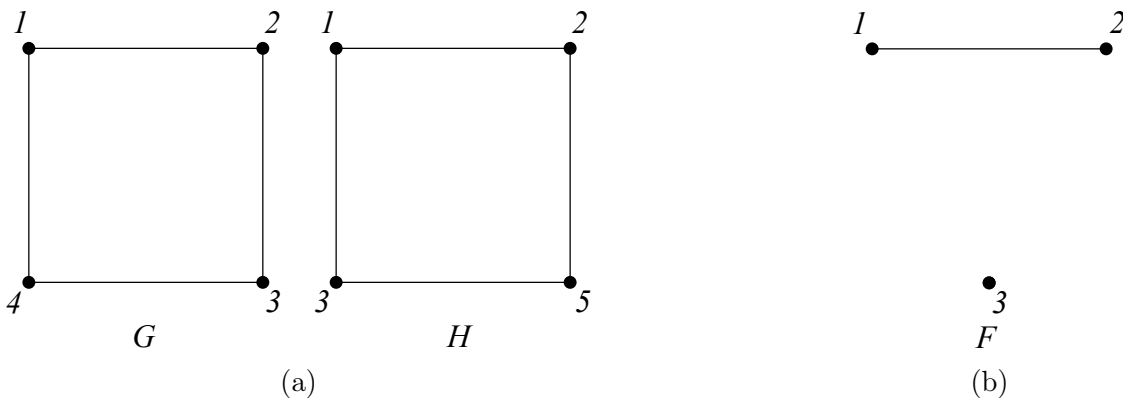


Рис. 17: Операция объединения графов G и H

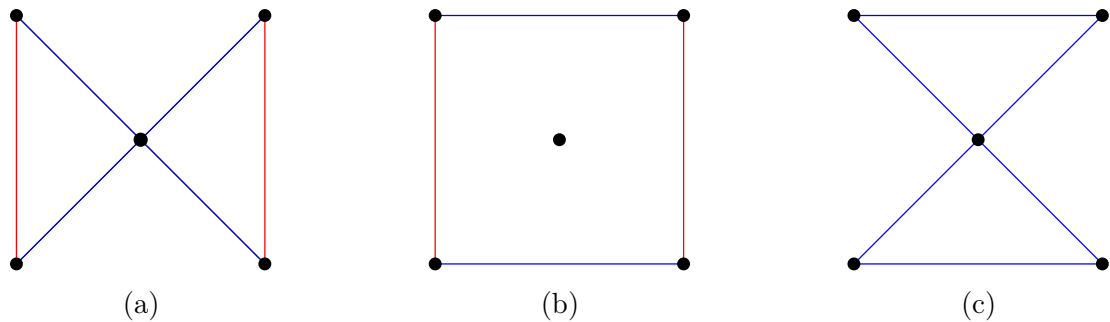


Рис. 18: Операция симметрической разности двух остовных подграфов

На рис.17,b приведен граф F , полученный в результате пересечения графов G и H , показанных на рис.17,a.

3.3.2. При изучении остовных подграфов H_1 и H_2 одного и того же графа G достаточно часто встречается операция Δ их симметрической разности. С помощью такой операции мы из остовных подграфов H_1 и H_2 получаем остовный подграф $H = H_1 \Delta H_2$, состоящий из ребер, принадлежащих H_1 , но не принадлежащих H_2 , и наоборот. На рис.18,c для примера показан остовный подграф H графа G , являющийся симметрической разностью подграфов H_1 и H_2 полного графа K_4 , нарисованных на рис.18,a и рис.18,b. Общие для H_1 и H_2 ребра помечены на этих рисунках красным цветом, а ребра, встречающиеся в одном подграфе и не встречающиеся в другом — синим цветом. Результирующий подграф H_3 состоит только из ребер, окрашенных синим цветом.

3.3.3. Помимо удаления ребра, в теории графов достаточно часто встречается и еще одна важная операция — операция стягивания ребра.



Рис. 19: Операция стягивания ребра

Определение 3.9. Говорят, что граф $G \setminus e$ получен из графа G стягиванием ребра e , если граф $G \setminus e$ получается из G удалением ребра e и стягиванием инцидентных e вершин x и y в одну (см.рис.19).

4 Изоморфизм и автоморфизм графов

4.1. Постараемся вначале ответить на вопрос, сколько же существует различных графов, построенных на n вершинах.

4.1.1. Сразу заметим, что в случае мультиграфов мы наряду с количеством вершин должны также фиксировать и количество ребер. Действительно, любые две вершины мультиграфа соединить произвольным количеством ребер, поэтому количество мультиграфов, построенных на n вершинах, без ограничения на количество ребер может быть сколь угодно большим. Поэтому мы ограничимся пока что задачей подсчета простых графов.

4.1.2. Количество g_n простых неориентированных графов достаточно легко сосчитать, используя формальное определение простого графа как некоторого подмножества множества $V^{(2)}$. Действительно, количество всех двухэлементных подмножеств n -элементного множества вершин (то есть мощность множества $V^{(2)}$) равно $|V^{(2)}| = \binom{n}{2}$. Нас же интересует множество Σ всех подмножеств $V^{(2)}$. Количество элементов в этом множестве, как известно, равно

$$|\Sigma| = 2^{|V^{(2)}|} = 2^{\binom{n}{2}}.$$

Следовательно, количество g_n всех простых графов на n вершинах равняется $2^{\binom{n}{2}}$.

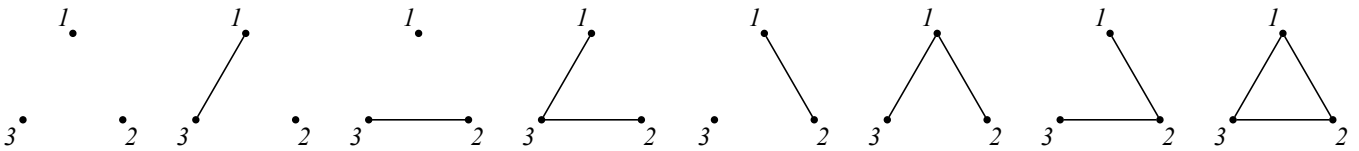


Рис. 20: Восемь простых графов на трех вершинах

Например, существует ровно $g_3 = 2^{\binom{3}{2}} = 2^3 = 8$ различных простых графов, построенных на трех вершинах (смотри рис.20).

4.1.3. Столь же просто подсчитать количество различных простых орграфов. Действительно, всего существует $n(n - 1)$ упорядоченных пар отличных друг от друга вершин. Как следствие, количество различных простых орграфов, построенных на n вершинах, равно $2^{n(n-1)}$.

4.2. Заметим, что многие из приведенных на рис.20 графов похожи друг на друга в том смысле, что отличаются они друг от друга только перенумерацией вершин. Таковыми, в частности, являются три графа, имеющих только лишь одно ребро, а также три графа, имеющих ровно два ребра. Формализовать эту похожесть можно с помощью понятия изоморфизма графов.

4.2.1. Определим вначале понятие изоморфизма для простых неориентированных графов.

Определение 4.1. Говорят, что два простых графа G_1 и G_2 *изоморфны* друг другу, если существует взаимно-однозначное отображение $\varphi: V(G_1) \rightarrow V(G_2)$, которое сохраняет отношение смежности. Последнее означает, что если в графе G_1 некоторая пара вершин $\{x, y\}$ соединена ребром, то в графе G_2 соответствующая ей пара вершин $\{\varphi(x), \varphi(y)\}$ также соединена ребром, и наоборот.

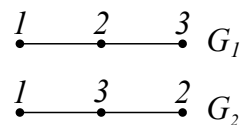


Рис. 21: Два изоморфных графа

Пример 4.2. Рассмотрим графы G_1 и G_2 , изображенные на рис.21. Они изоморфны друг другу, так как существует взаимно-однозначное отображение $\varphi: \{1, 2, 3\} \rightarrow \{1, 2, 3\}$ вида

$$\varphi(1) = 1, \quad \varphi(2) = 3, \quad \varphi(3) = 2,$$

при котором ребро $\{1, 2\}$ графа G переходит в ребро $\{\varphi(1), \varphi(2)\} = \{1, 3\} \in E(G_2)$, а ребро $\{2, 3\}$ графа G переходит в ребро $\{\varphi(2), \varphi(3)\} = \{3, 2\}$ графа G_2 .

4.2.2. Как и любой изоморфизм, изоморфизм графов вводит отношение эквивалентности на множестве всех неориентированных простых графов на n вершинах. Отношение эквивалентности, в свою очередь, разбивает все множество таких графов на классы эквивалентности — классы изоморфных друг другу графов. В теории графов эти классы эквивалентности имеют специальное название.

Определение 4.3. Любой описанный выше класс эквивалентности графов называется *непомеченным* графом. В этом смысле обычный простой граф (то есть любой представитель класса эквивалентности) часто называется *помеченным* графом.

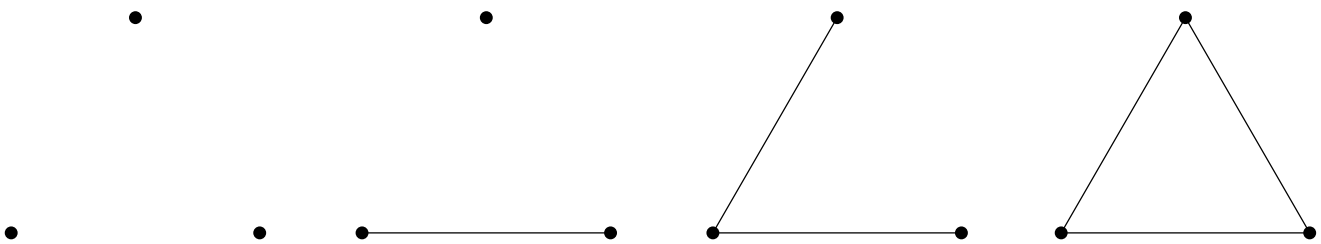


Рис. 22: Четыре простых непомеченных графа на трех вершинах

Так, из рис.20 видно, что существует четыре различных непомеченных графа на трех вершинах: один отвечает пустому графу, второй — трем помеченным графам с одним ребром, третий — трем помеченным графам с двумя ребрами, и наконец четвертый — полному графу K_3 (см.рис.22). Задача перечисления непомеченных графов на n вершинах для произвольного значения параметра n является значительно более сложной задачей по сравнению с задачей перечисления соответствующих помеченных графов.

4.2.3. Давайте теперь для полноты картины дадим определение изоморфизма для более общего случая неориентированных мультиграфов. Как мы помним, в общем случае любой неориентированный граф G задается тройкой, состоящей из множества $V(G)$ вершин, множества $E(G)$ ребер, а также отображения $I(G)$, сопоставляющего любому ребру $e \in E(G)$ неупорядоченную пару вершин $\{x, y\}$, которые это ребро e соединяет.

Определение 4.4. Говорят, что графы

$$G_1 = (V(G_1), E(G_1), I(G_1)) \quad \text{и} \quad G_2 = (V(G_2), E(G_2), I(G_2))$$

изоморфны друг другу, если существует пара взаимно-однозначных отображений

$$(\varphi, \vartheta), \quad \varphi: V(G_1) \rightarrow V(G_2), \quad \text{и} \quad \vartheta: E(G_1) \rightarrow E(G_2),$$

сохраняющих отношение смежности в графах G_1 и G_2 . Последнее в данном случае означает, что если отображение $I(G_1)$ сопоставляет в графе G_1 ребру $e \in E(G_1)$ пару $\{x, y\}$, то в графе G_2 отображение $I(G_2)$ сопоставляет ребру $\vartheta(e) \in E(G_2)$ пару вершин $\{\varphi(x), \varphi(y)\}$, и наоборот.

4.3. Часто на практике достаточно важно уметь отвечать на вопрос, являются ли два различных графа изоморфными друг другу или нет. Оказывается, проверка двух графов на изоморфизм является в общем случае довольно-таки нетривиальной задачей.

4.3.1. Начнем вначале с простого примера.

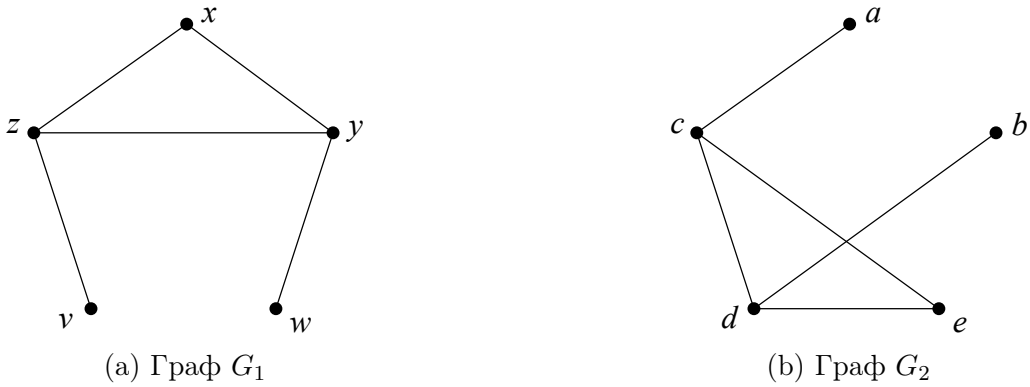


Рис. 23: Установление изоморфизма между графами

Пример 4.5. Рассмотрим два графа, изображенных на рис.23. Возникает вопрос, являются ли два таких графа изоморфными друг другу.

Для ответа на этот вопрос нам нужно, вообще говоря, найти взаимно-однозначное отображение φ из множества вершин $V(G_1)$ в множество $V(G_2)$, сохраняющее отношение смежности. Легче всего в таких простых задачах действовать следующим образом: нужно удалить пометки вершин у первого и у второго графов, пометить вершины первого графа числами от 1 до n , а затем попытаться разметить вершины второго графа теми же числами от 1 до n так, чтобы такой изоморфизм φ был бы очевиден.

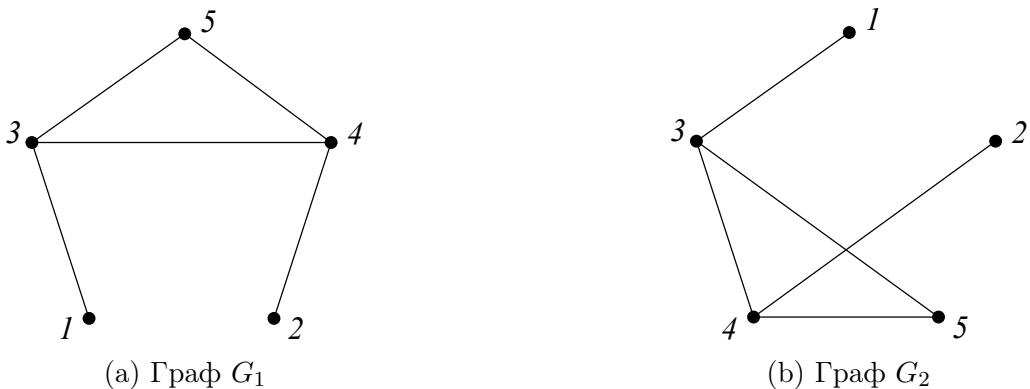


Рис. 24: Установление изоморфизма между графами

В данном конкретном примере сделать это довольно просто (см.рис.24): у графов G_1 и G_2 имеются всего лишь две вершины, степень которых равна единице. В графе G_1 они помечены цифрами 3 и 4. Поэтому нам будет удобно теми же самыми цифрами пометить одновалентные вершины и в графе G_2 . Далее, в графе G_1 эти вершины соединены с вершинами 1 и 2, имеющими степень, равную трем. Пометим аналогичные вершины в G_2 теми же цифрами. Нам остается пометить в графе G_2 оставшуюся вершину цифрой 5 и убедиться, что отображение $\varphi(i) = i$ задает изоморфизм этих графов.

Пример 4.6. Рассмотрим теперь графы, показанные на рис.25,а. Для таких графов уже не вполне очевидно то, что они изображают один и тот же непомеченный граф. Для доказательства этого факта можно, например, попытаться разметить вершины двух графов так, чтобы пара вершин $\{x, y\}$, связанная ребром в графе G_1 , оказалась бы связанной ребром и в графе G_2 , и наоборот. В данном примере это можно все же сделать достаточно просто, заметив, что граф G_2 состоит из двух непересекающихся циклов длины 4. Соответственно, выделив в графе G_1 замкнутый цикл аналогичной длины, мы можем уже однозначно пометить оставшиеся вершины графа G_1 так, чтобы списки смежности этих графов совпали (смотри рис.25,б).

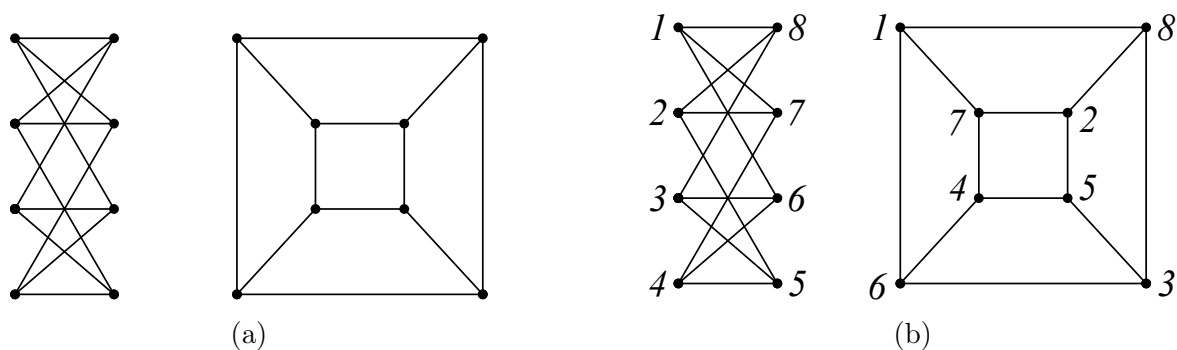


Рис. 25: Установление изоморфизма между графами

4.3.2. В практических задачах довольно часто встречаются графы, в которых содержатся сотни тысяч вершин. Для таких графов никакие визуальные методы оценки уже не работают. Не существует пока и никаких достаточно простых критериев, позволяющих достаточно быстро ответить на вопрос, являются ли два заданных графа изоморфными или нет. Конечно же, есть очевидные необходимые условия изоморфности двух графов — так, например, у изоморфных графов последовательности степеней их вершин должны, очевидно, совпадать. Но в случае, если подобного рода очевидные условия выполнены, то не остается ничего другого, как проверять заданную пару графов на изоморфность простым перебором.

Для этого вершины обоих графов помечают числами из одного и того же множества $[n]$, затем разметку первого графа фиксируют, а пометки второго графа начинают переставлять, вообще говоря, всеми $n!$ способами. Если на каком-то шаге матрицы смежности или же списки смежности этих двух графов совпадут, то графы являются изоморфными друг другу. В противном случае они неизоморфны.

Такого рода алгоритмы крайне неэффективны хотя бы потому, что при больших n значение $n!$ крайне велико. Существуют достаточно сложные алгоритмы, позволяющие более эффективно проверять на изоморфизм специальные подклассы графов, однако никакого быстрого алгоритма, позволяющего делать это в общем случае, пока не существует.

4.4. Наряду с изоморфизмом, в теории графов вводится также понятие автоморфизма — изоморфизма графа G в себя.

4.4.1. Прежде чем давать формальное определение автоморфизма, вернемся еще раз к очень важным понятиям непомеченных и помеченных графов. Давайте подумаем, сколькими способами мы можем разметить один и тот же непомеченный граф на n вершинах. Очевидно, что мы это можем сделать $n!$ способами. Однако при этом возникает вопрос — действительно ли все такие способы пометки дадут нам *различные* помеченные графы?

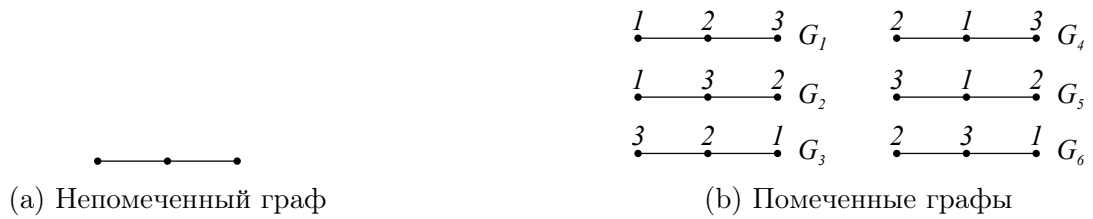


Рис. 26

В качестве примера рассмотрим простой немеченный граф на трех вершинах, имеющий два ребра (рис.26,а). Пометим для него $3!$ способами его вершины. В результате мы получим 6 помеченных графов, изображенных на рис.26,б. Однако мы с вами уже перебрали все помеченные графы, построенные на трех вершинах, и выяснили, что существует только лишь три *различных* помеченных графа, имеющих два ребра. Следовательно, среди шести графов, показанных на рис.26,б, половина лишняя в том смысле, что среди этих графов имеются графы, представляющие собой один и тот же помеченный граф.

Несложно, в частности, убедиться в том, что графы G_1 и G_3 , показанные на рис.26,б, представляют собой один и тот же помеченный граф — первый из них переходит во второй при его развороте на 180° . То же касается и графов G_2 и G_6 , а также графов G_4 и G_5 .

4.4.2. Давайте теперь постараемся разобраться с причиной того, что при разметке вершин немеченного графа $n!$ способами мы получаем лишние графы. Для этого возьмем немеченный граф \tilde{G} , как-то разметим его вершины (например, так, как это показано на рис.27,а), а затем рассмотрим поподробнее все функции $\varphi: [3] \rightarrow [3]$, задающие перестановки вершин полученного таким образом помеченного графа G_1 .



Рис. 27: Изоморфные помеченные графы на трех вершинах

Рассмотрим вначале функцию φ_1 , оставляющую единицу на месте и меняющую местами вершины 2 и 3:

$$\varphi_1(1) = 1, \quad \varphi_1(2) = 3, \quad \varphi_1(3) = 2.$$

В результате мы из графа G_1 получим помеченный граф G_2 (рис.27,б). Естественно, что графы G_1 и G_2 изоморфны друг другу, так как они получены из одного и того же немеченного графа. Однако важно то, что это — два различных графа: списки смежности этих двух графов различны:

$$G_1 : \begin{array}{l} 1 \text{ смежна с } 2 \\ 2 \text{ смежна с } 1 \text{ и } 3; \\ 3 \text{ смежна с } 2 \end{array} \quad G_2 : \begin{array}{l} 1 \text{ смежна с } 3 \\ 2 \text{ смежна с } 3 \\ 3 \text{ смежна с } 1 \text{ и } 2. \end{array}$$



Рис. 28: Идентичные помеченные графы на трех вершинах

Рассмотрим теперь перестановку φ_2 вершин графа G_1 вида

$$\varphi_2(1) = 3, \quad \varphi_2(2) = 2, \quad \varphi_2(3) = 1.$$

В результате такой перестановки мы из графа G_1 (рис.28,а) получим граф G_3 (рис.28,б). Важно, что у такого графа G_3 список смежности полностью совпадает со списком смежности исходного графа. Следовательно, в результате подобной перестановки φ_2 мы получаем тот же самый помеченный граф, что у нас был до перестановки. Такая перестановка вершин, при которой граф как помеченный объект переходит сам в себя, как раз и называется *автоморфизмом* графа.

Более формальное определение автоморфизма графа звучит следующим образом.

Определение 4.7. Взаимно-однозначное отображение $\varphi : V(G) \rightarrow V(G)$ множества $V(G)$ вершин простого графа G в себя называется автоморфизмом графа G , если она сохраняет отношение смежности в графе G . Последнее означает, что если вершины x и y соединены в графе G ребром $\{x, y\}$, то в том же самом графе G и образы $\varphi(x)$, $\varphi(y)$ этих вершин соединены между собой ребром.

4.4.3. Давайте еще раз сравним определения изоморфизма и автоморфизма графов. На первый взгляд эти определения очень похожи друг на друга, и сразу не очень понятно, чем они отличаются друг от друга. Казалось бы, отличие этих определений состоит в том, что в определении изоморфизма фигурируют различные множества $V_1(G_1)$ и $V_2(G_2)$ вершин, тогда как в определении автоморфизма эти множества совпадают. Однако это отличие в двух определениях абсолютно непринципиально. Как правило, вершины любого графа помечаются элементами одного и того же множества $[n] = \{1, 2, \dots, n\}$ первых n натуральных чисел, так что и в определении изоморфизма, и в определении автоморфизма в роли отображения φ обычно выступает какая-то перестановка $\sigma : [n] \rightarrow [n]$ первых n натуральных чисел.

Ключевое же отличие этих двух определений заключается в следующем. В определении автоморфизма мы в результате отображения φ получаем *тот же самый* граф G , тогда как в случае изоморфизма, не являющегося автоморфизмом, граф G_1 под действием отображения φ переходит в некоторый *другой* граф G_2 , отличный от исходного графа G_1 , причем это верно даже в том случае, когда множества $V_1(G_1)$ и $V_2(G_2)$ вершин этих двух графов совпадают. Легче всего это отличие описать в терминах матрицы или списка смежности графов: в случае автоморфизма список смежности графа G не изменится, тогда как в результате действия изоморфизма списки смежности графов G_1 и G_2 будут, вообще говоря, отличаться друг от друга.

4.4.4. Рассмотрим теперь множество всех автоморфизмов какого-то простого помеченного графа G . Очевидно, что тождественная перестановка id принадлежит этому множеству. Далее, для любого автоморфизма φ обратный к нему автоморфизм φ^{-1} также не меняет граф G , то есть принадлежит тому же множеству автоморфизмов. Наконец, если φ_1 и φ_2 являются автоморфизмами графа, то и их композиция задает нам некоторый автоморфизм. Иными словами, операция композиции автоморфизмов задает на множестве всех автоморфизмов группу $\text{Aut}(G)$, которая называется *группой автоморфизмов* помеченного графа G .

Так, в разобранный выше примере группа автоморфизмов помеченного графа G_1 состоит из двух перестановок — тождественной перестановки id , а также перестановки, меняющей местами вершины 1 и 3. Иными словами, эта группа изоморфна группе \mathbb{Z}_2 .

4.4.5. Несмотря на то, что группа автоморфизмов определена для помеченного графа G , она, на самом-то деле, отражает внутреннюю симметрию исходного непомеченного графа \tilde{G} . Так,

например, мы видим, что показанный на рис.26,а непомеченный граф \tilde{G} на трех вершинах симметричен относительно поворота на 180° . Проблема состоит лишь в том, что не очень понятно, как у непомеченного графа такие симметрии искать. И здесь нам как раз и приходится на помощь помеченный граф: мы берем непомеченный граф \tilde{G} , как-то размечаем его вершины, получаем помеченный граф G , вычисляем для него группу автоморфизмов $\text{Aut}(G)$, а затем говорим, что эта группа и отражает внутреннюю симметрию исходного непомеченного графа \tilde{G} .

4.4.6. После сделанных замечаний мы можем вернуться к проблеме, с которой мы начинали — к определению количества различных помеченных графов.

Утверждение 4.8. *Количество N различных помеченных графов, получаемых из непомеченного графа \tilde{G} разметкой его вершин, рассчитывается по формуле*

$$N = \frac{n!}{|\text{Aut}(G_1)|}, \quad (3)$$

где G_1 — произвольно выбранный помеченный граф, полученный из \tilde{G} некоторой разметкой его вершин.

Доказательство. Выберем произвольный помеченный граф G_1 и сосчитаем для него группу автоморфизмов $\text{Aut}(G_1)$. Предположим вначале, что порядок этой группы совпадает с $n!$. Это означает, что любая перестановка вершин переводит граф G_1 в себя, то есть у нас получается лишь один помеченный граф, отвечающий исходному непомеченному графу \tilde{G} . В этом случае формула (3) верна — она дает $n!/n! = 1$, то есть единственный помеченный граф.

Теперь предположим, что $|\text{Aut}(G_1)| < n!$. В этом случае среди всех $n!$ перестановок вершин найдется перестановка $\tilde{\varphi}$, не принадлежащая $\text{Aut}(G_1)$. Иными словами, эта перестановка переведет нам граф G_1 в некоторый изоморфный, но отличный от G_1 граф G_2 . Применим тогда к графу G_1 все перестановки вида

$$\tilde{\varphi} \cdot \varphi, \quad \varphi \in \text{Aut}(G_1).$$

Количество таких перестановок совпадает, очевидно, с мощностью $|\text{Aut}(G_1)|$ группы автоморфизмов графа G_1 . При этом все эти перестановки переведут граф G_1 в граф G_2 . Действительно, вначале мы с помощью перестановки φ переведем граф G_1 в себя, а затем с помощью $\tilde{\varphi}$ из графа G_1 получим граф G_2 .

Если мы на этом шаге все перестановки вершин графа G_1 перебрали, то мы на этом останавливаемся. В противном случае у нас найдется еще какая-то перестановка $\hat{\varphi}$, не принадлежащая $\text{Aut}(G_1)$ и переводящая G_1 в граф G_3 , отличный от G_1 и G_2 . Продолжая указанный процесс, мы в конце концов исчерпаем все множество Φ перестановок. При этом все это множество у нас окажется разбитым на N блоков, каждый из которых имеет мощность $|\text{Aut}(G_1)|$. Каждый такой блок соответствует перестановкам, переводящим граф G_1 в некоторый изоморфный, но отличный от G_1 помеченный граф G_i . Общее же количество N различных помеченных графов связано с количеством $n!$ всех перестановок множества $V(G_1)$ вершин графа G_1 формулой

$$N \cdot |\text{Aut}(G_1)| = n!,$$

из которой и следует формула (3). □

4.4.7. Переведем проведенные выше рассуждения на язык теории групп. Для этого рассмотрим множество \mathcal{X} всех помеченных графов, построенных на n -множестве $\{1, 2, \dots, n\}$ вершин, а

также действие группы S_n перестановок на этом множестве \mathcal{X} . Говорят, что графы G_1 и G_2 изоморфны друг другу, если существует такая перестановка σ множества вершин, которая переводит граф G_1 в граф $G_2 = \sigma \circ G_1$, то есть такая перестановка, которая любое ребро $\{i, j\}$ в графе G_1 переводит в ребро $\{\sigma(i), \sigma(j)\}$. На языке теории групп это означает, что эти графы эквивалентны относительно действия группы S_n , то есть принадлежат одной и той же *орбите*

$$S_n \circ G = \{\sigma \circ G : \sigma \in S_n\}$$

элемента G множества \mathcal{X} всех графов, построенных на n вершинах, под действием группы S_n . Каждая такая орбита на языке теории графов представляет собой некоторый непомеченный граф на n -множестве вершин.

Рассмотрим теперь подмножество $\text{Aut}(G)$ перестановок σ , любой элемент которого оставляет некоторый помеченный граф $G \in \mathcal{X}$ неподвижным:

$$\text{Aut}(G) := \{\sigma \in S_n : \sigma \circ G = G\}.$$

Несложно понять, что такое подмножество перестановок образует подгруппу $\text{Aut}(G)$ группы S_n перестановок. Такая подгруппа на языке теории групп называется *стабилизатором* элемента G множества \mathcal{X} . Эта подгруппа (как и любая подгруппа H группы S_n) задает разбиение группы S_n всех перестановок на блоки $\sigma \cdot \text{Aut}(G)$, называемые смежными классами группы S_n по подгруппе $\text{Aut}(G)$. Размер этих блоков одинаков и равен порядку $|\text{Aut}(G)|$ группы $\text{Aut}(G)$. Следовательно, порядок $|S_n|$ всей группы перестановок равен произведению порядка $|\text{Aut}(G)|$ группы автоморфизмов графа G на количество смежных классов $S_n / \text{Aut}(G)$ группы S_n по подгруппе $\text{Aut}(G)$ (теорема Лагранжа):

$$|S_n| = n! = |\text{Aut}(G)| \cdot |S_n / \text{Aut}(G)|.$$

В рассмотренном нами примере мы разбили множество S_3 всех перестановок вершин графа G_1 на блоки, состоящие из перестановок вида $\tilde{\varphi} \cdot \varphi$, где $\varphi \in \text{Aut}(G_1)$, и показали, что размер каждого блока равен двум — порядку группы $\text{Aut}(G_1)$.

Затем мы заметили, что любому такому блоку перестановок отвечает некоторый граф G_i , изоморфный исходному графу G . На языке теории групп данный факт формулируется в виде достаточно важного, и в то же время достаточно просто доказываемого утверждения — так называемой Orbit-Stabiliser Theorem. Согласно этой теореме, имеется взаимно-однозначное соответствие между всеми элементами орбиты $S_n \circ G$ графа G (то есть всеми изоморфными G графами на n вершинах) и фактор-множеством $S_n / \text{Aut}(G)$, то есть множеством всех смежных классов группы S_n по стабилизатору $\text{Aut}(G)$. Следствием этого утверждения как раз и является равенство

$$|S_n| = n! = |\text{Aut}(G)| \cdot |S_n \circ G| = |\text{Aut}(G)| \cdot N,$$

где N — количество графов на n вершинах, изоморфных G .

4.4.8. Дадим еще одно важное определение.

Определение 4.9. Граф G называется *асимметричным*, если его группа автоморфизмов тривиальна, то есть состоит только из тождественной перестановки множества $[n]$ его вершин.

Важно заметить, что чем выше n , тем большее количество графов асимметрично. Следовательно, при больших n количество \mathcal{M}_n непомеченных графов можно оценить по формуле

$$\mathcal{M}_n \sim \frac{2^{\binom{n}{2}}}{n!}.$$

5 Основные свойства деревьев

5.1. Одним из самых важных понятий теории графов является понятие дерева.

5.1.1. Начнем с формального определения этого понятия.

Определение 5.1. Деревом называется простой связный граф без циклов. Произвольный (не обязательно связный) простой граф без циклов называется лесом.

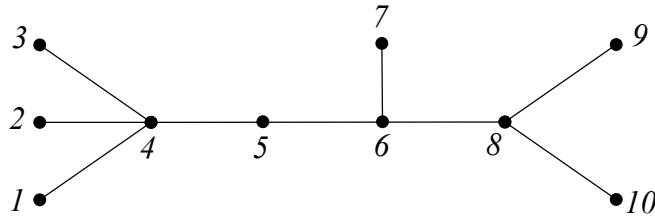


Рис. 29: Пример дерева на десяти вершинах

На рис.29 показан пример дерева T , построенного на 10 вершинах.

5.1.2. Изучим простейшие свойства деревьев. Начнем со следующего важного определения.

Определение 5.2. Вершина графа G , имеющая единичную степень, называется листом.

У дерева T , показанного на рис.29, имеется шесть листьев — вершины 1, 2, 3, 7, 9 и 10.

Лемма 5.3. У любого дерева T , построенного на $n \geq 2$ вершинах, имеется как минимум два листа.

Доказательство. Рассмотрим произвольный простой путь $P = (x_0, x_1, \dots, x_k)$ максимальной длины в T . Для изображенного на рис.29 дерева таким путем является, например, путь $P = (1, 4, 5, 6, 8, 9)$. Очевидно, что такой путь в T всегда существует, причем его длина k равна диаметру $\text{diam}(T)$ дерева T . Покажем, что концы этого пути — вершины x_0 и x_k — обязаны быть листьями.

Предположим, что это не так, то есть предположим, например, что из вершины x_k исходят два или более ребра. Одно из них — это ребро $\{x_{k-1}, x_k\}$. Любое другое исходящее из x_k ребро e не может соединять x_k ни с какой другой из оставшихся вершин пути P — в противном случае мы бы получили в графе цикл. Следовательно, ребро e соединяет x_k с какой-то новой вершиной x_{k+1} графа T . Но в таком случае мы получаем в графе T простой путь $(x_0, x_1, \dots, x_k, x_{k+1})$, длина которого на единицу больше длины пути P . А это, в свою очередь, противоречит тому, что путь P является максимальным. \square

5.1.3. Из доказанной леммы достаточно легко следует одно из основных свойств дерева T .

Теорема 5.4. В дереве T , построенном на n вершинах, имеется ровно $(n - 1)$ ребро:

$$|E| = |V| - 1 = n - 1.$$

Доказательство проведем индукцией по количеству n вершин в графе. База индукции очевидна — дерево, состоящее из $n = 1$ вершины, не имеет ни одного ребра. Предположим теперь, что

утверждение доказано для деревьев, построенных на n вершинах, и покажем, что оно остается верным для произвольного дерева с $(n + 1)$ -й вершиной.

Действительно, по доказанной выше лемме 5.3 у любого такого дерева имеется хотя бы один лист x . Удалим теперь этот лист x вместе с инцидентным ему ребром e . Полученный в результате такой операции граф T' останется, очевидно, связным, и дополнительных циклов в нем также не появится. Следовательно, граф T' является деревом, построенным на n вершинах. Но у такого дерева по индукционному предположению имеется $(n - 1)$ ребро. Следовательно, у исходного дерева T имеется ровно n ребер. \square

Достаточно очевидно и обратное утверждение.

Теорема 5.5. *Любой простой связный граф G , построенный на n вершинах и имеющий $(n - 1)$ ребро, является деревом.*

Доказательство. Действительно, выберем в графе G любую вершину x и покрасим ее, например, в красный цвет. Затем начнем последовательно выполнять следующие действия: будем выбирать в G произвольную неокрашенную вершину z , смежную с одной из уже окрашенных вершин y , и красить ее в красный цвет. Одновременно с этим будем окрашивать в тот же цвет и ребро, соединившее z с окрашенной вершиной y графа G . То, что такая вершина z всегда найдется, достаточно очевидно — граф G связный, и, выбирая любую из еще неокрашенных вершин, мы можем построить путь, соединяющий ее с одной из уже окрашенных вершин, например, с исходной вершиной x , и одна из вершин этого пути обязательно будет еще неокрашена и смежна с одной из уже окрашенных вершин графа G .

Отметим теперь следующий важный момент: в процессе выполнения этих действий мы каждый раз к уже окрашенным вершинам добавляем какую-то новую вершину графа. Цикл же в графе G может получиться только лишь в случае, когда мы на каком-то шаге пришли в уже окрашенную вершину. Иными словами, циклы у нас в процессе выполнения данного алгоритма появиться не могут. Заметим теперь, что за $(n - 1)$ шаг мы окрасим таким образом все вершины и ребра графа G , получив таким образом окрашенный связный граф без циклов. Тем самым мы показали, что граф G является деревом. \square

Следствие 5.6. *Всякий связный граф, построенный на n вершинах, имеет как минимум $(n - 1)$ ребро.*

Доказательство. Рассмотрим произвольный связный граф. Если такой граф не является деревом, то у него имеется хотя бы один цикл. В этом цикле мы всегда можем удалить одно ребро, и граф при этом останется связным. Будем продолжать этот процесс до тех пор, пока в графе не останется ни одного цикла. В результате получим простой связный граф без циклов, то есть дерево. Количество ребер у дерева равно $(n - 1)$. Следовательно, у исходного графа имеется как минимум $(n - 1)$ ребро. \square

Замечание 5.7. Данное следствие другими словами можно переформулировать и так: у всякого связного графа существует остовное дерево. Получить это дерево можно, либо удаляя лишние ребра, либо (что более рационально) используя алгоритм, использованный при доказательстве теоремы 5.5. Этот алгоритм, называемый поиском в глубину, очень часто используется в разнообразных приложениях, связанных с теорией графов.

Так, например, с помощью этого алгоритма можно найти все компоненты связности неориентированного графа. Делается это так: вначале мы полагаем, что каждая вершина в графе неокрашена (или не пройдена алгоритмом, $v.visited=false$), а число компонент связности графа

равно нулю. Затем в цикле мы перебираем все вершины графа. На каждом шаге цикла проверяем, окрашена ли текущая вершина. Если нет, то мы вызываем для нее процедуру поиска в глубину. В этой процедуре мы окрашиваем текущую вершину ($v.visited=true$), просматриваем список смежных с ней вершин, и в случае, если в этом списке обнаруживается неокрашенная вершина, рекурсивно вызываем для нее процедуру поиска в глубину. Каждый раз, когда мы выходим из процедуры во внешний цикл, мы увеличиваем количество связных компонент графа на единицу.

5.1.4. Отметим еще две важные характеристики дерева.

Определение 5.8. Простой связный граф называется минимально связным, если удаление любого ребра приводит к нарушению связности графа.

Утверждение 5.9. *Граф T является деревом тогда и только тогда, когда он является минимально связным графом.*

Доказательство практически очевидно. Действительно, предположим, что минимально связный граф T не является деревом. Тогда в нем имеется какой-то цикл. Удаление любого из входящих в этот цикл ребер связность графа не нарушит. Получили противоречие с тем, что T — минимально связный граф.

Предположим теперь, что в дереве T существует ребро $e = \{x, y\}$, после удаления которого граф остается связным. Последнее, в частности, означает, что в полученном после удаления ребра графе существует простой путь P , связывающий вершины x и y . Но тогда в исходном графе T существует цикл $P \cup \{x, y\}$. Это противоречит тому, что T является деревом. \square

Следствие 5.10. *Любое ребро в дереве является мостом.*

Утверждение 5.11. *Граф T является деревом тогда и только тогда, когда для любых двух его вершин существует единственный простой путь, соединяющий эти вершины.*

Доказательство. Пусть в T для любой пары вершин существует единственный соединяющий их простой путь. Тогда T является деревом. Действительно, в противном случае в T существует цикл, а любые две принадлежащие этому циклу вершины соединены между собой по крайней мере двумя простыми путями, что невозможно.

Обратно, пусть T является деревом, и пусть в нем существуют два различных простых пути P и Q , соединяющих какие-то две вершины x и y . Рассмотрим в P ребра, не принадлежащие Q , а в Q — ребра, не принадлежащие P . Объединение таких ребер образует, очевидно, один или несколько циклов, что противоречит тому, что T является деревом. \square

Следствие 5.12. *Пусть T есть остовное дерево связного графа G , и пусть e есть ребро, не принадлежащее T . Тогда граф $T + e$ содержит единственный цикл.*

Доказательство. Заметим, прежде всего, что любой цикл в графе $T + e$ обязан содержать ребро $e = \{x, y\}$ — в противном случае мы бы получили цикл и в дереве T , чего быть не может. Кроме того, C есть цикл в $T + e$ тогда и только тогда, когда $C - e$ есть xy -путь в исходном дереве T . Но в дереве T xy -путь единственен. Следовательно, $T + e$ содержит единственный цикл, проходящий через ребро e . \square

5.1.5. Рассмотрим теперь простой связный граф G , в котором имеется хотя бы одна пара отличных друг от друга остовных деревьев T и T' .

Утверждение 5.13. Пусть e есть ребро остовного дерева T , не принадлежащее остовному дереву T' . Тогда в остовном дереве T' найдется ребро e' , не принадлежащее остовному дереву T , такое, что граф $T - e + e'$ является остовным деревом графа G .

Решение. Удаление ребра $e = \{x, y\}$ разваливает T на две связные компоненты, такие, что $x \in V_1, y \in V_2$. В дереве T' существует единственный путь, соединяющий вершины x и y . Такой путь обязан иметь ребро e' из V_1 в V_2 . Так как e есть единственное ребро в дереве T между V_1 и V_2 , то ребро e' не принадлежит дереву T . Граф $T - e + e'$ является связным, содержит $n - 1$ ребро, а следовательно, является остовным деревом графа G .

Заметим, что так как в дереве T' имеется единственный путь между вершинами x и y , то добавление к этому пути ребра e приводит к появлению единственного цикла в графе $T' + e$. Удаление же из этого графа ребра e' уничтожает цикл, превращая граф $T' - e' + e$ в связный ациклический подграф — еще одно остовное дерево графа G . □

5.2. Иногда наряду с обычными деревьями и лесами рассматривают так называемые *корневые деревья* и *леса*.

Определение 5.14. *Корневым деревом* называется дерево с выделенной вершиной, называемой *корнем*. *Корневым лесом* называется лес, каждая компонента связности которого представляет собой корневое дерево.

5.2.1. Наглядно корневое дерево можно представлять себе так. Представим дерево в виде некоторого физического объекта, вершинами которого являются некоторые шарниры, а ребрами — соединяющие эти шарниры трубки. Подвесивая такое дерево за какую-то вершину x , мы как раз и получим корневое дерево с корнем в вершине x .

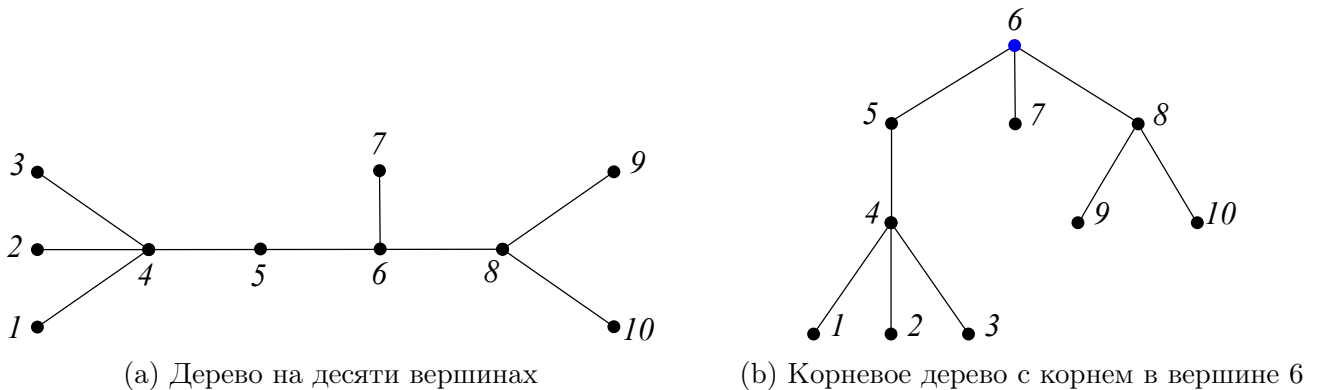


Рис. 30: Дерево и корневое дерево

В качестве примера подвесим показанное на рис.30,а дерево за вершину 6. В результате получим корневое дерево, показанное на рис.30,б.

5.2.2. В любом корневом дереве T_r с корнем в вершине r у нас имеется единственный путь из корня в произвольную вершину x . Длина этого пути определяет *уровень* вершины x в корневом дереве T_r . *Высотой* дерева называется длина наибольшего пути от корня r дерева до вершины x . Ясно, что в последнем случае вершина x обязана быть листом дерева T_r .

Так, изображенное на рис.30,б дерево T_r имеет высоту, равную трем. Уровень вершины 4 равен двум.

5.2.3. Иногда корневое дерево T_r удобно ориентировать от корня r дерева к его листьям (см.рис.31). В такой ориентации входящая степень любой вершины, отличной от корня, будет равняться единице. Корень дерева ориентированного дерева T'_r будет в этом случае источником, то есть вершиной, входящая степень которой равна нулю, а любой лист будет стоком, то есть вершиной, исходящая степень которой равна нулю.

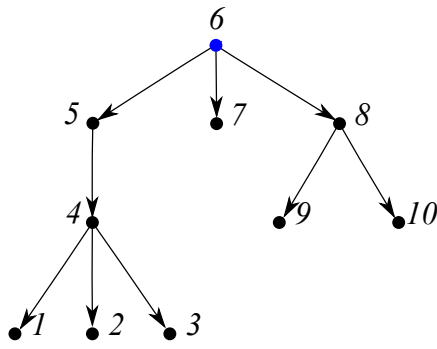


Рис. 31

В дереве T'_r вершина x называется родителем (parent) вершины y , а вершина y — ребенком (child) вершины x , если существует ориентированное ребро (x, y) , ведущее из x в y . Так, на рис.31 вершина 5 является родителем для вершины 4, а вершина 4 — ребенком для вершины 5.

Если существует ориентированный путь из вершины x в вершину z , то вершина x называется предком (ancestor) для вершины z , а вершина z — потомком (descendant) для вершины x . На рис.31 вершины 1, 2 и 3 являются потомками для вершины 5, а вершина 5 — их общим предком.

В случае, если наибольшая исходящая степень вершины в дереве равняется m , то такое дерево называется m -арным деревом. Так, дерево, показанное на рис.31, является 3-арным деревом.

5.2.4. Выбор корня в дереве T задает на множестве $V(T)$ вершин этого дерева частичный порядок, превращая это множество в частично упорядоченное. В одной из следующих глав мы подробнее поговорим о таком множестве, а сейчас лишь дадим его определение.

Определение 5.15. Частично упорядоченным множеством называется множество P (конечное или счетное) с введенным на нем бинарным отношением \preceq (так называемым *отношением частичного порядка*), удовлетворяющим следующим трем аксиомам:

1. Рефлексивность: для любого $x \in P$ справедливо $x \preceq x$.
2. Транзитивность: если $x \preceq y$ и $y \preceq z$, то $x \preceq z$.
3. Антисимметричность: если $x \preceq y$ и $y \preceq x$, то $x = y$.

В случае корневого дерева мы будем считать, что $x \preceq y$, если x лежит на пути, соединяющем корень r дерева с вершиной y . Иными словами, это означает, что в соответствующем T_r ориентированном дереве T'_r вершина x является предком для вершины y . Кроме того, любые вершины, лежащие на пути, соединяющем корневую вершину с одним из его листьев, сравнимы между собой, то есть, как говорят, образуют *цепь* в рассматриваемом частично упорядоченном множестве.

6 Перечисление деревьев. Формула Cayley

6.1. Перейдем теперь к задаче перечисления всех (помеченных) деревьев, построенных на n вершинах.

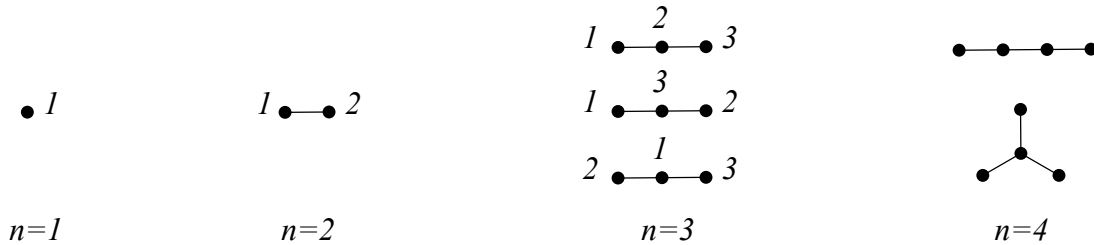


Рис. 32: Неизоморфные друг другу деревья для $n \leq 4$

6.1.1. Несложно убедиться, что существует по одному дереву, построенному на одной и на двух вершинах, а также три дерева, построенные на трех вершинах (см.рис.32). Все эти три дерева отличаются друг от друга только расположением меток вершин. В случае $n = 4$ таких деревьев существует уже 16 штук — 12 деревьев, отвечающих линейной цепочке вершин, и 4 дерева, построенных на “треножнике”. Задача этого пункта — доказать следующий общий результат, известный как формула Кэли.

Теорема 6.1 (Формула Кэли). *Количество a_n различных помеченных деревьев на n вершинах равно n^{n-2} .*

6.1.2. К настоящему моменту известно около шестнадцати различных способов доказательства этого результата (см., например, [?]). Метод, связанный с производящими функциями, мы рассмотрим в курсе комбинаторики. Здесь же мы приведем доказательство этой формулы, впервые предложенное Хайнсом Прюфером (Heinz Prüfer) в 1918 году. Это доказательство является, пожалуй, наиболее простым и понятным, и входит практически в любой учебник по комбинаторике и теории графов. В основе доказательства лежит идея построения по данному дереву некоторой строки длины $n - 2$ над алфавитом из n чисел, по которой можно однозначно восстановить исходное дерево.

6.1.3. Рассмотрим дерево T , вершины которого помечены элементами множества $[n]$, и покажем вначале, как сопоставить такому дереву числовую последовательность вида

$$P(T) = (y_1, y_2, \dots, y_{n-2}), \quad y_i \in [n],$$

называемую кодом Прюфера. Для этого на первом шаге выберем среди всех листьев дерева T вершину x_1 с минимальным номером, удалим эту вершину вместе с инцидентным ей ребром $e_1 = \{x_1, y_1\}$, а затем запишем смежную с удаленным листом вершину y_1 в качестве первого символа в нашем коде. Затем возьмем дерево T_1 , полученное из T удалением вершины x_1 , и выберем среди его листьев вершину x_2 с минимальным номером. Удалим этот лист вместе с инцидентным ему ребром $\{x_2, y_2\}$, а в код Прюфера запишем инцидентную x_2 вершину y_2 . Будем повторять этот процесс до тех пор, пока у нас не останется дерево, состоящее ровно из двух вершин и соединяющего их ребра.

Продемонстрируем этот алгоритм на дереве T , показанном на рис.33. Среди всех его листьев минимальный номер имеет вершина $x_1 = 3$. Удалим эту вершину 3 вместе с инцидентным ей

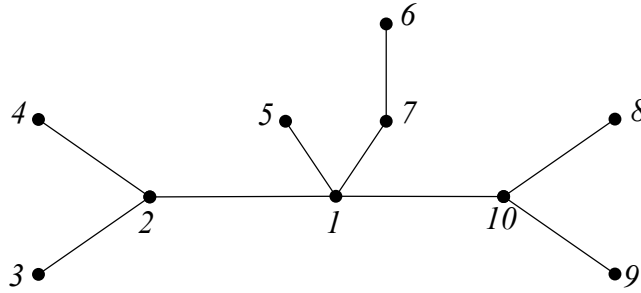


Рис. 33: Дерево T на десяти вершинах

ребром $e_1 = \{3, 2\}$, а в последовательность Прюфера добавим смежную с ней вершину y_1 , то есть вершину 2. Среди листьев дерева T_1 , полученного из T удалением вершины 3, минимальный номер имеет вершина $x_2 = 4$. Ее удаление добавляет нам второй член $y_2 = 2$ последовательности Прюфера. Продолжая этот процесс, на заключительном, 8-м шаге мы получаем дерево, построенное на вершинах 10 и 9, а также код Прюфера

$$P(T) = (2, 2, 1, 1, 7, 1, 10, 10).$$

Нам осталось доказать, что описанное выше отображение, сопоставляющее любому помеченному дереву T числовую последовательность $P(T)$, взаимно-однозначно. Для этого нам достаточно показать, что по любому коду Прюфера $P(T)$ мы сможем однозначно восстановить исходное дерево T .

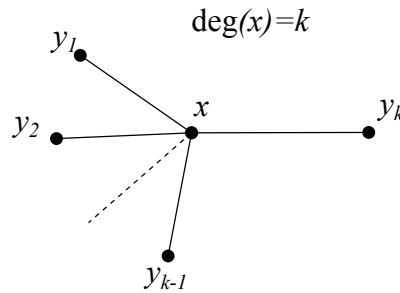


Рис. 34

6.1.4. Заметим, прежде всего, что листья исходного дерева T в этом коде вовсе не встречаются. Действительно, удаляя лист, мы записываем в последовательность Прюфера не номер листа, а номер смежной с ним вершины. Рассмотрим теперь более общий случай вершины x , степень $\deg(x) = k$ которой строго больше единицы (рис.34). В процессе построения кода Прюфера мы постепенно удаляем смежные с x вершины y_i , $i = 1, \dots, k$, до тех пор, пока степень вершины x не станет равной единице. На этом шаге сама вершина x стала листом, и, если нам придется удалить вершину x , то мы уже в последовательность Прюфера включим не вершину x , а смежную с ней вершину y_k . Таким образом, любая вершина x встречается в коде Прюфера $P(T)$ ровно $\deg_T(x) - 1$ раз.

6.1.5. Давайте теперь подумаем, что в идеале нам требуется, чтобы восстановить дерево на n вершинах. Если бы мы в процессе удаления вершины записывали бы не инцидентную ей вершину, а пару вершин — смежную и удаляемую, то мы легко бы по такому набору пар вершин восстановили исходное дерево. Действительно, рассмотрим пару последовательностей

$$\begin{array}{cccccccc} 2 & 2 & 1 & 1 & 7 & 1 & 10 & 10 \\ 3 & 4 & 2 & 5 & 6 & 7 & 1 & 8 \end{array}$$

для показанного на рис.33 дерева T . Первая из них — это последовательность Прюфера $P(T)$. Вторая последовательность — это список удаляемых на каждом шаге ребер. Зная эти пары чисел, мы знаем все ребра исходного дерева T кроме, разве что, последнего ребра. Однако и его мы легко восстановим. Действительно, последним у нас всегда остается ребро, соединяющее вершину наибольшей степени с некоторой вершиной z . Вершина z — это единственная вершина, отличная от вершины с максимальным номером, не встречающаяся во второй последовательности — последовательности удаляемых вершин. Так, в случае дерева, показанного на рис.33, у нас в конце остается ребро, соединяющее вершину 10 с вершиной $z = 9$, не встречающейся во второй строчке таблицы и отличной от 10.

Итак, все, что нам не хватает — это восстановить по верхней последовательности $P(T)$ — коду Прюфера — соответствующую ей нижнюю последовательность вершин, удаляемых на соответствующем шаге алгоритма:

$$P(T) = (y_1, y_2, \dots, y_{n-3}, y_{n-2}) \\ (x_1, x_2, \dots, x_{n-3}, x_{n-2}) + \{x_{n-1}, x_n \equiv n\}.$$

Так как любую вершину в процессе построения кода Прюфера $P(T)$ мы удаляем лишь однажды, то все числа в этой последовательности $\{x_i\}$ различны. Кроме того, в эту последовательность не входит вершина $x_n \equiv n$ с максимальным номером, а также вершина x_{n-1} , которая остается соединенной с n ребром на последнем шаге построения последовательности Прюфера.

6.1.6. Начнем строить последовательность $\{x_i\}$ с восстановления числа x_1 , то есть с восстановления первой удаленной нами вершины. Напомним, что на первом шаге мы брали листья дерева, выбирали из них вершину с минимальным номером и ее удаляли. С точки зрения кода Прюфера листья — это все числа, не вошедшие в последовательность $P(T) = \{y_i\}$. Минимальное из этих чисел и будет нужным нам числом x_1 . Так, для дерева, показанного на рис.33, в код Прюфера $P(T)$ не входят числа

$$\{3, 4, 5, 6, 8, 9\}.$$

Число 3 является минимальным среди этих чисел. Следовательно, именно вершину под номером 3 мы и удалили из дерева T на первом шаге построения кода Прюфера $P(T)$ для дерева T . В общем же случае мы рассматриваем множество чисел вида

$$X = \{1, 2, \dots, n-1, n\},$$

исключаем из него числа, входящие в последовательность $P(T)$, выбираем из оставшихся минимальное — число x_1 , и записываем его под числом y_1 :

$$(y_1, y_2, \dots, y_{n-3}, y_{n-2}) \\ (x_1,$$

Тем самым мы находим первое ребро — ребро $\{x_1, y_1\}$ — восстанавливаемого нами дерева.

6.1.7. Теперь нам нужно восстановить число x_2 , то есть вершину, удаленную на втором шаге построения кода Прюфера $P(T)$. На этом шаге мы рассматривали поддерево T_1 , полученное из T удалением листа x_1 , и в этом поддереве снова выбирали лист с минимальным номером. С точки зрения кода Прюфера поддереву T_1 отвечает числовая последовательность, полученная из $P(T)$ удалением первого члена y_1 :

$$P(T_1) = (y_2, y_3, \dots, y_{n-3}, y_{n-2}).$$

Действительно, удаляя в дереве T вершину x_1 , мы удаляем и соединяющее ее с вершиной y_1 ребро, а следовательно, уменьшаем степень y_1 на единицу. Но тогда мы должны уменьшить на единицу и вхождение y_1 в код Прюфера. Этого мы и достигаем, удаляя из последовательности $P(T)$ число y_1 , стоящее в этой последовательности на первом месте.

А теперь для восстановления x_2 мы должны провести те же рассуждения, что и на предыдущем шаге. Именно, у нас имеется множество $X \setminus \{x_1\}$ оставшихся после первого шага чисел. Исключим из него числа, входящие в код Прюфера $P(T_1)$ поддерева T_1 , а затем выберем среди оставшихся чисел минимальное. Выбранное число x_2 и будет соответствовать вершине, которую мы удалили на втором шаге при построении кода Прюфера исходного дерева T . Записывая его под числом y_2 в коде Прюфера, мы восстанавливаем второе ребро дерева T :

$$\begin{aligned} &(y_1, y_2, \dots, y_{n-3}, y_{n-2}) \\ &(x_1, x_2, \end{aligned}$$

Так, в рассматриваемом примере у нас после первого шага осталось множество чисел

$$X_1 := X \setminus \{x_1\} = \{1, 2, 4, 5, 6, 7, 8, 9, 10\}.$$

Исключим из них числа, входящие в обрезанный слева на единицу код Прюфера

$$P(T_1) = (2, 1, 1, 7, 1, 10, 10).$$

Среди оставшихся в множестве X_1 чисел минимальным является число 4. Следовательно, удаленным на втором шаге оказалось ребро $\{x_2, y_2\} = (4, 2)$ дерева T .

6.1.8. Теперь мы можем уже описать и общий шаг алгоритма восстановления дерева T по коду Прюфера $P(T)$. Предположим, что мы уже восстановили ребра $\{x_1, y_1\}, \dots, \{x_{k-1}, y_{k-1}\}$ дерева T . Рассмотрим множество чисел

$$X_{k-1} := X \setminus \{x_1, x_2, \dots, x_{k-1}\}.$$

Исключим из этого множества все числа, встречающиеся в коде Прюфера $P(T_k)$, полученном из исходного кода Прюфера $P(T)$ удалением стоящих на первых $(k - 1)$ позициях чисел y_1, \dots, y_{k-1} . Выбирая из оставшихся в множестве X_{k-1} чисел минимальное число x_k и записывая его под числом y_k , мы тем самым восстанавливаем ребро $\{x_k, y_k\}$ дерева T , отвечающего последовательности Прюфера $P(T)$:

$$\begin{aligned} &(y_1, y_2, \dots, y_k, \dots, y_{n-3}, y_{n-2}) \\ &(x_1, x_2, \dots, x_k, \end{aligned}$$

6.1.9. Продолжая этот процесс, мы восстановим все числа x_k , $k = 1, \dots, (n - 2)$, то есть $n - 2$ ребра $\{x_i, y_i\}$ дерева T , $i = 1, \dots, n$. Оставшееся ребро $\{x_{n-1}, n\}$ отвечает двум не вошедшим в набор $\{x_i\}$ числам x_{n-1} и n . Последовательно присоединяя к ребру $\{x_{n-1}, n\}$ ребра $\{x_{n-2}, y_{n-2}\}$, $\{x_{n-3}, y_{n-3}\}$, \dots , мы построим в конечном итоге дерево T , отвечающее коду Прюфера $P(T)$.

В рассматриваемом примере мы по коду Прюфера

$$P(T) = (2, 2, 1, 1, 7, 1, 10, 10)$$

восстанавливаем числа

$$(x_1, \dots, x_8) = (3, 4, 2, 5, 6, 7, 1, 8),$$

а следовательно, 8 ребер дерева T . Добавляя эти ребра к ребру $\{9, 10\}$, мы в результате получаем дерево T , показанное на рис.33.

7 Подсчет остовных деревьев в графе. Матричная теорема о деревьях

7.1. Доказанную в предыдущем параграфе формулу Кэли мы можем также интерпретировать как формулу подсчета количества всех остовных деревьев в полном графе K_n . Основная задача данного параграфа — научиться решать более общую задачу подсчета количества $t(G)$ всех различных остовных деревьев для произвольного связного графа G .

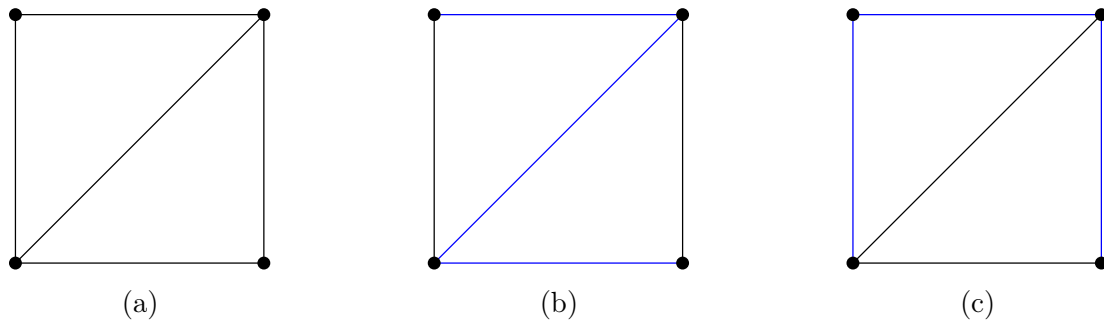


Рис. 35

7.1.1. Начнем с примера графа G , показанного на рис.35,а. У такого графа имеется восемь различных остовных деревьев — четыре из них проходят через диагональ этого графа (см.рис.35,б), а четыре охватывают внешний цикл графа G (рис.35,с).

В рассмотренном примере мы все множество остовных деревьев разбили на два блока, один из которых содержит деревья, проходящие через выбранное ребро e , а второй — деревья, через e не проходящие. Подобного рода рассуждения лежат в основе рекурсивной процедуры подсчета количества $t(G)$ всех остовных подграфов графа G , основанной на следующем утверждении.

Утверждение 7.1. Пусть e — ребро, не являющееся петлей графа G . Тогда

$$t(G) = t(G - e) + t(G \setminus e), \tag{4}$$

где $G - e$ — граф, полученный из G удалением ребра e , а $G \setminus e$ — граф, полученный из G стягиванием этого ребра.

Доказательство. Все множество остовных деревьев графа G разбивается на два блока — блок деревьев, содержащих ребро e , и блок деревьев, e не содержащих. Очевидно, что все остовные деревья, которые не содержат ребро e , являются остовными деревьями графа $G - e$. Поэтому количество таких деревьев совпадает с количеством $t(G - e)$. Давайте теперь докажем, что количество остовных деревьев, содержащих e , совпадает с количеством остовных деревьев графа $G \setminus e$.

Действительно, рассмотрим остовное дерево T графа G , содержащее ребро e . Стягивание этого ребра уменьшает количество ребер и количество вершин в графе $T \setminus e$ на единицу. Кроме того, эта операция не приводит к потере связности получающегося графа $G \setminus e$. Следовательно, граф $T \setminus e$ вновь является деревом — остовным деревом графа $G \setminus e$. Кроме того, никакие два различных остовных дерева графа G , содержащих e , не сольются в одно в результате этой операции. Обратно, любое остовное дерево в графе $G \setminus e$ превращается в остовное дерево графа G , содержащее e , расщеплением вершины на две и добавлением между ними ребра. \square



Рис. 36: Операция стягивания ребра

7.1.2. Сделаем несколько замечаний, касающихся вычислительных аспектов расчета $t(G)$ по формуле (4). Прежде всего, даже если мы стартуем с простого связного графа, то в процессе удаления ребер граф может перестать быть связным, а в процессе стягивания ребер в графе могут появиться петли и мультиребра. Так, например, после стягивания ребра e_1 у графа G , показанного на рис.36,а, ребро e_2 , связывающее в исходном графе G те же вершины, что и e_1 , превращается в петлю.

Со случаем несвязных графов справиться просто — у таких подграфов остовных деревьев не существует, поэтому для таких графов $t(G) = 0$. Мультиребра в графе не мешают нам пользоваться рекуррентным соотношением (4). Однако при появлении петель нам нужно быть осторожнее — рекуррентное соотношение (4) перестает работать в случае, когда ребро e представляет собой петлю. Рассмотрим, к примеру, граф, состоящий из одиночной вершины и петли. У такого графа имеется единственное остовное дерево, состоящее из изолированной вершины. Формула же (4) применительно к данному случаю даст нам $t(G) = 2$. Как следствие, в процессе подсчета по формуле (4) нам нужно такие петли либо просто игнорировать, либо сразу удалять в процессе их появления — на количество остовных деревьев это никак не повлияет.

Наконец, рекурсивный подсчет по формуле (4) требует задания начальных условий. В процессе удаления и стягивания ребер мы в конце концов получим графы, любое ребро которых представляет собой петлю. Если в полученном графе при этом останется одна вершина, то количество остовных деревьев в таком графе равно единице. В противном случае $t(G) = 0$.

7.1.3. Полученное нами рекуррентное соотношение (4) приводит к достаточно громоздким вычислениям даже для случая графов, имеющих не очень большое количество m ребер. С увеличением m количество вычислений экспоненциально возрастает. К счастью, наряду с (4) для подсчета $t(G)$ имеется замкнутая аналитическая формула, полученная еще в 1847 году Кирхгофом. Эта формула носит название матричной теоремы о деревьях (Matrix Tree Theorem) и выражает $t(G)$ через определитель некоторой матрицы. Определитель же квадратной матрицы размером $n \times n$ мы можем сосчитать, используя менее чем n^3 операций.

Рассмотрим связный граф G без петель, построенный на множестве вершин $[n] = \{1, 2, \dots, n\}$. Мы знаем, что матрица M_a смежности такого графа симметрична, и все диагональные элементы этой матрицы равны нулю. Возьмем теперь матрицу M_d размером $n \times n$, в которой диагональные элементы $d_{i,i}$ равны степеням $\deg(i)$ соответствующих вершин в графе G , а остальные элементы равны нулю, и вычтем из нее матрицу M_a . В результате получим матрицу $L = M_d - M_a$, которая носит название *матрицы Лапласа*.

Теорема 7.2 (Matrix tree theorem). *Количество $t(G)$ остовных деревьев графа G равно определителю матрицы L^* , полученной удалением из матрицы Лапласа L произвольной строки и*

произвольного столбца.

Прежде чем доказывать эту теорему, посмотрим, как она работает для примера графа G , показанного на рис.35,а. Для такого графа матрица Лапласа имеет вид

$$L = \begin{pmatrix} 3 & -1 & -1 & -1 \\ -1 & 3 & -1 & -1 \\ -1 & -1 & 2 & 0 \\ -1 & -1 & 0 & 2 \end{pmatrix}.$$

Удаляя из нее первую строку и первый столбец, мы получаем матрицу

$$L^* = \begin{pmatrix} 3 & -1 & -1 \\ -1 & 2 & 0 \\ -1 & 0 & 2 \end{pmatrix},$$

определитель которой равен 8.

7.1.4. При доказательстве матричной теоремы о деревьях удобно перейти от неориентированного графа G к некоторому ориентированному графу D , полученному произвольной ориентацией ребер графа G . Оказывается, что при этом имеется достаточно простая связь между матрицей Лапласа L и матрицей M_i инцидентности орграфа D . Именно, справедлива следующая

Лемма 7.3. Пусть D есть орграф, полученный из G произвольной ориентацией ребер графа G , а M_i есть матрица инцидентности орграфа D . Тогда

$$L = M_i \cdot M_i^T.$$

Доказательство. Напомним, что в матрице M_i элемент $m_{i,j} = 1$, если ребро e_j выходит из вершины i , $m_{i,j} = -1$ в случае, если e_j входит в вершину i , и $m_{i,j} = 0$ в остальных случаях. Любой элемент $l_{i,j}$ матрицы $M_i \cdot M_i^T$ есть скалярное произведение i -й и j -й строк матрицы M_i .

В случае $i \neq j$ это скалярное произведение дает нам столько -1 , сколько имеется ребер, соединяющих вершины i и j . Действительно, если $m_{i,k} = 1$ и $m_{j,k} = -1$, то это означает, что в графе G имеется ребро e_k , соединяющее вершины i и j . В противном случае хотя бы один из этих сомножителей равен нулю, поэтому слагаемое $m_{i,k} \cdot m_{j,k} = 0$ и в скалярное произведение вклада не дает.

В случае $i = j$ мы имеем $m_{i,k}^2$, что дает единицу в случае, если ребро e_k инцидентно вершине i . Суммируя по всем k , мы получаем на диагонали число $l_{i,i}$, равное степени вершины i . \square

Следствие 7.4. Матрица L^* , полученная из матрицы Лапласа L удалением произвольной строки и произвольного столбца, равна

$$L^* = M_i^* \cdot (M_i^*)^T,$$

где M_i^* есть матрица, полученная из M_i удалением произвольной строки.

7.1.5. Приступим теперь к доказательству матричной теоремы о деревьях. Пусть B есть квадратная подматрица $(n-1) \times (n-1)$ матрицы M_i^* . Пусть также G есть неориентированный граф, полученный из D заменой любого ориентированного ребра на неориентированное (underlying graph). Столбцы подматрицы B отвечают какому-то набору из $n-1$ ориентированных ребер в

орграфе D . Соответствующие этому набору неориентированные ребра вместе с инцидентными им вершинами образуют в underlying graph G некоторый подграф H . Покажем, что определитель подматрицы B равен ± 1 в случае, если отвечающий этой подматрице подграф H образует остовное дерево графа G , и нулю в противном случае.

Предположим вначале, что подграф H представляет собой остовное дерево. В этом случае в H обязательно найдутся хотя бы два листа. Так как в подматрице B отсутствует только лишь одна строка исходной матрицы M_i инцидентности орграфа D , то в ней обязательно найдется хотя бы одна строка j , отвечающая вершине x подграфа H степени 1. В этой строке имеется единственный ненулевой элемент $m_{j,k} = \pm 1$. Раскладывая определитель матрицы B по j -й строке, мы получаем равенство вида

$$\det B = m_{j,k} \cdot \det B' = \pm 1 \cdot \det B',$$

где B' — подматрица, полученная из B вычеркиванием j -й строки и k -го столбца. На языке подграфа H данная операция отвечает удалению вершины x степени 1 вместе с инцидентным ей ребром. В результате мы получаем подграф $H - x$, который вновь является остовным деревом, но уже графа $G - x$. Продолжая описанный процесс далее, мы на последнем шаге придем к остовному дереву, построенному на двух вершинах, для которого $\det B = \pm 1$.

Теперь допустим, что соответствующий подматрице B подграф H не является остовным деревом. Следовательно, в таком подграфе обязательно найдется цикл C . Покажем, что в этом случае обязательно найдется нетривиальная линейная комбинация столбцов, равная нулю. Отсюда будет следовать, что строки матрицы линейно зависимы, то есть что $\det B = 0$.

Рассмотрим вначале ребра, не входящие в цикл C . Соответствующие им столбцы матрицы B включим в линейную комбинацию с коэффициентами, равными нулю. Предположим теперь, что направление всех ребер, входящих в цикл, совпадает с каким-то из направлений обхода цикла C . В этом случае линейная комбинация соответствующих этим ребрам столбцов равнялась бы нулю (сколько ребер вошло в вершину x цикла, столько же ребер из нее вышло). Однако в цикле C могут найтись и ребра, которые направлены в сторону, противоположную выбранному направлению обхода цикла C . Тогда мы можем включить отвечающие этим ребрам столбцы в линейную комбинацию с коэффициентами, равными -1 , и вновь получим линейную комбинацию столбцов, равную нулевому столбцу.

Для завершения доказательства теоремы нам теперь достаточно воспользоваться формулой Бине-Коши, позволяющей вычислять определитель квадратной матрицы, являющейся произведением двух прямоугольных матриц. Согласно этой формуле,

$$\det L^* = \det M_i^* \cdot (M_i^*)^T = \sum (\det B)^2,$$

где суммирование проводится по всем $(n-1) \times (n-1)$ -подматрицам B матрицы M_i^* . Мы только что показали, что $(\det B)^2 = 1$ в случае, если B отвечает остовному поддереву графа G , и $(\det B)^2 = 0$ в противном случае. Тем самым матричная теорема о деревьях доказана. \square

7.2. В заключение темы, связанной с матричной теоремой о деревьях, сформулируем следствие из этой теоремы, сводящее подсчет количества остовных деревьев к подсчету собственных значений матрицы Лапласа $L(G)$ графа G .

7.2.1. Предварительно напомним некоторые основные понятия, связанные с собственными значениями и собственными числами матриц. Прежде всего, собственным вектором \mathbf{x} матрицы A

называется отличный от $\mathbf{0}$ вектор, удовлетворяющий уравнению

$$A \cdot \mathbf{x} = \lambda \cdot \mathbf{x}.$$

Соответствующее этому \mathbf{x} число λ называется собственным числом матрицы A . Собственные числа $\lambda_1, \dots, \lambda_n$ можно определить из уравнения

$$\det(A - \lambda E) = 0 \quad \Longleftrightarrow \quad \det(\lambda E - A) = 0.$$

Многочлен

$$\varphi(\lambda) = (-1)^n \det(A - \lambda E) = \det(\lambda E - A) = (\lambda - \lambda_1) \cdot (\lambda - \lambda_2) \cdot \dots \cdot (\lambda - \lambda_n)$$

называется характеристическим многочленом матрицы A . Собственные числа матрицы A есть в точности корни (возможно, кратные) этого характеристического многочлена. Кратность корня λ_i характеристического многочлена $\varphi(\lambda)$ называется алгебраической кратностью $\alpha(\lambda_i)$ собственного числа λ_i .

При заданном собственном значении λ собственные векторы \mathbf{x} находятся из решения системы однородных линейных уравнений

$$(A - \lambda E) \cdot \mathbf{x} = \mathbf{0}.$$

Вместе с вектором $\mathbf{0}$ они составляют подпространство

$$V_\lambda(A) = \text{Ker}(A - \lambda E),$$

называемое собственным подпространством n -мерного пространства, отвечающим собственному числу λ . Размерность этого подпространства называется геометрической кратностью $\gamma(\lambda)$ собственного числа λ . Следовательно,

$$\gamma(\lambda) = \dim(\text{Ker}(A - \lambda E)) = n - \text{rank}(A - \lambda E).$$

Геометрическая кратность $\gamma(\lambda)$ собственного числа не превосходит алгебраической кратности $\alpha(\lambda)$. В случае, когда все собственные числа вещественны и различны, геометрическая и алгебраическая кратности собственных чисел совпадают и равны 1. В более общем случае эти кратности могут не совпадать. В качестве первого характерного примера рассмотрим матрицу $A = E$. Характеристическое уравнение для этой матрицы имеет вид

$$(\lambda - 1)^n = 0.$$

Следовательно, E имеет собственное число $\lambda = 1$, алгебраическая кратность $\alpha(\lambda)$ которого равна n . Кроме того, $\gamma(\lambda) = n - \text{rank}(A - 1 \cdot E) = n - \text{rank}(0) = n$, поэтому в этом случае алгебраическая и геометрическая кратности совпадают. В случае же матрицы

$$A = \begin{pmatrix} 2 & 1 \\ 0 & 2 \end{pmatrix}$$

характеристическое уравнение имеет вид

$$(\lambda - 2)^2 = 0 \quad \Longrightarrow \quad \lambda = 2, \quad \alpha(\lambda) = 2.$$

С другой стороны,

$$\gamma(\lambda) = n - \text{rank}(A - \lambda E) = 2 - 1 = 1,$$

так как ранг матрицы

$$A - \lambda E = A - 2E = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$$

равен единице.

Обычно спектр матрицы A записывают в следующем виде:

$$\text{Sp}(A) = \begin{pmatrix} \lambda_1 & \lambda_2 & \dots & \lambda_r \\ \gamma(\lambda_1) & \gamma(\lambda_2) & \dots & \gamma(\lambda_r) \end{pmatrix}.$$

Заметим также, что если $\text{rank}(A) = k < n$, то $\lambda = 0$ есть собственное число этой матрицы кратности $n - k$. Действительно, в этом случае $\gamma(0) = n - \text{rank}(A) = n - k$.

Наконец, отметим следующее полезное свойство собственных чисел: если добавить ко всем диагональным элементам матрицы одно и то же число c , то все собственные числа матрицы увеличатся на это число. Действительно, $\lambda + c$ является корнем характеристического уравнения

$$\det(\lambda E - (A + cE)) = 0 \quad \iff \quad \det((\lambda - c)E - A) = 0$$

тогда и только тогда, когда λ есть корень характеристического уравнения

$$\det(\lambda E - A) = 0.$$

7.2.2. Теперь сформулируем утверждение, позволяющее вычислять количество остовных деревьев по спектру матрицы Лапласа графа G .

Следствие 7.5. Пусть $\lambda_1, \lambda_2, \dots, \lambda_n$ есть собственные значения матрицы $L(G)$, причем $\lambda_1 = 0$. Тогда количество остовных деревьев в графе G рассчитывается по формуле

$$t(G) = \frac{1}{n} \lambda_2 \lambda_3 \dots \lambda_n.$$

Заметим, что фигурирующее в формулировке данного следствия утверждение о том, что у матрицы L одно из собственных чисел равняется нулю, достаточно очевидно. Дело в том, что сумма всех строк матрицы L даст нам нулевую строку — в любом столбце i количество -1 в точности равно степени вершины i , стоящей на диагонали матрицы L . Как следствие, строки матрицы L линейно зависимы, и потому одно из собственных чисел L обязано равняться нулю. Мы будем это собственное число обозначать через λ_1 .

Позже мы покажем, что кратность собственного числа $\lambda_1 = 0$ равна единице. Следовательно, оставшиеся собственные числа $\lambda_2, \dots, \lambda_n$ матрицы L отличны от нуля. Мы уже напоминали, что собственные числа матрицы представляют собой корни характеристического уравнения

$$\det(\lambda \cdot E - L) = 0 \quad \iff \quad \lambda \cdot (\lambda - \lambda_2) \cdot \dots \cdot (\lambda - \lambda_n) = 0,$$

где E — единичная матрица, L — матрица Лапласа. Предположим, что мы хотим найти коэффициент при λ^1 этого полинома. С одной стороны, этот коэффициент равен

$$(-1)^{n-1} \lambda_2 \cdot \dots \cdot \lambda_n.$$

С другой стороны, его можно вычислить по формуле

$$\frac{d}{d\lambda} \det(\lambda \cdot E - L)|_{\lambda=0}.$$

Попытаемся воспользоваться этой формулой и выразить коэффициент при λ^1 через количество остовных деревьев графа G . Для этого нам понадобится формула, позволяющая вычислять производную определителя $\det(A(t))$ матрицы $A(t)$, зависящей от произвольного параметра t .

7.2.3. Перепишем матрицу $A(t)$ в виде

$$A(t) = (a_1(t), a_2(t), \dots, a_n(t)),$$

где $a_i(t)$ есть столбцы матрицы $A(t)$. Пользуясь мультилинейностью определителя, мы можем записать производную $d/dt(\det(A(t)))$ как

$$\frac{d}{dt} \det(A(t)) = \det(a'_1(t), a_2(t), \dots, a_n(t)) + \det(a_1(t), a'_2(t), \dots, a_n(t)) + \dots + \det(a_1(t), a_2(t), \dots, a'_n(t)).$$

Действительно, определитель представляет собой сумму $n!$ слагаемых, каждое из которых с точностью до сомножителя ± 1 есть произведение вида $a_{\sigma(1),1} \cdot a_{\sigma(2),2} \cdot \dots \cdot a_{\sigma(n),n}$, где σ есть некоторая перестановка n -множества $[n]$. В каждое из таких произведений обязательно входит ровно по одному элементу от каждого столбца матрицы A . После дифференцирования каждого такого произведения по t мы получим в точности n слагаемых, причем j -е слагаемое будет иметь вид

$$a_{\sigma(1),1} \cdot a_{\sigma(2),2} \cdot \dots \cdot a'_{\sigma(j),j} \cdot \dots \cdot a_{\sigma(n),n}.$$

При фиксированном j мы имеем ровно $n!$ слагаемых описанного вида. Собирая все эти слагаемые, мы получаем определитель матрицы $A'_j(t) := (a_1, \dots, a'_j, \dots, a_n)$, полученной из матрицы $A(t)$ заменой элементов j -го столбца на производные этих элементов по t .

Для подсчета определителя вида $\det(a_1(t), \dots, a'_j(t), \dots, a_n(t))$ воспользуемся формулой Лапласа:

$$\det(a_1(t), \dots, a'_j(t), \dots, a_n(t)) = \sum_{i=1}^n (-1)^{i+j} a'_{i,j}(t) \cdot M_{i,j}(t).$$

Здесь $M_{i,j}(t)$ есть минор матрицы $A'_i(t)$, то есть определитель матрицы, полученной из $A'_i(t)$ вычеркиванием i -й строки и j -го столбца. Как следствие,

$$\frac{d}{dt} \det(A(t)) = \sum_{j=1}^n \sum_{i=1}^n (-1)^{i+j} a'_{i,j}(t) \cdot M_{i,j}(t) = \text{tr}(C^T \cdot A'(t)), \quad (5)$$

где $C = \{C_{i,j}\}$ — матрица, составленная из элементов вида $(-1)^{i+j} M_{i,j}$ (алгебраических дополнений или кофакторов элементов $a_{i,j}$), $A'(t)$ — матрица, составленная из элементов $a'_{i,j}$, $\text{tr}(A)$ — след матрицы A . Матрицу C^T иногда называют союзной или присоединенной к A матрицей. Формулу (5) называют формулой Якоби вычисления производной определителя матрицы, зависящей от параметра t .

7.2.4. В рассматриваемом нами случае

$$A(t) = t \cdot E - L \quad \Longrightarrow \quad a'_{i,j}(t) = \begin{cases} 1, & \text{если } i = j; \\ 0, & \text{если } i \neq j; \end{cases} \quad \Longleftrightarrow \quad A'(t) = E.$$

Как следствие,

$$\frac{d}{d\lambda} \det(\lambda \cdot E - L) = \sum_{i=1}^n (-1)^{i+i} M_{i,i}(\lambda) = \sum_{i=1}^n M_{i,i}(\lambda),$$

где $M_{i,j}$ есть минор i -го диагонального элемента матрицы $\lambda \cdot E - L$, то есть определитель размера $(n-1) \times (n-1)$ матрицы $\lambda \cdot E - L$, из которого вычеркнуты i -я строка и i -й столбец. Подставляя в этот определитель значение $\lambda = 0$, мы получим умноженный на $(-1)^{n-1}$ определитель матрицы L^* — матрицы, полученной из матрицы Лапласа вычеркиванием i -го столбца и i -й строки. Следовательно,

$$\frac{d}{d\lambda} \det(\lambda \cdot E - L)|_{\lambda=0} = (-1)^{n-1} \sum_{i=1}^n \det(L^*) = (-1)^{n-1} \cdot n \cdot \det(L^*).$$

Но из матричной теоремы о деревьях следует, что для графа G определитель $\det(L^*) = t(G)$ — количеству остовных деревьев графа G . Поэтому

$$(-1)^{n-1} \lambda_2 \cdot \dots \cdot \lambda_n = (-1)^{n-1} \cdot n \cdot t(G) \quad \implies \quad t(G) = \frac{1}{n} \lambda_2 \cdot \dots \cdot \lambda_n.$$

8 Подсчет остовных деревьев в орграфе. The BEST Theorem

8.1. Доказанную в предыдущей главе матричную теорему о деревьях можно обобщить на случай орграфов. Такое обобщение было получено в 1948 году W.Tutte. С помощью этого обобщения мы затем сможем подсчитать количество всех эйлеровых циклов в связном графе G .

8.1.1. Рассмотрим произвольный ориентированный граф D , построенный на n вершинах. Для неориентированного графа G мы ввели матрицу Лапласа L как матрицу, полученную вычитанием матрицы M_a смежности графа G из диагональной матрицы M_d с элементами $M_d[i, i] = \deg(x_i)$. Для ориентированного графа мы можем ввести аналоги матрицы Лапласа — матрицы L^- и L^+ , полученные вычитанием матрицы M_a смежности орграфа D из диагональных матриц M_d^- и M_d^+ с элементами $M_d^-[i, i] = \text{outdeg}(x_i)$ и $M_d^+[i, i] = \text{indeg}(x_i)$ соответственно:

$$L^- = M_d^- - M_a, \quad L^+ = M_d^+ - M_a.$$

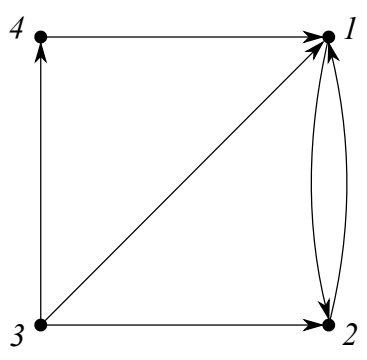


Рис. 37

В качестве примера рассмотрим орграф D , изображенный на рис.37. Матрицы Лапласа для этого орграфа записываются так:

$$L^- = \begin{pmatrix} 1 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ -1 & -1 & 3 & -1 \\ -1 & 0 & 0 & 1 \end{pmatrix}, \quad L^+ = \begin{pmatrix} 3 & -1 & 0 & 0 \\ -1 & 2 & 0 & 0 \\ -1 & -1 & 0 & -1 \\ -1 & 0 & 0 & 1 \end{pmatrix}.$$

Заметим сразу же, что, как и в неориентированном случае, матрицы L^- и L^+ — это матрицы неполного ранга. Действительно, количество исходящих из вершины x_i ребер равно $\text{outdeg}(x_i)$, так что сумма элементов любой строки матрицы L^- равна нулю. Аналогично, $-\sum_j M_a[i, j]$ есть взятое со знаком минус количество ребер, входящих в вершину j . Складывая это с $\text{indeg}(x_j)$, мы получаем, что сумма элементов любого столбца матрицы L^+ равна нулю. Следовательно, столбцы матрицы L^- и строки матрицы L^+ являются линейно зависимыми.

8.1.2. Для того, чтобы сформулировать теорему Татта, нам осталось напомнить понятие корневых остовных деревьев, все ребра которых ориентированы либо к корню x (так называемых *spanning arborescences converging to x*), либо от корня x (так называемых *spanning arborescences diverging from x*). В деревьях первого типа исходящая степень любой вершины, отличной от x , равна единице, а в деревьях второго типа равна единице входящая степень любой вершины, отличной от x . При этом исходящая степень вершины x и входящая степень вершины x равны нулю в деревьях первого и второго типа соответственно.

8.1.3. Теорема Татта позволяет подсчитать количество корневых остовных деревьев с корнем в вершине i .

Теорема 8.1 (W.Tutte, 1948). *Количество $t^-(D, i)$ корневых остовных деревьев, все ребра которых ориентированы к корню i , равно определителю подматрицы матрицы Лапласа L^- , из которой выкинуты i -я строка и i -й столбец:*

$$t^-(D, i) = \det(L_{i,i}^-).$$

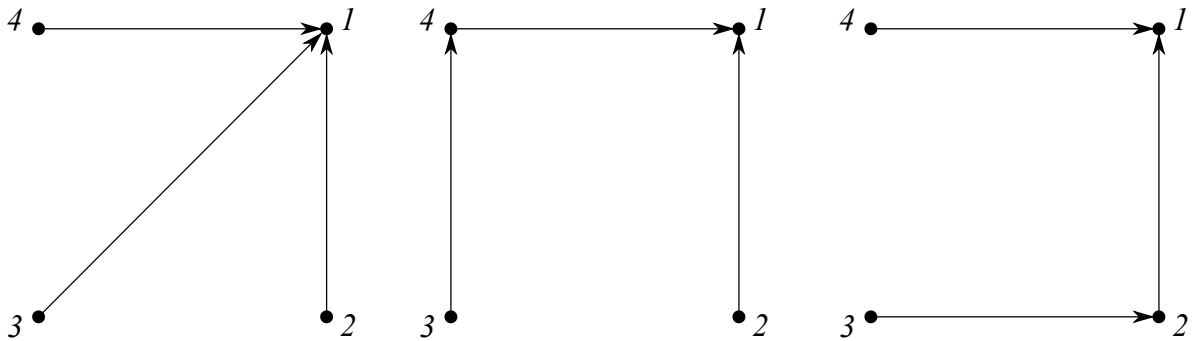


Рис. 38

Так, для показанного на рис.37 графа D количество $t^-(D, 1)$ остовных деревьев, все ребра которых ориентированы к вершине 1, равно

$$t^-(D, 1) = \det(L_{1,1}^-) = \begin{vmatrix} 1 & 0 & 0 \\ -1 & 3 & -1 \\ 0 & 0 & 1 \end{vmatrix} = 3.$$

Все эти деревья изображены на рисунке 38.

8.1.4. Приступим к доказательству теоремы Татта. Для определенности будем считать, что в качестве корня мы выбрали вершину 1. Сразу заметим, что в случае, когда в орграфе D существует вершина с номером $i > 1$, у которой $\text{outdeg}(i) = 0$, количество остовных деревьев, все ребра которых ориентированы к вершине 1, равно нулю. Тот же результат дает нам и теорема Татта: в таком случае как в матрице $L_{1,1}^-$, так и в исходной матрице L^- строка, отвечающая вершине i , нулевая. Действительно, в этом случае диагональный элемент этих матриц равен

$\text{outdeg}(i) = 0$. Кроме того, любой другой элемент i -й строки есть взятое со знаком минус количество ребер, исходящих из вершины i в соответствующую этому элементу вершину j , то есть также равен нулю. Следовательно, и определитель $|L_{1,1}^-|$ соответствующей матрицы равен нулю.

Теперь рассмотрим случай, когда для некоторого $i > 1$ из вершины i исходит $l > 1$ ребер (то есть когда для некоторого i степень $\text{outdeg}(i) = l > 1$). В этом случае мы вместо орграфа D можем рассмотреть орграфы D_1 и D_{l-1} , идентичные D , за исключением ребер из вершины i в остальные вершины. Именно, в орграфе D_1 мы оставим лишь одно исходящее из i ребро (не важно, куда именно), а все остальные ребра из i оставим в орграфе D_{l-1} . Так как любое остовное дерево должно содержать ровно одно исходящее из i ребро, то

$$t^-(D, 1) = t^-(D_1, 1) + t^-(D_{l-1}, 1).$$

Заметим, что те же рассуждения мы можем провести и на языке матриц $L_{1,1}$. Именно, матрицы $L_{1,1}^-(D_1)$ и $L_{1,1}^-(D_{l-1})$ за исключением i -й строки выглядят идентично. В i -й строке матрицы $L_{1,1}^-(D_1)$ диагональный элемент равен 1, какой-то из элементов $a_{i,j}$, $i \neq j$, равен -1 , а остальные элементы равны нулю. При этом i -я строка $L_{1,1}^-(D)$ равна сумме i -х строк матриц $L_{1,1}^-(D_1)$ и $L_{1,1}^-(D_{l-1})$. Но тогда мы можем записать, что

$$\det(L_{i,i}^-(D)) = \det(L_{i,i}^-(D_1)) + \det(L_{i,i}^-(D_{l-1})).$$

Следовательно, нам достаточно доказать утверждение теоремы для случая, когда все диагональные элементы матрицы $L_{1,1}^-$ равны единице, то есть для случая, когда из любой вершины $i > 1$ исходит единственное ребро.

8.1.5. В условиях сделанного выше предположения рассмотрим вначале ситуацию, при которой в орграфе D найдется ориентированный цикл, не проходящий через вершину 1. Ясно, что в таком случае пути из вершин этого цикла в вершину 1 отсутствуют, а потому количество остовных деревьев такого орграфа с корнем в вершине 1 равняется нулю. Покажем теперь, что и соответствующий такому орграфу определитель $\det(L_{1,1}^-) = 0$. Рассмотрим в качестве примера орграф D , построенный на четырех вершинах, у которого через вершины 2, 3 и 4 проходит простой цикл длины 3. Соответствующая такому графу D матрица $L_{1,1}^-$ имеет вид

$$\begin{pmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \\ -1 & 0 & 1 \end{pmatrix}$$

Видно, что линейная комбинация строк, отвечающих вершинам цикла, равна нулю. Аналогичная ситуация будет наблюдаться и в общем случае: сумма отвечающих вершинам цикла строк будет равняться нулю. Это, в свою очередь, означает, что строки такой матрицы линейно зависимы, а потому определитель $\det(L_{1,1}^-) = 0$.

8.1.6. Рассмотрим, наконец, ситуацию, при которой в орграфе любой замкнутый ориентированный цикл, если он вообще существует, проходит через вершину 1. В этом случае, стартуя из любой вершины $i > 1$, мы обязательно придем в вершину 1. Удаляя все ребра, исходящие из 1 (а такие ребра ни в одно нужное нам остовное дерево не войдут), мы превратим D в ориентированное остовное дерево, все ребра которого направлены к вершине 1. Так как такое дерево получается однозначно, то в D существует единственное нужное нам остовное дерево. Все, что нам осталось показать — это то, что в данном случае определитель $\det(L_{1,1}^-) = 1$.

С точки зрения матрицы L^- удаление ребер, исходящих из 1, означает обнуление первой строки этой матрицы. Иными словами, на матрицу $L_{1,1}^-$ эти операции никак не повлияют. Выберем

теперь в остовном дереве, полученном после удаления всех исходящих из 1 ребер, какой-то лист i , и рассмотрим i -й столбец матрицы $L_{1,1}^-$. Диагональный элемент этого столбца равен единице. Остальные элементы этого столбца равны нулю — в вершину i никакие ребра из других вершин орграфа D , отличных от 1, не приходят. Раскладывая тогда определитель матрицы по этому столбцу, мы по индукции приходим к равенству $\det(L_{1,1}^-) = 1$. Теорема Татта доказана.

8.1.7. Заметим, что из теоремы Татта легко получить матричную теорему о количестве $t(G)$ остовных деревьев в неориентированном графе G . Заменяем для этого в графе G любое неориентированное ребро на пару ребер, ориентированных в двух различных направлениях. Выбирая теперь произвольную вершину x полученного орграфа D , мы можем записать, что

$$t(G) = t^-(D, x).$$

Действительно, любому корневому ориентированному дереву, все ребра которого направлены к вершине x , отвечает некоторое остовное дерево графа G , полученное из корневого ориентированного дерева заменой ориентированных ребер на неориентированные. Обратно, пусть T есть остовное дерево графа G . Ориентируя ребра T единственным образом по направлению к выбранной вершине x , мы получим в D корневое ориентированное дерево, все ребра которого направлены в сторону вершины x .

Кроме того, заметим, что имеет место равенство

$$t^+(D, x) = t^-(D, x).$$

Действительно, для его доказательства достаточно в орграфе D заменить ориентацию всех его ребер на противоположную.

8.2. Теорема Татта о количестве остовных деревьев в орграфе D позволяет подсчитать количество эйлеровых циклов в произвольном ориентированном эйлеровом графе D .

8.2.1. Этот результат известен как the BEST theorem. Свое название теорема получила по первым буквам фамилий математиков, ее доказавшим: в 1941 году Cedric Smith и William Tutte подсчитали количество эйлеровых циклов в графе, исходящая степень вершин которого равна двум, а в 1951 году Nicolaas de Bruijn и Tatyana van Aardenne-Ehrenfest обобщили эту формулу на случай произвольного значения исходящих степеней вершин орграфа.

Теорема 8.2 (de Bruijn, van Aardenne-Ehrenfest, Smith, and Tutte). *Количество $e(D)$ эйлеровых циклов в орграфе D рассчитывается по формуле*

$$e(D) = t^-(D, x) \cdot \prod_{y \in D} (\text{outdeg}(y) - 1)!,$$

где x — произвольная вершина орграфа D , $t^-(D, x)$ — количество корневых остовных деревьев, все ребра которых направлены к корню в вершине x .

Следствие 8.3. *Количество $t^-(D, x)$ корневых остовных деревьев в орграфе D не зависит от выбора корня, то есть вершины x .*

8.2.2. Зафиксируем произвольную вершину x в орграфе D (вершина 1 на рис.39), а также некоторое ребро $e = (x, y)$, являющееся стартовым ребром для любого эйлерова цикла W в графе (ребро $e_1 = (1, 3)$ на рис.39). Покажем вначале, как по эйлерову циклу построить остовное дерево с корнем в вершине x .

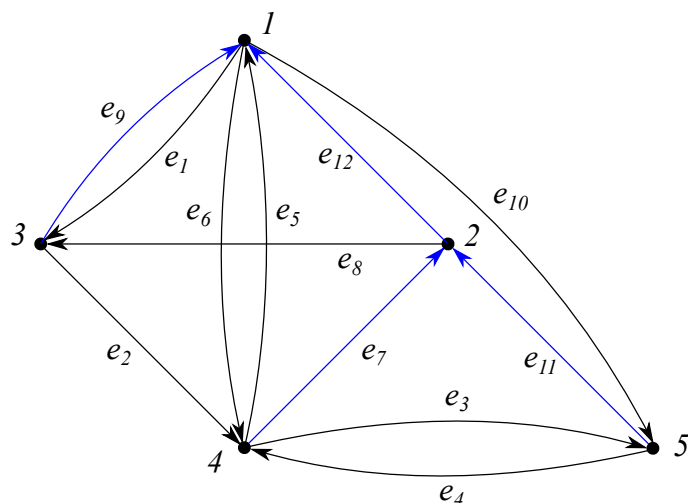


Рис. 39

Мы знаем, что эйлеров цикл проходит по каждому ребру орграфа D ровно один раз, задавая на множестве $E(D)$ ребер некоторый линейный порядок (см. ребра e_1, \dots, e_{12} на рис.39). Кроме того, он хотя бы один раз проходит через каждую вершину орграфа D . Выберем для каждой вершины $y \neq x$ исходящее из y ребро с наибольшим номером (см. ребра, помеченные синим цветом на рис.39). Покажем, что порожденный этими ребрами подграф T представляет собой корневое остовное дерево с корнем в вершине 1.

Действительно, из любой вершины $y \neq x$ исходит лишь одно ребро с наибольшим номером. Мы при доказательстве теоремы Татта уже указывали, что в этой ситуации подграф, порожденный этими ребрами, представляет собой либо дерево, либо простой цикл. Но в данном случае простого цикла быть не может. Действительно, если бы мы, выйдя из какой-то вершины z по ребру e_i , вошли бы в нее же по какому-то ребру e_j , то мы должны были бы получить, что $j > i$. Но мы обходим вершины орграфа D по эйлеровому циклу, начинающемуся в вершине 1. Это означает, что, войдя в какую-то вершину $z \neq x$ по ребру с номером j , мы выйти из нее должны по ребру с номером $j + 1$. Но тогда ребро e_i — это не ребро с наибольшим номером, выходящее из вершины z . Получили противоречие. Таким образом, синие ребра в орграфе D циклов не образуют, покрывают все вершины орграфа, а количество этих ребер равняется $n - 1$. Следовательно, эти ребра порождают остовное дерево в орграфе D с корнем в вершине 1.

8.2.3. Рассмотрим теперь какое-то остовное дерево T , все ребра которого направлены к корню в вершине x . Нам нужно понять, сколько различных эйлеровых циклов мы сможем построить из одного и того же остовного дерева T . Для этого давайте пронумеруем (с повторениями) все ребра орграфа D согласно следующему принципу. Зафиксированному в самом начале ребру (x, y) , с которого, как мы договорились, будут начинаться все эйлеровы циклы в орграфе D , мы присвоим номер 1 (ребро e_1 на рис.39). Остальным $\text{outdeg}(x) - 1$ ребрам назначим $(\text{outdeg}(x) - 1)!$ количеством способов произвольные номера из множества $\{2, \dots, \text{outdeg}(x)\}$. Для любой другой вершины $z \neq x$ орграфа D мы присвоим номер, равный $\text{outdeg}(z)$, исходящему из вершины z ребру, принадлежащему остовному дереву. Остальным ребрам мы вновь $(\text{outdeg}(z) - 1)!$ количеством способов назначим произвольные номера из множества $\{1, \dots, \text{outdeg}(z) - 1\}$. Всего, таким образом, мы получаем

$$d := \prod_{z \in D} (\text{outdeg}(z) - 1)!$$

способов пронумеровать ребра орграфа D .

Убедимся теперь, что любая такая нумерация ребер задает нам один из d ориентированных эйлеровых циклов в орграфе D . Для этого при заданной нумерации ребер определим маршрут в орграфе D следующим образом. Начнем маршрут с ребра (x, y) , помеченного номером 1. При заходе в любую вершину z орграфа покинем эту вершину по еще не использованному ребру с наименьшим номером. Так как при входе в любую вершину $z \neq x$ у нас количество исходящих из z ребер на единицу больше количества входящих, то мы всегда из z сможем перейти в какую-то другую вершину. Остановиться же мы сможем только в вершине x . При этом все входящие в x ребра к этому моменту будут уже использованы. По построению это означает, что использованными оказались все ребра каждой из вершин, находящихся на расстоянии 1 от корня в остовном дереве. Далее по индукции можно показать, что использованными оказались вообще все ребра орграфа D .

8.2.4. Итак, мы для любого остовного дерева построили d различных эйлеровых циклов в D . Кроме того, различные остовные деревья порождают нам различные множества эйлеровых циклов. Таким образом, $e(D)$ действительно равно $t^-(D, x) \cdot d$. Теорема доказана.

9 Линейное пространство ребер. Циклы и разрезы

9.1. Перейдем к изучению еще одного важного понятия, тесно связанного с остовным деревом графа G — понятия фундаментального цикла.

9.1.1. Пусть T есть некоторое остовное дерево связного графа G . Ребра графа G , не вошедшие в T , образуют, как говорят, *кодерево* \bar{T} графа G . Кодерево \bar{T} можно еще трактовать как остовный подграф, полученный из G удалением всех ребер, принадлежащих остовному дереву T . Ребра дерева T называют иногда ветвями (branches) этого дерева, а ребра кодерева \bar{T} называют хордами исходного дерева T .

Рассмотрим произвольное ребро $e = \{x, y\}$, принадлежащее кодереву \bar{T} . Согласно утверждению ??, граф $T + e$, полученный добавлением к T ребра e , содержит единственный цикл. Этот цикл обозначается обычно C_e и называется *фундаментальным циклом* графа G по отношению к остовному дереву T .

9.1.2. Рассмотрим теперь все множество ребер из \bar{T} . Каждому из этих ребер e соответствует свой фундаментальный цикл C_e , полученный добавлением e к T . Множество всех таких циклов образует, как говорят, *фундаментальное множество циклов*, ассоциированное с T . Количество же циклов в этом множестве, совпадающее с количеством ребер в \bar{T} , называется *цикловым рангом* $\gamma(G)$ графа G . Понятно, что для любого связного графа G цикловой ранг $\gamma(G)$ не зависит от выбора остовного дерева T в графе и равен $m - n + 1$, где m — количество ребер в графе G .

В качестве примера рассмотрим граф G , показанный на рис.40. Ребра остовного дерева T на этом рисунке помечены синим цветом. У такого графа имеется четыре ребра e_i , $i = 2, \dots, 5$, входящие в кодерево \bar{T} . Как следствие, цикловой ранг $\gamma(G)$ данного графа равен четырем. Соответствующие каждому из ребер $e_i \in \bar{T}$ фундаментальные циклы C_{e_i} помечены красным цветом на рис.41.

9.2. Важно заметить, что по построению порождающее цикл C_{e_i} ребро e_i не входит ни в какие другие фундаментальные циклы графа G , порожденные остальными ребрами $e_j \in \bar{T}$. Это

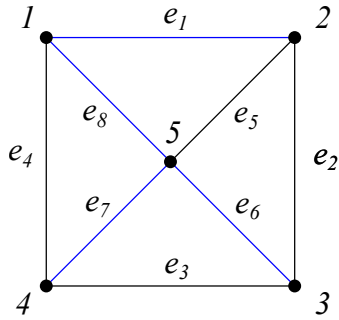


Рис. 40: Остовное дерево графа

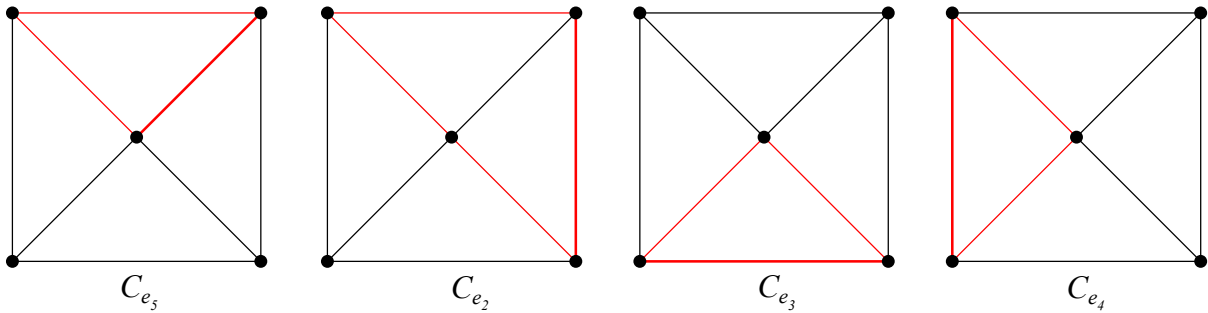


Рис. 41: Фундаментальные циклы графа

свойство позволяет рассматривать множество фундаментальных циклов как базис в линейном подпространстве некоторого линейного пространства ребер, к описанию которого мы сейчас и перейдем.

9.2.1. Рассмотрим простой связный граф G , построенный на n вершинах и имеющий m ребер. Введем множество $\mathcal{E}(G)$ всех подмножеств множества $E(G)$ ребер графа G . Его можно рассматривать как линейное пространство \mathcal{E} над двухэлементным полем $\mathbb{F}_2 = \{0, 1\}$ относительно операции Δ симметрической разности. Нулевым элементом в этом пространстве является пустое подмножество ребер \emptyset . Кроме того, так как $S \Delta S = \emptyset$ для любого подмножества S ребер (то есть элемента множества $\mathcal{E}(G)$), то обратный элемент $-S$ к S в этом пространстве совпадает с S . Так как множество $E(G)$ всех ребер образует в этом пространстве базис (так называемый стандартный базис пространства \mathcal{E}), то размерность этого пространства равна m .

Любой элемент $S \in \mathcal{E}$ в стандартном базисе можно представить в виде бинарной строки длины m . В случае, если ребро $e_i \in S$, на i -й позиции этой строки стоит единица. В противном случае там стоит ноль. Например, множество ребер остовного дерева T , показанного на рис.40, описывается вектором

$$(1, 0, 0, 0, 0, 1, 1, 1).$$

Стандартная операция “исключающее или” (exclusive-or, или xor) двух таких строк задает бинарную операцию на множестве таких строк, эквивалентную операции симметрической разности Δ на множестве $\mathcal{E}(G)$ подмножеств множества ребер графа G . Иными словами, мы можем вместо \mathcal{E} рассматривать изоморфное ему линейное пространство бинарных строк длины m над полем \mathbb{F}_2 с операцией xor.

9.2.2. Линейное пространство \mathcal{E} можно превратить в евклидово, введя скалярное произведение его элементов S и Q по формуле

$$(S, Q) := s_1q_1 + \dots + s_mq_m \in \mathbb{F}_2,$$

где $s_i \in \{0, 1\}$ и $q_i \in \{0, 1\}$ есть коэффициенты в разложении подмножеств S и Q по стандартному базису. Из этого определения видно, что два элемента S и Q ортогональны друг другу тогда и только тогда, когда они пересекаются по четному числу ребер.

9.2.3. Давайте теперь введем достаточно важное понятие декомпозиции графа G .

Определение 9.1. Декомпозицией графа G называется семейство \mathcal{H} попарно реберно непересекающихся подграфов H_1, \dots, H_k графа G , объединение которых дает все множество ребер $E(G)$:

$$\cup_{H_i \in \mathcal{H}} E(H_i) = E(G).$$

Если семейство \mathcal{H} состоит только из путей либо только из циклов, то \mathcal{H} называется декомпозицией G на пути (path decomposition) или на циклы (cycle decomposition).

Сразу заметим, что декомпозиция на пути для любого графа G существует всегда, так как всегда существует так называемая тривиальная декомпозиция, у которой любое подмножество H_i состоит из одиночного ребра e_i графа G . Декомпозиция же на циклы существует не всегда — любое дерево T представляет тривиальный пример графа, для которого такая декомпозиция отсутствует.

Предположим, что граф G допускает декомпозицию на циклы. В этом случае степень любой вершины $x \in V(G)$ графа G четна. Действительно, эта степень представляет собой удвоенное количество циклов, проходящих через вершину x . Иными словами, мы показали, что любой граф, допускающий декомпозицию на циклы, является *четным* графом.

Определение 9.2. Граф G называется четным, если любая его вершина имеет четную степень.

Оказывается, верно и обратное утверждение.

Теорема 9.3 (Veblen, 1912). *Граф G допускает декомпозицию на циклы тогда и только тогда, когда G четный.*

Доказательство. Нам осталось доказать, что любой четный граф допускает декомпозицию на циклы. Будем доказывать это утверждение индукцией по количеству ребер. В случае пустого графа это утверждение верно — мы G можем разложить на пустое множество циклов. Пусть теперь G является непустым четным графом. Рассмотрим в нем подграф H , индуцированный вершинами графа G с положительными четными степенями. Так как любая вершина этого подграфа больше или равна двум, то такой подграф содержит цикл C . Рассмотрим тогда в графе G остовный подграф $G' := G \setminus E(C)$, полученный удалением ребер цикла C . Этот подграф имеет меньшее количество ребер, а потому допускает разложение на циклы. Но тогда и граф G допускает разложение на те же циклы плюс цикл C . \square

9.2.4. Рассмотрим теперь множество всех четных подграфов заданного графа G .

Утверждение 9.4. *Симметрическая разность двух четных подграфов H_1 и H_2 есть снова четный подграф.*

Доказательство. Предположим вначале, что подграфы H_1 и H_2 не имеют общих вершин. В этом случае их симметрическая разность представляет собой четный подграф $H_1 \cup H_2$. В противном случае рассмотрим произвольную общую вершину x четных подграфов H_1 и H_2 . Обозначим через S_1 подмножество инцидентных x ребер, принадлежащих H_1 , а через S_2 — подмножество ребер, принадлежащих H_2 . Симметрическая разность $S_1 \Delta S_2$ представляет собой объединение

двух блоков — подмножества $B_1 = S_1 \setminus (S_1 \cap S_2)$ ребер, принадлежащих H_1 и не принадлежащих H_2 , а также подмножества $B_2 = S_2 \setminus (S_1 \cap S_2)$ ребер, принадлежащих H_2 и не принадлежащих H_1 . Как следствие,

$$|S_1 \Delta S_2| = |S_1 \setminus (S_1 \cap S_2)| + |S_2 \setminus (S_1 \cap S_2)| = |S_1| + |S_2| - 2|S_1 \cap S_2|.$$

Так как $|S_1|$ и $|S_2|$ четны, то четным является и число $|S_1 \Delta S_2|$. Иными словами, все вершины остовного подграфа, полученного в результате симметрической разности двух четных подграфов H_1 и H_2 , имеют четную степень, а следовательно, подграф $H = H_1 \Delta H_2$ является четным подграфом. \square

Следствие 9.5. *Множество четных подграфов замкнуто относительно операции Δ , то есть образует в \mathcal{E} линейное подпространство \mathcal{C} .*

Учитывая теорему Veblen, мы можем сказать, что это линейное подпространство порождается всеми циклами графа G . Как следствие, подпространство \mathcal{C} называется *пространством циклов* графа G .

9.2.5. Ранее мы сказали, что любой фундаментальный цикл C_{e_i} , связанный с некоторым остовным деревом T графа G , содержит ребро e_i (хорду дерева T), не входящее ни в какой другой фундаментальный цикл. С точки зрения линейной алгебры это означает, что такие циклы являются линейно независимыми: линейная комбинация

$$\alpha \cdot (0, 1, \dots, \underbrace{1}_i, \dots, \underbrace{0}_j, \dots, 0) + \beta \cdot (1, 1, \dots, \underbrace{0}_i, \dots, \underbrace{1}_j, \dots, 0) = (0, 0, \dots, \underbrace{0}_i, \dots, \underbrace{0}_j, \dots, 0)$$

таких циклов равна пустому циклу (нулевому вектору) тогда и только тогда, когда коэффициенты $\alpha = \beta = 0$. Следовательно, мы имеем $m - n + 1$ линейно независимый вектор в пространстве \mathcal{C} циклов графа G . Нам хочется показать, что набор таких векторов образует базис пространства \mathcal{C} . В принципе, для этого нужно доказать, что любой цикл графа может быть представлен в виде линейной комбинации этих фундаментальных циклов C_{e_i} . Однако можно поступить несколько проще — на самом деле, нам достаточно показать, что размерность пространства \mathcal{C} равна $m - n + 1$. Если это так, то любой набор из $m - n + 1$ линейно независимых векторов, и в том числе набор фундаментальных циклов, образует в \mathcal{C} базис.

Для достижения этой цели мы, в свою очередь, рассмотрим в пространстве \mathcal{E} еще одно подпространство — так называемое пространство \mathcal{B} разрезов графа G . Затем мы докажем, что, во-первых, это пространство ортогонально \mathcal{C} , а во-вторых, что в этом пространстве существует $n - 1$ линейно независимых векторов. Так как $(n - 1) + (m - n + 1) = m$, то отсюда сразу будет следовать, во-первых, что размерность пространства \mathcal{C} равна $m - n + 1$, а во-вторых, что фундаментальные циклы C_{e_i} образуют в этом пространстве базис.

9.3. Для определения подпространства \mathcal{B} нам понадобятся несколько важных понятий.

9.3.1. Начнем с понятия реберно разделяющего множества F .

Определение 9.6. Подмножество $F \subset E(G)$ называется *реберно разделяющим множеством* связного графа G , если после удаления всех ребер из F граф $G - F$ перестает быть связным.

Рассмотрим в качестве примера граф, показанный на рис.42. Синим цветом на каждом из этих рисунков помечены ребра, после удаления которых граф перестает быть связным. Следовательно, любое из таких подмножеств ребер образует реберно разделяющее множество.

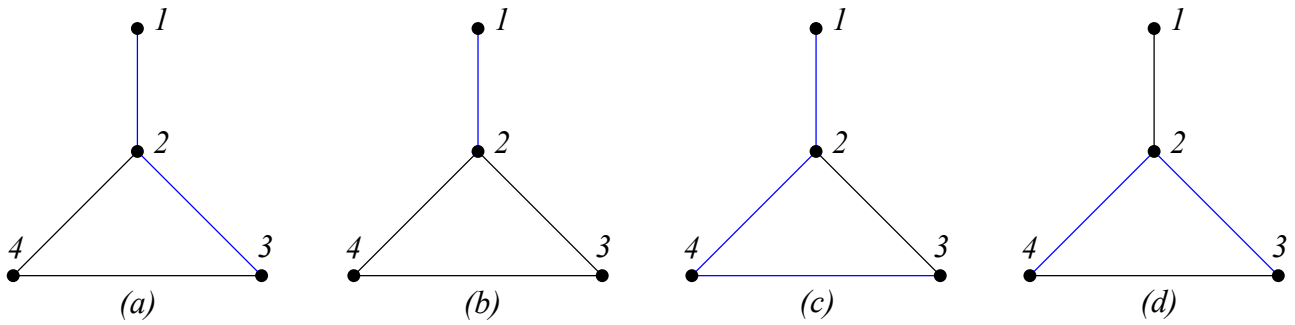


Рис. 42

9.3.2. Следующее важное для нас понятие — это понятие *реберного разреза* (edge cut). Пусть у нас имеются два подмножества S_1 и S_2 множества $V(G)$ вершин графа G (не обязательно непересекающихся). Рассмотрим все ребра графа G , один из концов которых принадлежит S_1 , а второй — S_2 . Множество таких ребер обозначается обычно через $[S_1, S_2]$. Случаю $S_1 = S_2 = S$ отвечает набор ребер, оба конца которых лежат в S . Нас же сейчас будет интересовать противоположный случай, а именно, случай, когда $S_2 = V(G) \setminus S_1$.

Определение 9.7. Пусть S есть некоторое подмножество множества вершин $V(G)$ графа, а $\bar{S} = V(G) \setminus S$. Набор ребер $[S, \bar{S}]$ называется *реберным разрезом* графа G , связанным с подмножеством S . Иногда он также называется *кограницей* S и обозначается через $\partial(S)$.

Очевидно, что $\partial(S) = \partial(\bar{S})$, а также то, что $\partial(V(G)) = \emptyset$. Кроме того, граф G является *двудольным*, если в нем существует такое подмножество S вершин, что $\partial(S)$ совпадает со всем множеством $E(G)$ ребер графа G . Реберный разрез $\partial(x)$, связанный с отдельной вершиной x , является, как говорят, *тривиальным реберным разрезом* и представляет собой набор ребер, инцидентных x . По аналогии с этим частным случаем размер $|\partial(S)|$ подмножества ребер $\partial(S)$ в случае мультиграфа без петель называется *степенью* подмножества вершин S .

Так как в графе $G - [S, \bar{S}]$ пути между S и \bar{S} отсутствуют, то любой непустой реберный разрез $[S, \bar{S}]$ является *реберно-разделяющим* множеством. Обратное, конечно же, неверно. Так, реберно-разделяющее множество, показанное синим цветом на рис.42,а, реберным разрезом не является — наряду с ребром $\{1, 2\}$ оно содержит лишнее ребро $\{2, 3\}$. Все остальные реберно-разделяющие множества, показанные на рис.42,б, с и d, являются одновременно и реберными разрезами. Так, на рис.42,б представлен реберный разрез $\partial(1)$, связанный с вершиной 1, на рис.42,с — реберный разрез $\partial(1, 4)$ подмножества вершин $\{1, 4\}$, а на рис.42,d — реберный разрез $\partial(1, 2)$, связанный с подмножеством $\{1, 2\}$ множества вершин.

9.3.3. Наконец, введем последнее нужное нам в этом месте понятие — понятие *минимального реберного разреза* (bond).

Определение 9.8. Реберный разрез называется *минимальным* (bond), если он представляет собой непустой реберный разрез, любое собственное подмножество которого не является реберным разрезом.

Среди показанных на рис.42 реберных разрезов только два являются минимальными — реберные разрезы $\partial(1)$ (рис.42,б) и $\partial(1, 2)$ (рис.42,d). Реберный разрез $\partial(1, 4)$, показанный на рис.42,с, минимальным не является — он содержит в качестве своего собственного подмножества ребро $\{1, 2\}$, являющееся реберным разрезом, связанным с вершиной 1.

9.3.4. Докажем теперь для связного графа G следующий важный критерий минимальности реберного разреза.

Теорема 9.9. В любом связном графе G любой непустой реберный разрез $\partial(S)$ является минимальным тогда и только тогда, когда граф $G - \partial(S)$ имеет в точности две связные компоненты.

Доказательство. Предположим вначале, что граф $G - \partial(S)$ имеет в точности две компоненты связности G_1 и G_2 . Реберный разрез $F := \partial(S)$, по определению, содержит те и только те ребра, которые соединяют в графе G эти две компоненты. Поэтому, если у F существует собственное подмножество F' , то граф $G - F'$ содержит в качестве подграфов те же самые две компоненты связности G_1 и G_2 графа $G - F$ плюс по крайней мере одно ребро, соединяющее эти компоненты. Следовательно, граф $G - \partial(G)$ является связным, а $\partial(S)$ — минимальным реберным разрезом.

Обратно, предположим, что граф $G - \partial(S)$ содержит более чем две связные компоненты. Это означает, что либо подграф H , индуцированный подмножеством вершин S , либо подграф \bar{H} , индуцированный подмножеством вершин \bar{S} , содержит по крайней мере две связные компоненты. Пусть, для определенности, таковым является подграф H . В этом случае $S = A \cup B$, причем между вершинами подмножеств A и B ребра отсутствуют. Но тогда, например, $[A, \bar{A}]$ есть собственное подмножество множества $\partial(S)$, являющееся реберным разрезом, а это противоречит минимальности $\partial(S)$. □

9.3.5. Докажем, наконец, следующее важное утверждение.

Утверждение 9.10. Симметрическая разность двух различных реберных разрезов есть снова реберный разрез, причем

$$\partial(X) \Delta \partial(Y) = \partial(X \Delta Y). \tag{6}$$

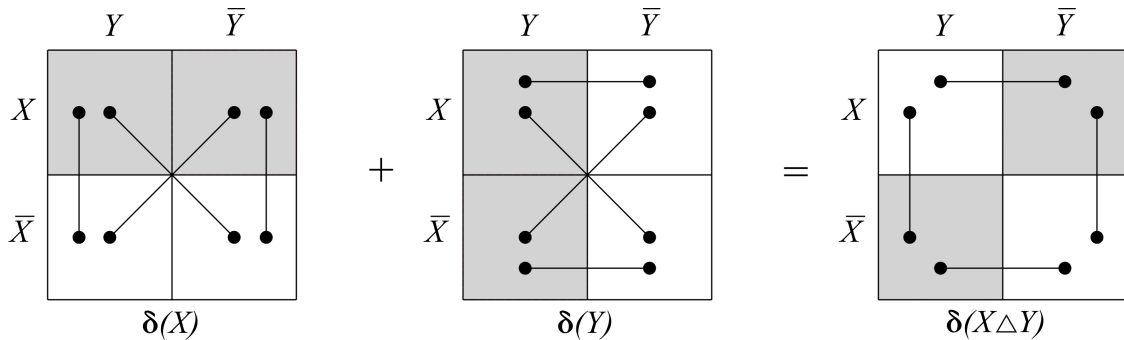


Рис. 43

Доказательство. То, что симметрическая разность двух различных реберных разрезов есть снова реберный разрез, следует из формулы (6). Для ее доказательства рассмотрим рис.43. Все варианты разбиения множества $V(G)$ с помощью двух подмножеств имеют следующий вид:

$$X \cap Y, \quad X \setminus Y, \quad Y \setminus X, \quad \bar{X} \cap \bar{Y}.$$

Все эти варианты можно наглядно представить в виде следующей таблицы:

	Y	\bar{Y}
X	$X \cap Y$	$X \setminus Y$
\bar{X}	$Y \setminus X$	$\bar{X} \cap \bar{Y}$

Правые нижние четыре клетки этой таблицы образуют разбиение квадрата на четыре части, как раз и описывающие все четыре блока разбиения множества $V(G)$ вершин графа G .

Теперь рассмотрим набор ребер $\partial(X)$. Эти ребра могут исходить либо из $X \cap Y$, либо из $X \setminus Y$, и входить в любые две клетки нижней части описанного выше квадрата. В итоге получается четыре типа ребер (см.рис.43). Аналогично, ребра из набора ∂Y могут выходить из левых клеток квадрата в любую из правых ее клеток. В итоге вновь получаем четыре варианта. Теперь осталось образовать из этих наборов ребер симметрическую разность. Видно, что в нее не будут входить те ребра, которые соединяют диагональные клетки квадрата. Но это и есть $\partial(X \Delta Y)$ — множеству вершин $X \Delta Y$ отвечают правый верхний и левый нижний блоки разбиения $V(G)$, а ребра из $\partial(X \Delta Y)$ соединяют их с двумя оставшимися блоками разбиения $V(G)$. \square

Следствие 9.11. *Множество всех реберных разрезов вместе с операцией Δ образует в линейном пространстве \mathcal{E} всех подмножеств множества $E(G)$ ребер графа G линейное подпространство \mathcal{B} , называемое линейным пространством реберных разрезов графа G .*

9.4. Напомним, что в начале этого параграфа мы ввели важное понятие фундаментального множества циклов графа G . С этим понятием тесно связано не менее важное понятие фундаментального множества разрезов в графе G .

9.4.1. Пусть T есть произвольное остовное дерево связного графа G . Удаление любого ребра e этого дерева разбивает все множество вершин дерева T на два блока X и Y . Рассмотрим тогда минимальный реберный разрез $\partial(X) =: \partial_e$, связывающий в исходном графе G вершины из блоков X и Y . Такой реберный разрез называется *фундаментальным разрезом*, связанным с деревом T , а множество всех таких разрезов — *фундаментальным множеством разрезов*, связанным с деревом T .

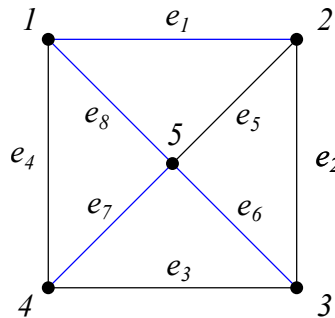


Рис. 44: Остовное дерево графа

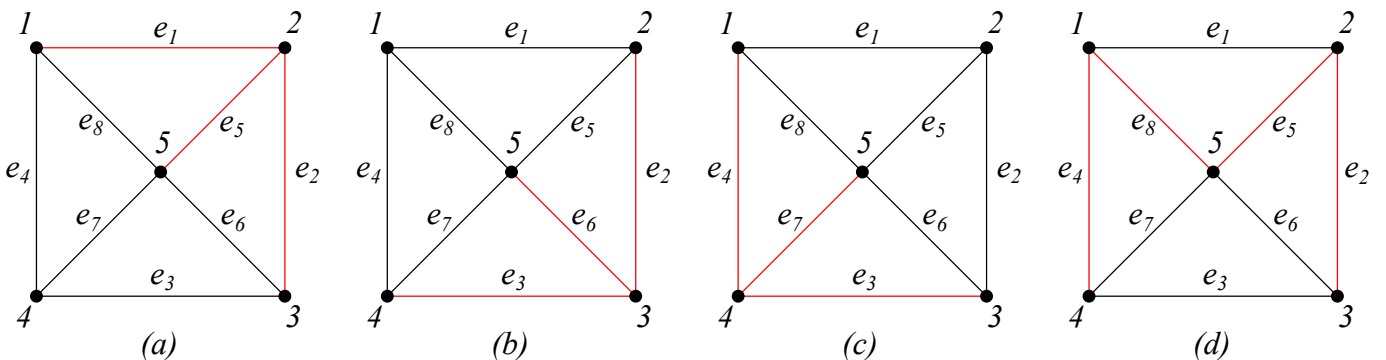


Рис. 45: Фундаментальные разрезы графа

9.4.2. В качестве примера рассмотрим граф G , показанный на рис.44. Остовное дерево T этого графа состоит из ребер e_1, e_6, e_7 и e_8 (синие ребра на рисунке). Удаляя ребро e_1 , мы разобьем множество вершин дерева T на два блока — блок, состоящий из единственной вершины 2, а также блок, состоящий из вершин $\{1, 3, 4, 5\}$. Соответствующий этому разбиению реберный разрез $\partial(2) = \partial_{e_1}$ состоит из ребер e_1, e_2, e_5 (ребра, помеченные красным цветом на рис.45,a). Реберный разрез ∂_{e_6} , отвечающий ребру e_6 остовного дерева, равен $\partial(3) = \{e_2, e_3, e_6\}$ (рис.45,b), а реберный разрез ∂_{e_7} , соответствующий ребру e_7 дерева T , есть $\partial(4) = \{e_3, e_4, e_7\}$ (рис.45,c). Наконец, ребру e_8 отвечает реберный разрез $\partial_{e_8} = \partial(1, 2) = \{e_2, e_4, e_5, e_8\}$ (рис.45,d).

Важно заметить, что любой фундаментальный реберный разрез состоит из ребра e_i остовного дерева, которое порождает этот разрез, а также еще из каких-то ребер, которые соединяют X и Y в графе G , но не соединяют их в остовном дереве T . Иными словами, все остальные ребра этого разреза дереву T не принадлежат, то есть все они принадлежат кодереву \bar{T} .

Как следствие, ребро e_i остовного дерева T , порождающее соответствующий разрез ∂_{e_i} , по построению входит только в этот фундаментальный разрез и не входит ни в какой другой фундаментальный разрез, связанный с другими ребрами e_j остовного дерева T . На языке пространства \mathcal{B} реберных разрезов это означает, что такие циклы образуют набор линейно независимых векторов этого пространства. Количество таких линейно независимых векторов совпадает с количеством ребер в остовном дереве связного графа G и равно $n - 1$.

9.4.3. Как и в случае фундаментальных циклов, мы хотим доказать, что любой реберный разрез графа может быть представлен в виде линейной комбинации некоторых фундаментальных разрезов ∂_{e_i} , то есть что набор фундаментальных разрезов графа G образует базис пространства \mathcal{B} . Отсюда, в частности, следовало бы, что размерность данного подпространства равна $n - 1$, а количество всех реберных разрезов связного графа G равно 2^{n-1} . В случае несвязного графа G , содержащего k компонент, размерность \mathcal{B} равна $n - k$.

Можно этот факт доказать и напрямую. Мы же докажем этот факт, равно как и аналогичное утверждение, касающееся фундаментального множества циклов, доказав, что ортогональное дополнение пространства \mathcal{B} разрезов совпадает с пространством \mathcal{C} циклов и наоборот.

Теорема 9.12. *Пространство циклов \mathcal{C} и пространство реберных разрезов \mathcal{B} любого графа связаны соотношениями*

$$\mathcal{C} = \mathcal{B}^\perp, \quad \mathcal{B} = \mathcal{C}^\perp.$$

Доказательство. Заметим, прежде всего, что любой цикл C графа G пересекается с любым реберным разрезом $\partial(X)$ по четному числу ребер. Действительно, любая вершина цикла C входит либо в подмножество X вершин, либо в подмножество $\bar{X} = V(G) \setminus X$ вершин. При движении по циклу C количество переходов из X в \bar{X} совпадает с количеством обратных переходов из \bar{X} в X . Следовательно, $|E(C) \cap \partial(X)|$ есть четное число. А это, в свою очередь, означает, что любой цикл ортогонален реберному разрезу $\partial(X)$, то есть что $\mathcal{C} \subseteq \mathcal{B}^\perp$.

Обратно, рассмотрим произвольное подмножество ребер $F \notin \mathcal{C}$. Согласно теореме Veblen, среди вершин, инцидентных этому подмножеству ребер, найдется хотя бы одна вершина x , инцидентная нечетному количеству ребер из F . Следовательно, подмножество F и реберный разрез $\partial(x)$ пересекаются по нечетному количеству ребер, а потому их скалярное произведение равно единице. Иными словами, в пространстве \mathcal{B} реберных разрезов обязательно найдется хотя бы один элемент — разрез $\partial(x)$, не ортогональный F . Поэтому любое подмножество $F \notin \mathcal{C}$ ортогональному дополнению \mathcal{B} не принадлежит. С учетом $\mathcal{C} \subseteq \mathcal{B}^\perp$ отсюда следует, что \mathcal{C} в точности совпадает с \mathcal{B}^\perp .

Для доказательства равенства $\mathcal{B} = \mathcal{C}^\perp$ теперь достаточно показать, что $\mathcal{B} = (\mathcal{B}^\perp)^\perp$. Мы уже доказали, что любой реберный разрез ортогонален циклу, то есть $\mathcal{B} \subseteq \mathcal{C}^\perp = (\mathcal{B}^\perp)^\perp$. Но так как

$$|\mathcal{B}| + |\mathcal{B}^\perp| = m = |\mathcal{C}| + |\mathcal{C}^\perp| = |\mathcal{B}^\perp| + |(\mathcal{B}^\perp)^\perp|,$$

то размерности \mathcal{B} и $(\mathcal{B}^\perp)^\perp$ совпадают, а значит, $\mathcal{B} = (\mathcal{B}^\perp)^\perp$. \square

Следствие 9.13. Для любого связного графа G , построенного на n вершинах и имеющего m ребер, размерности пространств \mathcal{B} и \mathcal{C} равны

$$|\mathcal{B}| = n - 1, \quad |\mathcal{C}| = m - n + 1, \quad (7)$$

а фундаментальные разрезы и фундаментальные циклы образуют в этих пространствах базис.

Решение. Действительно, следствием теоремы 9.12 является равенство

$$|\mathcal{B}| + |\mathcal{C}| = m.$$

Поэтому для доказательства равенств (7) достаточно в пространствах \mathcal{B} и \mathcal{C} найти $n-1$ и $m+n-1$ линейно независимых векторов. А мы такие векторы уже нашли: в пространстве \mathcal{B} таковыми являются фундаментальные разрезы, а в пространстве \mathcal{C} — фундаментальные циклы. \square

10 Гамильтоновы циклы

10.1. Во многих практических задачах наряду с эйлеровыми циклами часто встречаются и так называемые *гамильтоновы циклы* — простые циклы, проходящие через каждую вершину графа.

10.1.1. Пожалуй, наиболее известная из таких задач — это так называемая *задача о коммивояжере*. В этой задаче торговец должен обойти все города из некоторого списка, заходя в каждый город только один раз, и вернуться в исходный город, с которого он начал свое путешествие. Обычно при этом указывается некоторый критерий оптимальности маршрута (кратчайший, самый дешевый и прочее). В случае, когда дополнительные критерии не указаны, задача сводится к поиску гамильтонова цикла в графе, вершинами которого являются города, а ребрами — соединяющие их дороги.

Еще одна задача, собственно, и дала имя гамильтонову циклу — она была описана в письме Гамильтона своему другу в форме математической игры на додекаэдре (рис.46). В этой игре один из игроков должен был вставить палочки в любые пять последовательно идущих вершин додекаэдра, а второй — продолжить этот путь на все оставшиеся вершины. Иными словами, игроки должны были найти простой цикл, проходящий через все вершины додекаэдра. Одно из возможных решений показано на рис.46

10.1.2. Как и в случае эйлерова цикла, первый вопрос, который возникает при анализе подобного рода задач, это вопрос существования гамильтонова цикла в заданном графе. Очевидными необходимыми условиями существования гамильтонова цикла в графе G является связность этого графа, а также отсутствие в нем вершин степени $\deg(x) = 1$, то есть листьев. Чуть менее тривиальным является следующее необходимое условие существования гамильтонова цикла в графе.

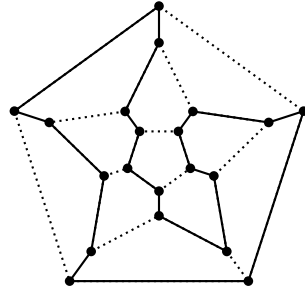


Рис. 46: Гамильтонов цикл в додекаэдре

Утверждение 10.1. Пусть в графе G имеется гамильтонов цикл. Тогда количество $c(G - S)$ компонент связности, получающихся в результате удаления вершин из некоторого непустого подмножества $S \subset V(G)$ вершин графа, не превосходит количества удаленных вершин:

$$c(G - S) \leq |S|. \quad (8)$$

Доказательство. Действительно, начнем обход гамильтонова цикла в графе из произвольной вершины x подмножества S . Входя в первую компоненту связности U_1 графа $G - S$ через вершину $s_1 \in S$, мы обязаны выйти из нее через какую-то другую вершину $s_2 \in S$. Иными словами, количество вершин подмножества S , через которые гамильтонов цикл входит в компоненты связности U_i , совпадает с количеством этих компонент связности. Следовательно, подмножество S обязано содержать как минимум $c(G - S)$ вершин. \square

10.1.3. Основная проблема со всеми известными на сегодняшний момент необходимыми условиями существования гамильтонова цикла в графе состоит в том, что ни одно из них не является одновременно и достаточным.

Конечно же, существуют графы, для которых вопрос о существовании гамильтонова цикла очевиден. Так, в простом связном графе на $n = 2$ вершинах гамильтонов цикл не существует (хотя существует *гамильтонов путь* — простой путь, проходящий через все вершины графа). В любом циклическом графе (или многоугольнике, или полигоне), построенном на $n > 2$ вершинах, существуют ровно один гамильтонов цикл. В случае полного графа K_n , $n > 2$, существует $(n - 1)!/2$ гамильтоновых циклов (смотри упражнение ??).

В общем же случае ответ на вопрос, существует ли в данном графе гамильтонов цикл, совершенно нетривиален. В частности, на настоящий момент нет никаких простых критериев существования гамильтонова цикла, подобных тому, что мы сформулировали выше для, казалось бы, очень похожего понятия эйлерова цикла. Более того, в 1972 году Ричард Карп доказал, что задача определения того, существует ли в произвольном графе гамильтонов цикл, является NP -полной задачей.

Несмотря на это печальное обстоятельство, в теории графов все же имеется целый ряд достаточных условий существования гамильтонова пути или цикла в графе. Так, интуитивно понятно, что чем больше у (простого связного) графа, построенного на n вершинах, ребер, тем больше вероятность того, что в нем существует гамильтонов цикл. Это видно хотя бы из того, что в полном графе K_n , количество ребер в котором максимально, гамильтонов цикл гарантированно существует. Ниже мы сформулируем несколько результатов, которые формализуют это интуитивное наблюдение.

10.1.4. Предположим вначале, что мы смогли каким-то образом построить в простом графе G гамильтонов путь $P = x_1, \dots, x_n$, $n > 2$. Возникает вопрос, когда мы этот путь можем

достроить до гамильтонова цикла C . Ответ очевиден в том случае, когда концы построенного пути — вершины x_1 и x_n — оказываются смежными. В этом случае мы всегда можем достроить путь P до гамильтонова цикла C , добавив к нему ребро $\{x_1, x_n\}$. Случай несмежных вершин x_1 и x_n является уже не столь очевидным.

Лемма 10.2. Пусть в простом графе G имеется гамильтонов путь $P = x_1, \dots, x_n$, $n > 2$, соединяющий пару несмежных вершин x_1 и x_n . Достаточным условием существования гамильтонова цикла в таком графе является выполнение следующего неравенства:

$$\deg(x_1) + \deg(x_n) \geq n. \quad (9)$$

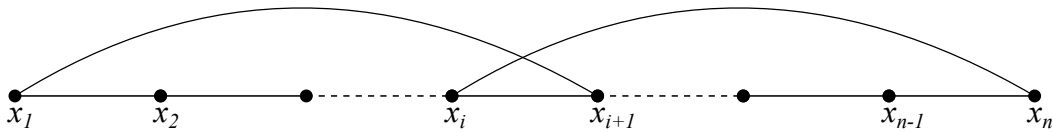


Рис. 47: Построение гамильтонова цикла в графе

Доказательство. Основная идея доказательства этого утверждения довольно проста. Пусть в графе G существует гамильтонов путь

$$P = (x_1, x_2, \dots, x_n),$$

соединяющий несмежные между собой вершины x_1 и x_n . Нам нужно показать, что в G обязательно существует принадлежащее этому пути ребро $e = \{x_i, x_{i+1}\}$, такое, что вершина x_{i+1} смежна с x_1 , а вершина x_i смежна с x_n (рис.47). Тогда мы всегда сможем заменить путь P на гамильтонов цикл

$$C = (x_1, x_2, \dots, x_i, x_n, x_{n-1}, \dots, x_{i+1}, x_1).$$

Покажем, что такое ребро e обязательно найдется. Пусть степень вершины x_1 равна l . Условие (9) означает, что среди любых l отличных от x_n вершин графа G найдется хотя бы одна вершина, смежная с x_n . Действительно, если бы это было не так, то степень вершины x_n была бы меньше или равна $(n - l - 1)$: l не смежных с x_n вершин плюс сама вершина x_n вкладает в $\deg(x_n)$ не дают. Это, в свою очередь, противоречит неравенству (9).

Теперь возьмем l смежных с x_1 вершин графа G и отступим от этих вершин на одно ребро назад вдоль гамильтонова пути P . Получим набор из l вершин, одна из которых — какая-то вершина x_i — обязательно смежна с x_n . Тогда ребро $\{x_i, x_{i+1}\}$ и будет нужным нам ребром e . \square

Следствие 10.3. Пусть $P = x_1, \dots, x_k$, $k > 2$, есть наибольший по включению простой путь в графе G . Тогда этот путь можно превратить в простой цикл C либо в случае, когда концы пути P — вершины x_1 и x_k — являются смежными, либо в случае, когда сумма степеней этих вершин больше или равна k :

$$\deg(x_1) + \deg(x_k) \geq k. \quad (10)$$

Доказательство. Действительно, в случае, когда вершины x_1 и x_k являются смежными, искомым цикл C получается добавлением к пути P ребра $\{x_1, x_k\}$.

Предположим, что вершины x_1 и x_k оказались несмежными. Заметим, что эти вершины соединены ребрами только с какими-то другими вершинами того же пути — в противном случае мы смогли бы продолжить путь P на какие-то другие вершины.

Рассмотрим подграф H , индуцируемый всеми вершинами пути P . Согласно сделанному выше замечанию, степени вершин x_1 и x_k в этом подграфе останутся прежними. Как следствие, неравенство (10) для этих вершин окажется верным и в подграфе H . Но в этом подграфе путь P является гамильтоновым. Следовательно, его можно превратить в гамильтонов цикл C способом, описанным при доказательстве леммы 10.2. \square

10.1.5. Следующая теорема дает нам достаточные условия существования гамильтонова пути в графе.

Теорема 10.4 (Оре). Пусть G — простой граф, построенный на $n > 2$ вершинах. Если для любых двух несмежных вершин x, y графа G выполняется условие

$$\deg(x) + \deg(y) \geq n - 1, \quad (11)$$

то граф G имеет гамильтонов путь.

Доказательство. Заметим, прежде всего, что любой граф G , удовлетворяющий условиям (11), является связным. Действительно, рассмотрим произвольную пару несмежных вершин x и y . Условие (11) вместе с принципом Дирихле гарантирует нам, что среди оставшихся $(n - 2)$ -х вершин обязательно найдется хотя бы одна вершина z , смежная как с x , так и с y . Следовательно, любую пару несмежных вершин в таком графе мы можем соединить путем длины два, то есть граф G связан.

Теперь предположим, что у нас выполнено условие (11), а гамильтонов путь в графе G не существует. Это означает, что максимальный простой путь P в таком графе содержит $k < n$ вершин. Кроме того, в любом связном графе, построенном на $n > 2$ вершинах, длина простого пути максимальной длины больше или равна двум. Поэтому, согласно следствию 10.3, в графе существует простой цикл длины k .

Так как $k < n$ и граф G связан, то в G обязана существовать вершина y , не входящая в цикл C и смежная хотя бы с одной из вершин $x \equiv x_1$ этого цикла $C = (x_1, \dots, x_k)$. Но в таком случае мы всегда можем построить в графе G простой путь y, x_1, x_2, \dots, x_k длины, большей, чем k , что противоречит предположению о том, что P есть путь максимальной длины. Полученное противоречие доказывает теорему. \square

10.1.6. Сформулируем теперь несколько очевидных следствий из теоремы Оре.

Следствие 10.5. Пусть G — граф, построенный на $n > 2$ вершинах. Если для любой пары несмежных вершин выполняется условие

$$\deg(x) + \deg(y) \geq n, \quad (12)$$

то в графе G имеется гамильтонов цикл.

Доказательство. Действительно, в силу теоремы Оре, в таком графе обязан существовать гамильтонов путь $P = x_1, \dots, x_n$. Согласно следствию 10.3, его всегда можно превратить в гамильтонов цикл. \square

Следствие 10.6 (Дирак). Пусть G — простой граф на $n > 2$ вершинах. Если степень каждой из его вершин больше или равна $(n - 1)/2$, то в графе существует гамильтонов путь, а если больше или равна $n/2$ — то в нем существует и гамильтонов цикл.

10.2. Исторически первым достаточным условием существования гамильтонова цикла в графе была теорема Дирака, доказанная им в 1952 году. Эта теорема явилась отправной точкой для получения целого ряда все более и более слабых условий на степени вершин графа, достаточных для существования гамильтонового цикла в нем. Теорема Оре, доказанная в 1960 году, была одним из наиболее важных результатов на этом пути. В 1972 году Вацлав Хватал получил достаточное условие, охватывающее все полученные ранее результаты. Для того, чтобы его сформулировать, нам понадобятся некоторые дополнительные понятия и факты.

10.2.1. Прежде всего, докажем следующее несложное следствие теоремы Оре.

Лемма 10.7. Пусть G есть простой граф, в котором существует пара несмежных между собой вершин x, y , суммарная степень которых больше или равна n . Тогда в G существует гамильтонов цикл тогда и только тогда, когда он существует в графе $G + \{x, y\}$.

Доказательство. В одну сторону утверждение очевидно — если в графе G существует гамильтонов цикл, то уж тем более он существует и в графе $G + \{x, y\}$. Обратное, предположим, что в $G + \{x, y\}$ существует гамильтонов цикл. Тогда в графе G существует гамильтонов путь с концами в вершинах x и y , который мы всегда можем трансформировать в гамильтонов цикл способом, описанным при доказательстве леммы 10.2. \square

10.2.2. Теперь сформулируем следующее полезное понятие.

Определение 10.8. Замыканием $C(G)$ графа G называется граф, полученный из G последовательным соединением в нем ребрами пар несмежных между собой вершин, суммарные степени которых больше или равны n , до тех пор, пока ни одной такой пары в графе не останется.

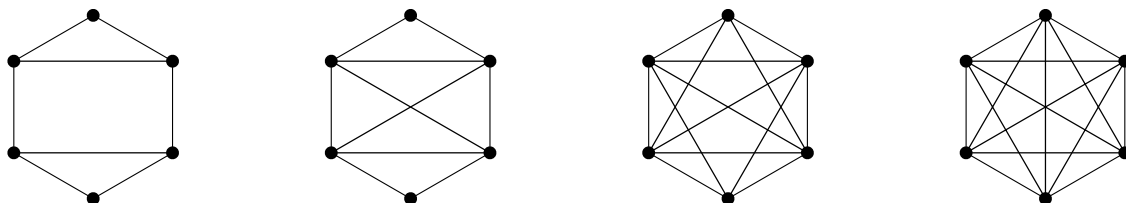


Рис. 48: Построение замыкания графа G

В качестве примера на рис.48 показано замыкание графа G , представляющее собой полный граф K_6 .

Замечание 10.9. Граф $C(G)$, полученный в результате процедуры замыкания графа G , не зависит от порядка выбора ребер, соединяющих несмежные вершины в графе G .

Доказательство. Обозначим через e_1, \dots, e_r и f_1, \dots, f_s последовательности ребер, добавляемых в процессе получения замыканий G_1 и G_2 графа G . Покажем, что любое ребро e_i присутствует в графе G_2 , а каждое ребро f_j появляется в графе G_1 . Действительно, предположим обратное. Пусть тогда $e_k = \{x, y\}$ есть первое ребро в последовательности e_i , которое не вошло в граф G_2 . Рассмотрим тогда граф $H = G + \{e_1, \dots, e_{k-1}\}$. Из определения графа G_1 следует, что

$$\deg_H(x) + \deg_H(y) \geq n.$$

Но мы так как все $e_i, i < k$ принадлежат G_2 , то граф H является также и подграфом графа G_2 , а потому

$$\deg_{G_2}(x) + \deg_{G_2}(y) \geq n.$$

Кроме того, так как в G_2 ребро $\{x, y\}$ отсутствует, то в графе G_2 вершины x и y несмежны. А это, в свою очередь, противоречит алгоритму построения графа G_2 . Следовательно, $G_1 = G_2$, так что $C(G)$ определен однозначно.

10.2.3. Следствием сформулированной выше леммы 10.7 является следующее утверждение.

Теорема 10.10 (Bondy-Chvatal, 1976). *Простой граф G является гамильтоновым тогда и только тогда, когда его замыкание $C(G)$ является гамильтоновым графом.*

Следствие 10.11. *Если $C(G) = K_n$, то граф G является гамильтоновым.*

Так, в случае $\delta(G) \geq n/2$ каждая пара несмежных вершин графа G должна быть соединена ребром, так что $C(G) = K_n$. Поэтому теорема Дирака есть частный случай теоремы 10.10.

10.2.4. Теперь мы готовы доказать наилучшую из известных оценок на степени вершин, гарантирующую существование гамильтонова цикла в графе.

Теорема 10.12 (Chvatal, 1972). *Пусть G есть простой граф, построенный на $n > 2$ вершинах, степенная последовательность (d_1, \dots, d_n) которого удовлетворяет следующему условию:*

$$\forall i < n/2 \quad \text{либо } d_i > i, \text{ либо } d_{n-i} \geq n - i. \quad (13)$$

Тогда в G существует гамильтонов цикл.

Доказательство. Мы покажем, что замыкание $C(G)$ графа G является полным графом K_n . Будем доказывать это от противного. Именно, предположим, что $C(G) \neq K_n$. Выберем тогда в графе $C(G)$ пару таких несмежных между собой вершин x, y , для которых сумма s степеней максимальна. Мы знаем, что эта сумма ограничена сверху величиной n :

$$s = \deg(x) + \deg(y) < n.$$

Положим для определенности $i := \deg(x) \leq \deg(y)$. В силу последнего неравенства, $i < n/2$, а $\deg(y) < (n - i)$.

Мы выбрали несмежную пару вершин с наибольшей суммой s степеней вершин. Поэтому любая несмежная с y вершина имеет степень, меньшую или равную $\deg(x) = i$. Таких вершин в графе $C(G) - y$ имеется $n - 1 - \deg(y)$ штук, причем из неравенства $s \leq (n - 1)$ следует, что количество этих вершин больше или равно i :

$$\deg(x) + \deg(y) \leq n - 1 \quad \implies \quad n - 1 - \deg(y) \geq \deg(x) = i.$$

Иными словами, в графе $C(G)$ гарантированно имеется i вершин, степени которых меньше или равны i .

Далее, любая несмежная с x вершина имеет степень, меньшую или равную $\deg(y) < (n - i)$. Таких вершин имеется $n - 1 - \deg(x) = n - 1 - i$ штук. Кроме того, сама вершина x имеет степень, меньшую $(n - i)$. Поэтому мы можем добавить эту вершину к $(n - 1 - i)$ вершинам графа, степень которых меньше $(n - i)$. Иными словами, мы получили, что количество вершин графа $C(G)$, степень которых меньше $(n - i)$, равно $(n - i)$.

Теперь заметим, что G есть остовный подграф графа G . Поэтому если в $C(G)$ какая-то вершина имеет степень, меньшую некоторого числа, то и в G степень этой вершины не превосходит того же числа. Следовательно, в графе G нашлись как минимум i вершин степени, меньшей или равной i , и $n - i$ вершин, степень которых меньше $n - i$, а это противоречит условиям (13). \square

10.3. В заключение данного параграфа поговорим о гамильтоновых циклах в орграфах.

10.3.1. Оказывается, что любое обобщение описанных выше достаточных условий существования гамильтоновых путей и циклов в неориентированных графах на ориентированный случай дается довольно тяжело. В частности, существует следующее обобщение теоремы Дирака на случай орграфов.

Теорема 10.13. Пусть D есть сильно связный орграф, исходящая и входящая степени любой вершины x в котором больше или равна $n/2$. Тогда в D существует гамильтонов цикл.

Доказательство этой теоремы значительно сложнее доказательства теоремы Дирака для неориентированных графов и здесь приводиться не будет.

10.3.2. Вместо этого мы рассмотрим важный подкласс орграфов, для которых вопрос существования гамильтонова пути или цикла решается значительно проще.

Определение 10.14. Турниром T называется орграф, любые две вершины которого соединены ровно одним ориентированным ребром. Иными словами, турниром называется орграф, полученный из полного графа K_n произвольной ориентацией его ребер.

Теорема 10.15 (Redei, 1934). В любом турнире T существует ориентированный гамильтонов путь.

Доказательство проведем индукцией по количеству n вершин в турнире D . Очевидно, что тривиальный гамильтонов путь, состоящий из одной вершины, существует в тривиальном же турнире, построенном на одной вершине. Предположим теперь, что при любом фиксированном $n \geq 2$ любой турнир, построенный на $(n - 1)$ -й вершине, имеет гамильтонов путь. Рассмотрим тогда в турнире T , построенном на n вершинах, произвольную вершину x . Орграф $T' = T - x$ является турниром на $(n - 1)$ -й вершине, так что в нем существует гамильтонов путь $P' = (x_1, \dots, x_{n-1})$. В этом случае возможно всего три варианта.

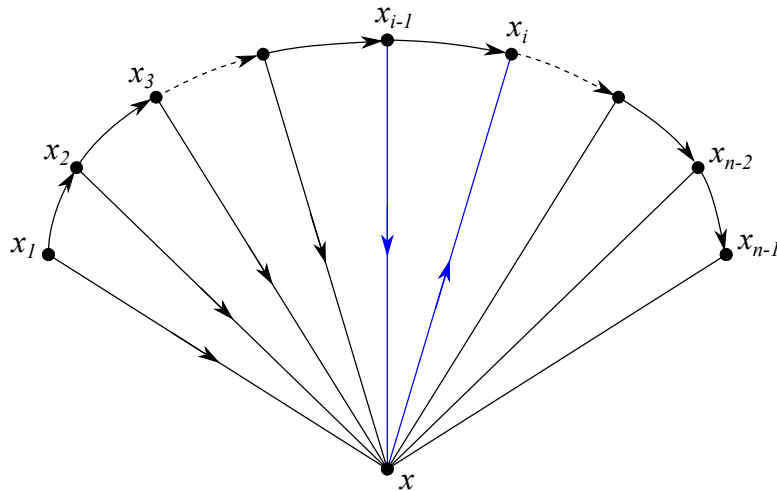


Рис. 49

1. В орграфе T существует ребро (x, x_1) . Тогда в T существует и ориентированный гамильтонов путь (x, x_1, \dots, x_{n-1}) .
2. В орграфе T существует ребро (x_1, x) , и кроме того, в T существует ребро (x, x_i) для некоторого $i > 1$ (см.рис.49). Тогда в T имеется гамильтонов путь, а именно, путь

$$(x_1, \dots, x_{i-1}, x, x_i, \dots, x_{n-1}).$$

3. Наконец, в орграфе T не существует ребра (x, x_i) ни для какого $i = 1, \dots, n - 1$. Но тогда в T существует ребро (x_{n-1}, x) , а значит, и гамильтонов путь (x_1, \dots, x_{n-1}, x) .

11 Граф де Брейна

11.1. Как мы уже упоминали, эйлеровы и гамильтоновы графы встречаются в самых разнообразных практических задачах. В качестве очень красивого и полезного примера остановимся на одной из таких задач — задаче о последовательностях де Брейна.

11.1.1. Формальная постановка задачи такова: найти наименьшую циклическую последовательность (циклическое слово) над алфавитом из n букв, содержащую все возможные подстроки длины k (так называемые k -меры).

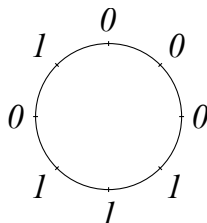


Рис. 50: Циклическая последовательность $B(2, 3)$

Пример 11.1. Рассмотрим циклические последовательности над алфавитом из двух букв — чисел 0 и 1, содержащие все возможные подпоследовательности длины три (тримеры). Таковых, как мы знаем, существует $2^3 = 8$ штук — восемь первых чисел в двоичной системе исчисления:

000, 001, 010, 011, 100, 101, 110, 111.

Расставляя их по кругу, мы, конечно же, получаем циклическую последовательность, состоящую из $k \cdot n^k = 3 \cdot 2^3 = 24$ символов и содержащую все возможные тримеры. Однако такое слово минимальным не будет — несложно видеть, что изображенная на рис.50 циклическая последовательность длины $2^3 = 8$ содержит все тримеры, причем каждый из них она содержит по одному разу. Понятно, что циклическую последовательность длины меньшей, чем 2^3 , построить невозможно — она не сможет содержать все 8 тримеров. Поэтому показанная на рис.50 циклическая последовательность минимальна.

Проведенные в примере 11.1 рассуждения позволяют предположить, что и в общем случае для произвольных n и k минимальная циклическая последовательность над алфавитом из n букв, содержащая все n^k возможных k -меров, также имеет длину n^k . В этой связи возникают сразу три вопроса — как доказать это предположение, как понять, сколько таких минимальных строк существует, ну и наконец, как их всех найти.

На все эти вопросы дал ответ Николас де Брейн в своей работе [?] 1946 года. Во-первых, он доказал, что действительно для произвольных n и k существуют циклические последовательности длины n^k над алфавитом из n букв, содержащие все возможные k -меры — в его честь такие циклические последовательности называют теперь *последовательностями де Брейна $B(n, k)$* порядка k . Во-вторых, он подсчитал количество таких последовательностей. Наконец, он указал конструктивный алгоритм построения этих последовательностей. И сделал он это, построив для заданных n и k некоторый орграф специального вида.

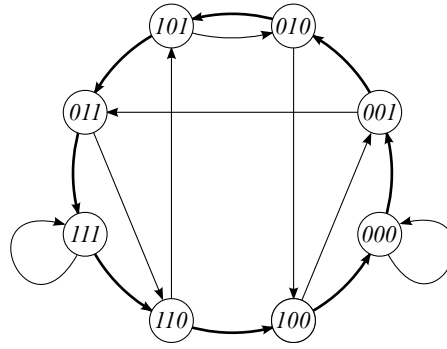


Рис. 51: Построение циклической последовательности с помощью гамильтонова цикла в графе

11.1.2. Итак, попытаемся, вслед за де Брейном, сопоставить множеству всевозможных k -меров над алфавитом из n букв некоторый оргграф D , обход которого даст нам какую-то циклическую последовательность $B(n, k)$ длины n^k . Как правило, первое, что приходит в голову при анализе этой задачи, это взять в качестве вершин будущего графа все k -меры, и соединить любые две вершины направленным ребром в том случае, если $(k - 1)$ -мер, отвечающий суффиксу первой вершины, совпадает с $(k - 1)$ -мером, соответствующим префиксу второй.

Так, на рис.51 показан оргграф, построенный с помощью этого алгоритма для разобранный выше примера 11.1. В этом оргграфе, к примеру, вершина 011 соединена с вершиной 110 ребром потому, что суффикс первой из них — 2-мер 11 — совпадает с префиксом второй.

Заметим теперь, что обход этого оргграфа по простому циклу, содержащему все вершины (помеченный жирными стрелками на рис.51 цикл), позволяет построить одну из искомым нами циклических последовательностей $B(2, 3)$, а именно, последовательность 11101000. Тот же факт справедлив и в общем случае. Иными словами, при таком подходе задача поиска циклической последовательности $B(n, k)$ сводится к построению гамильтонова цикла в оргграфе D .

11.1.3. На первый взгляд кажется, что мы решили поставленную задачу — действительно, нам удалось формализовать задачу поиска последовательностей де Брейна на языке теории графов. Однако у такого решения имеется множество недостатков. Во-первых, априори совсем не очевидно, что гамильтонов цикл у любого такого графа существует. Во-вторых, даже если он и существует, то непонятно, как его там искать — как мы знаем, задача построения гамильтонова цикла весьма нетривиальна. Наконец, если такие циклы и существуют, то не ясно, как подсчитать их количество.

Основное достижение де Брейна состояло в том, что он предложил другой, далеко не столь очевидный, подход к формализации данной задачи. Как показал де Брейн, эту задачу можно изящно переформулировать, сведя ее к сравнительно легко решаемой задаче построения эйлерова цикла в оргграфе специального вида — графе де Брейна.

Алгоритм построения графа де Брейна следующий: возьмем в качестве вершин оргграфа вместо k -меров все возможные $(k - 1)$ -меры (их, очевидно, $n^{(k-1)}$ штук) и свяжем любые две из них ориентированным ребром в случае, если существует k -мер, префиксом которого является первая вершина, а суффиксом — вторая.

На рис.52 показан граф де Брейна, построенный для данных из примера 11.1. Так как существуют тримеры 010 и 011, то вершина 01 соединена в этом графе с вершинами 10 и 11. Двум входящим в эту вершину ребрам отвечают тримеры 101 и 001. Так как к любому 2-меру можно двумя способами слева приписать единицу или ноль, и справа приписать единицу и ноль,

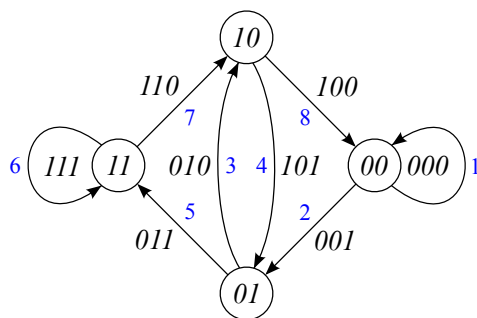


Рис. 52: Граф де Брейна для $n = 2$, $k = 3$

то аналогичный факт справедлив и для любой другой вершины — в любую вершину входят ровно $n = 2$ ребра и выходят из этой вершины также ровно $n = 2$ ребра. Как следствие, в таком орграфе обязательно существует эйлеров цикл. Обход графа по эйлерову циклу 000, 001, 010, 101, 011, 111, 110, 100 позволяет восстановить искомую циклическую последовательность $B(2, 3) = 00010111$.

Аналогичные рассуждения проходят и в общем случае. Следовательно, любой граф де Брейна является эйлеровым, а любой эйлеров цикл в нем отвечает некоторой последовательности де Брейна $B(n, k)$.

Итак, построив для любых n и k граф де Брейна, мы доказали существование соответствующей последовательности де Брейна. Далее, используя алгоритмы поиска эйлеровых циклов в орграфе, мы можем находить такие последовательности. Осталось понять, сколько всего таких последовательностей существует. Де Брейн ответил и на этот вопрос, доказав, что их количество равно $(n!)^{n^{k-1}}/n^k$. Два различных доказательства этого результата — комбинаторное и алгебраическое — можно найти, например, в учебнике [?].

11.1.4. Как это часто бывает в прикладной математике, задача, очень похожая на рассмотренную выше, возникла относительно недавно еще в одной, достаточно молодой области прикладной математики — в биоинформатике [?]. Одной из наиболее актуальных задач в этой науке является задача ассемблирования (сборки) геномов из так называемых ридов (reads) — относительно коротких (содержащих порядка 100 символов — нуклеотидов) строк над 4-буквенным алфавитом $\{A, C, G, T\}$, получаемых в результате секвенирования (разделения) генома (а точнее, очень большого количества одинаковых геномов). Первые методы сборки генома из таких ридов как раз и базировались на построении графа, вершинам которого сопоставлялись риды, а ребрам — перекрытия между этими ридами фиксированной длины. При этом исходный геном восстанавливался с помощью построения гамильтонова цикла в подобном графе.

Рассмотрим в качестве простейшего примера очень короткий циклический геном, показанный на рис.53,а. Предположим, что в результате секвенирования мы получили из него пять ридов CGTGCAA, ATGGCGT, CAATGGC, GGCGTGC и TGCAATG длины 7. Соответствующий этой последовательности ридов граф показан на рис.53,б. Каждому из пяти ридов поставлена в соответствие одна из вершин этого графа. Две вершины соединяются ребром в случае, если ширина перекрытия соответствующих ридов составляет пять нуклеотидов (см. рис.53,б). Проход по гамильтонову циклу

$$ATGGCGT \rightarrow GGCGTGC \rightarrow CGTGCAA \rightarrow TGCAATG \rightarrow CAATGGC \rightarrow ATGGCGT$$

позволяет путем объединения первых двух нуклеотидов в каждом риде восстановить исходный геном ATGGCGTGCA.

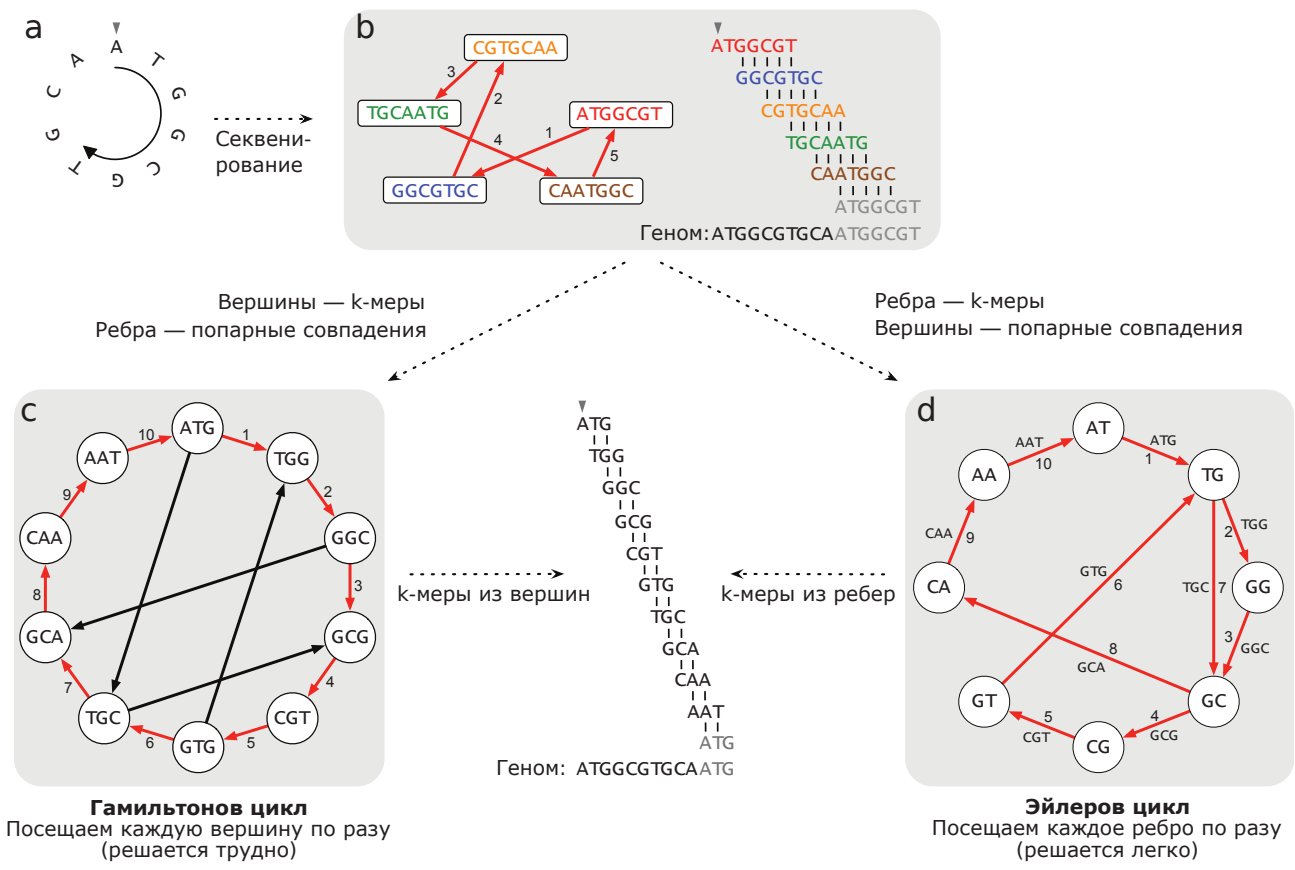


Рис. 53: Сборка простейшего генома

Более современные методы сборки геномов обычно работают со строками определенной длины k (которые как раз и называются k -мерами), значительно более короткими, нежели исходные риды. Например, типичный 100-нуклеотидный рид разбивается вначале на 55-меры, длина перекрывающихся участков которых равна сорока шести. В нашем модельном примере каждый 7-нуклеотидный рид разбивается на пять 3-меров, перекрывающихся между собой по двум нуклеотидам (см. рис.53,с,d). Даже для этого примера найти соответствующий исходному геному гамильтонов цикл нелегко. В реальной же ситуации из одного генома в процессе секвенирования получают миллионы (10^6) ридов, триллионы (10^{12}) k -меров, то есть графы с огромным количеством вершин. Задача поиска гамильтонова цикла в таком графе практически нерешаема.

Павел Певзнер в 1989 году предложил для таких случаев использовать подход де Брейна. Переход от задачи поиска гамильтонова цикла в графе (рис.53,с) к задаче поиска эйлерова цикла (рис.53,d) существенным образом ускорил процесс асемблирования геномов и стал общепринятым в большинстве современных асемблеров, предназначенных для сборки генома из коротких ридов.

12 Циркуляции и напряжения. Электрические сети

12.1. В предыдущем параграфе мы показали, что четные подграфы и реберные разрезы неориентированного графа G образуют линейные ортогональные друг другу подпространства про-

пространства \mathcal{E} ребер графа G над двухэлементным полем \mathbb{F}_2 — пространство циклов \mathcal{C} и пространство разрезов \mathcal{B} . В данном параграфе мы введем аналоги этих конструкций для ориентированных графов.

12.1.1. Заметим, прежде всего, что наряду с обычными орграфами мы можем рассматривать и так называемые взвешенные орграфы.

Определение 12.1. *Взвешенным орграфом (D, f) называется орграф D вместе с функцией $f: E(D) \rightarrow \mathbb{F}$, заданной на множестве ребер орграфа D и принимающей значения из некоторого поля \mathbb{F} . Значение $f(e)$ этой функции на ребре e называется *весом* ребра e , а сумма значений этой функции для ребер из некоторого подграфа H орграфа D называется *весом* соответствующего подграфа H .*

Взвешенные графы встречаются в большом количестве важных с практической точки зрения задач. Вес ребра в таких задачах позволяет, например, моделировать стоимость транспортировки того или иного продукта вдоль ребра, величину потока вещества вдоль ребра, или какие-то другие параметры сети, математической моделью которой является ориентированный граф D .

Замечание 12.2. Далее для простоты мы будем считать, что поле \mathbb{F} совпадает с полем \mathbb{R} вещественных чисел. Ситуации, в которых \mathbb{F} отлично от \mathbb{R} , будут оговариваться отдельно.

Часто удобно рассматривать функцию $f: E(D) \rightarrow \mathbb{R}$ как вектор \mathbf{f} , координаты которого помечены ребрами орграфа D . Так как такие векторы мы можем складывать и умножать на вещественные числа, то множество этих векторов образует для любого орграфа D линейное пространство \mathbb{R}^m , $m = |E(D)|$.

12.1.2. Довольно часто на практике имеет смысл накладывать дополнительные ограничения на функцию f взвешенного орграфа (D, f) . Так, в случае моделирования процесса переноса вещества с помощью взвешенных орграфов функция f должна удовлетворять так называемому закону сохранения вещества, согласно которому количество вещества, поступающего в заданную вершину x орграфа D , должно совпадать с количеством вещества, из этой вершины исходящего. Если эти условия выполняются в каждой вершине орграфа, то соответствующая функция f носит название циркуляции в орграфе D .

Определение 12.3. Функция f , заданная на ребрах орграфа D , называется циркуляцией в орграфе D , если в каждой вершине x соответствующего взвешенного орграфа (D, f) выполняются следующие законы сохранения:

$$f^+(x) = f^-(x) \quad \text{для любой вершины } x \in V(D). \quad (14)$$

Здесь $f^+(x)$ есть сумма весов ребер орграфа, входящих в вершину x , а $f^-(x)$ — сумма весов исходящих из x ребер.

Пример 12.4. Рассмотрим взвешенный орграф (D, f) , показанный на рис.54,а. Синими цифрами на этом рисунке показаны веса $f(e)$ соответствующих ориентированных ребер $e \in E(D)$. Видно, что в каждой вершине x выполняется закон сохранения (14). Следовательно, отвечающий этим весам вектор \mathbf{f} представляет собой циркуляцию в орграфе D .

Условия (14) можно переписать в векторном виде, используя матрицу \mathbf{M}_i инцидентности орграфа D :

$$\mathbf{M}_i \cdot \mathbf{f} = \mathbf{0}. \quad (15)$$

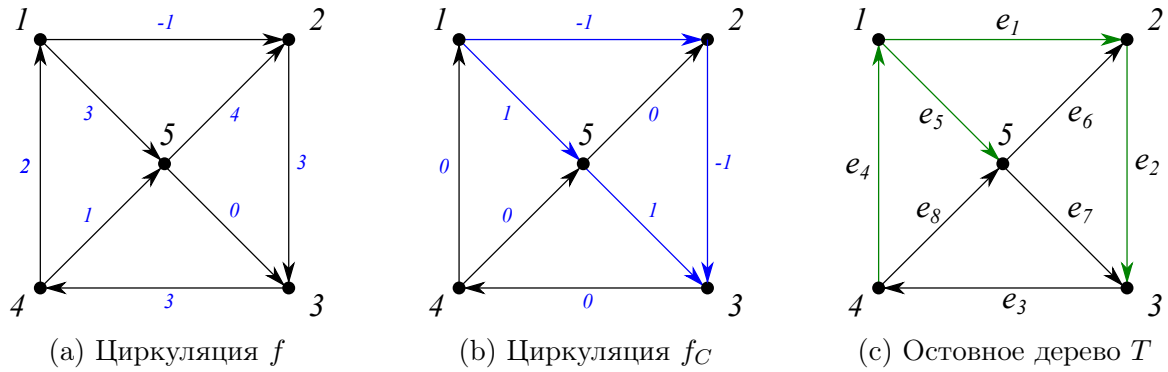


Рис. 54

Действительно, произведение любой строки матрицы \mathbf{M}_i на столбец \mathbf{f} даст нам сумму $f(e_i)$ по всем ребрам, инцидентным соответствующей данной строке вершине x орграфа D . При этом значения f на ребрах, входящих в x , войдут в эту сумму со знаком минус, а значения f на ребрах, исходящих из x — со знаком плюс.

Из равенства (15), в частности, следует, что множество всех циркуляций в орграфе D образует линейное подпространство — пространство векторов, ортогональных пространству строк матрицы инцидентности \mathbf{M}_i . Это подпространство называется пространством циркуляций \mathcal{C} орграфа D .

12.1.3. Рассмотрим наряду с ориентированным графом D соответствующий ему неориентированный (underlying) граф G . Оказывается, с любым циклом C в графе G мы можем связать некоторую циркуляцию. Именно, зададим какое-то направление обхода цикла C и определим функцию f_C на множестве ребер орграфа D следующим образом:

$$f_C(e) = \begin{cases} 1, & \text{если направление вектора } e \text{ совпадает с направлением обхода цикла } C, \\ -1, & \text{если } e \text{ направлен в сторону, противоположную направлению обхода } C, \\ 0, & \text{если } e \text{ не принадлежит циклу } C. \end{cases} \quad (16)$$

Несложно убедиться в том, что функция f_C удовлетворяет закону сохранения (14). Действительно, количество входящих в любую вершину x неориентированных ребер цикла C обязано совпадать с количеством исходящих из нее неориентированных ребер цикла C . Поэтому в случае, когда каждое инцидентное x ориентированное ребро, принадлежащее циклу, совпадает с направлением обхода C , закон сохранения (14) с очевидностью выполняются. Если же направление какого-то инцидентного x ориентированного ребра не совпадает с направлением цикла, то мы, назначая функции f_C на таком ребре значение, равное -1 , подправляем сумму значений f на инцидентных x ребрах так, чтобы закон сохранения (14) выполнялся.

В качестве примера на рис.54,b показана циркуляция f_C , связанная с циклом, помеченным синим цветом на рисунке. Обход этого цикла C осуществляется против часовой стрелки.

12.1.4. Важно заметить, что описанные выше циркуляции f_C , связанные с циклами C , порождают все пространство \mathcal{C} циркуляций в орграфе D . Именно, справедливо следующее

Утверждение 12.5. *Любая циркуляция в орграфе D представляет собой линейную комбинацию циркуляций, связанных с циклами.*

Для доказательства этого утверждения нам понадобятся некоторые полезные дополнительные понятия и факты.

Определение 12.6. *Носителем функции f называется подмножество S элементов области определения f , на котором эта функция отлична от нуля:*

$$S = \{x \mid f(x) \neq 0\}.$$

Лемма 12.7. *Носитель S отличной от нуля циркуляции \mathbf{f} в орграфе D содержит хотя бы один цикл.*

Доказательство. Рассмотрим подграф H , индуцированный ребрами носителя S . Степень любой вершины x в этом подграфе больше или равна двум. Как следствие, подмножество S ребер содержит хотя бы один цикл C . \square

12.1.5. Приступим теперь к доказательству утверждения 12.5. Рассмотрим произвольную циркуляцию f с носителем S . Будем доказывать данное утверждение индукцией по $|S|$. Случай $S = \emptyset$ тривиален. Пусть теперь $|S| > 0$. Согласно лемме 12.7, подмножество S ребер содержит хотя бы один цикл C . Рассмотрим произвольное ребро $e \in C$, выберем направление обхода C , совпадающее с ориентацией ребра e , и построим на C циркуляцию \mathbf{f}_C вида (16). Тогда у циркуляции $\mathbf{f}' := \mathbf{f} - f(e) \cdot \mathbf{f}_C$ подмножество S' ребер, на которых f' отлична от нуля, будет собственным подмножеством множества S . По индукционному предположению, f' можно представить в виде линейной комбинации циркуляций, связанных с циклами в графе G . Следовательно, то же утверждение верно и для циркуляции

$$\mathbf{f} = \mathbf{f}' + f(e) \cdot \mathbf{f}_C.$$

12.1.6. Рассмотрим теперь произвольное остовное дерево T графа G (см.рис.54,с, на котором ребра остовного дерева T помечены зеленым цветом). С каждым таким деревом мы можем связать набор из $m - (n - 1)$ фундаментальных циклов C_{e_i} , $e_i \notin T$, каждый из которых получается добавлением к ребрам из T ребра e_i . В свою очередь, для каждого из этих циклов мы можем построить циркуляцию \mathbf{f}_{e_i} , связанную с этим циклом, полагая $\mathbf{f}_{e_i}(e_i) = 1$ и используя правила (16).

В качестве примера рассмотрим остовное дерево T , показанное на рис.54,с. С каждым из четырех ребер $e \notin E(T)$ мы можем связать фундаментальный цикл C_e . Соответствующие этим циклам циркуляции \mathbf{f}_{e_i} имеют следующий вид:

	e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8
$\mathbf{f}_{e_3}^T$	1	1	1	1	0	0	0	0
$\mathbf{f}_{e_6}^T$	-1	0	0	0	1	1	0	0
$\mathbf{f}_{e_7}^T$	-1	-1	0	0	1	0	1	0
$\mathbf{f}_{e_8}^T$	0	0	0	-1	-1	0	0	1

Заметим, что векторы \mathbf{f}_{e_i} линейно независимы. Действительно, i -я координата каждого из этих векторов равна единице, а координаты $j \neq i$, связанные с оставшимися векторами e_j , не принадлежащими $E(T)$, равны нулю. Как и в неориентированном случае, нам осталось показать, что набор этих линейно независимых векторов образует базис пространства \mathcal{C} . И как и в неориентированном случае, это проще сделать, рассмотрев двойственное к \mathcal{C} пространство — в данном случае, пространство \mathcal{B} строк матрицы \mathbf{M}_i .

12.1.7. По определению, любой элемент \mathbf{g} пространства \mathcal{B} представляет собой линейную комбинацию строк \mathbf{m}_i матрицы \mathbf{M}_i :

$$\mathbf{g} = \sum_{i=1}^n p(i) \cdot \mathbf{m}_i = \mathbf{p} \cdot \mathbf{M}_i, \quad n = |V(D)|. \quad (17)$$

Здесь $\mathbf{p} = (p(1), \dots, p(n))$ есть некоторый вектор-строка линейного пространства \mathbb{R}^n . По сути дела, вектор \mathbf{p} сопоставляет каждой вершине $i \in V(D)$ орграфа D некоторое вещественное число $p(i)$. Пусть $\mathbf{g}(e)$ есть компонента вектора \mathbf{g} , отвечающая ребру $e = (x, y)$, направленному от вершины x к вершине y . Так как любой столбец матрицы \mathbf{M}_1 содержит всего два ненулевых числа, а именно, 1 в строке, отвечающей вершине x , и -1 в строке, отвечающей вершине y , то

$$\mathbf{g}(e) = \mathbf{p}(x) - \mathbf{p}(y). \quad (18)$$

Выписанные соотношения допускают следующую интересную физическую трактовку. Вектор \mathbf{p} мы можем рассматривать как вектор потенциалов в электрической сети, моделируемой ориентированным графом D . Вектор \mathbf{g} мы можем тогда рассматривать как вектор разности этих потенциалов, то есть как вектор напряжений в сети. Из этих соображений пространство, натянутое на строки матрицы \mathbf{M}_1 , называется пространством \mathbf{V} напряжений, а элементы этого пространства называются напряжениями.

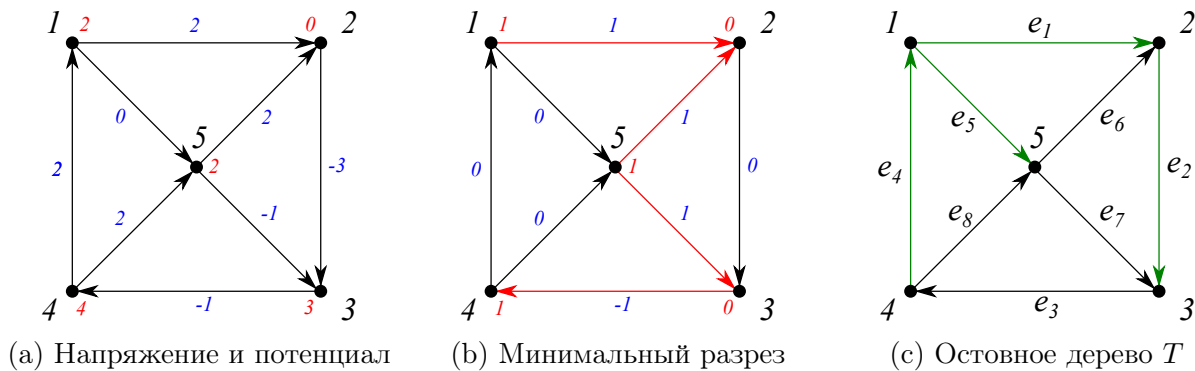


Рис. 55

На рис.55,а в качестве примера показан взвешенный оргграф (D, g) , веса ребер которого описываются координатами вектора \mathbf{g} напряжений (синие цифры на рисунке). Помеченные красным цветом числа обозначают значения функции $\mathbf{p}(x)$ на множестве вершин. Видно, что координаты векторов \mathbf{g} и \mathbf{p} связаны равенствами (18).

12.1.8. Мы уже знаем, что любой цикл графа G порождает в оргграфе D некоторую циркуляцию \mathbf{f}_C . Аналогично, любой минимальный реберный разрез $\partial(X) =: B$ графа G , связанный с подмножеством вершин X , порождает в оргграфе D напряжение

$$g_B(e) = \begin{cases} 1, & \text{если } e \in \partial^+(X), \text{ то есть если ребро } e \text{ исходит из } X, \\ -1, & \text{если } e \in \partial^-(X), \text{ то есть если ребро } e \text{ входит в } X, \\ 0, & \text{если } e \text{ не принадлежит разрезу } \partial(X). \end{cases} \quad (19)$$

Такому напряжению отвечает потенциал \mathbf{p}_B вида

$$p_B(x) = \begin{cases} 1, & \text{если } x \in X, \\ 0, & \text{если } x \notin X. \end{cases} \quad (20)$$

На рис.55,б в качестве примера показан минимальный реберный разрез $B = \partial(X)$, $X = \{1, 4, 5\}$, помеченный красными ребрами на рисунке. Значения отвечающего этому разрезу напряжения \mathbf{g}_B помечены синими числами, а значения потенциала — красными числами.

12.1.9. Довольно очевидным является тот факт, что все пространство \mathbf{V} порождается напряжениями \mathbf{g}_B , связанными с реберными разрезами.

Лемма 12.8. *Носитель S любого отличного от нуля напряжения \mathbf{g} в орграфе содержит хотя бы один минимальный реберный разрез $B = \partial(X)$.*

Доказательство. Пусть $\mathbf{g} = \mathbf{p} \cdot \mathbf{M}_i$ есть напряжение, для которого $S \neq \emptyset$. Выделим произвольное ребро $(x, y) \in S$ и в качестве X возьмем множество вершин, потенциалы которых совпадают с $p(x)$. По построению множества X , напряжение $g(e)$ на любом ребре e , принадлежащем разрезу $\partial(X)$, отлично от нуля. Так как S есть все множество ребер, на которых функция g отлична от нуля, то $\partial(X) \subseteq S$. Кроме того, $\partial(X) \neq \emptyset$ — это множество обязательно содержит ребро (x, y) . Поэтому $\partial(X)$ есть нетривиальный реберный разрез. Но такой разрез либо сам является минимальным, либо содержит хотя бы один минимальный реберный разрез B . Отсюда следует, что и S содержит B .

Утверждение 12.9. *Любое напряжение \mathbf{g} в орграфе представляет собой линейную комбинацию напряжений \mathbf{g}_B , связанных с минимальными реберными разрезами.*

Доказательство полностью аналогично доказательству утверждения 12.5. Именно, пусть \mathbf{g} есть произвольное напряжение, а S — подмножество ребер, на которых функция g отлична от нуля. Докажем утверждение индукцией по $|S|$. Утверждение тривиально в случае $S = \emptyset$. Предположим, что $|S| > 0$. Согласно лемме 12.8, S содержит минимальный реберный разрез $\partial(X) =: B$. Выберем произвольное ребро $e = (x, y) \in \partial(X)$, исходящее из X . В случае, если такое ребро отсутствует, то есть в случае, когда все ориентированные ребра D входят в X , выберем в качестве B реберный разрез $\partial(V(G) \setminus X)$, а в качестве e — произвольное ребро, входящее в X . Рассмотрим теперь напряжение $\mathbf{g}' := \mathbf{g} - g(e) \cdot \mathbf{g}_B$. На ребре e напряжение $\mathbf{g}'(e) = 0$, так что его носитель $S' \subset S$. Следовательно, к нему можно применить индукционное предположение, то есть представить его в виде линейной комбинации напряжений, связанных с минимальными реберными разрезами в G . Но тогда это же утверждение верно и для напряжения

$$\mathbf{g} := \mathbf{g}' + g(e) \cdot \mathbf{g}_B.$$

12.1.10. Как и в неориентированном случае, мы всегда в \mathcal{B} можем найти набор, состоящий из $n - 1$ линейно независимых векторов \mathbf{g}_i напряжений. Для этого нам достаточно рассмотреть остовное дерево T в графе G , выбрать в этом графе $n - 1$ фундаментальный разрез B_{e_i} , $e_i \in E(T)$, и построить соответствующие этим разрезам напряжения \mathbf{g}_{e_i} .

В качестве примера рассмотрим остовное дерево T , показанное на рис.55,с. С каждым из четырех ребер $e_i \in E(T)$ мы можем связать фундаментальный разрез B_{e_i} . Соответствующие этим разрезам напряжения \mathbf{g}_{e_i} записываются в следующем виде:

	e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8
\mathbf{g}_{e_1}	1	0	-1	0	0	1	1	0
\mathbf{g}_{e_2}	0	1	-1	0	0	0	1	0
\mathbf{g}_{e_4}	0	0	-1	1	0	0	0	1
\mathbf{g}_{e_5}	0	0	0	0	1	-1	-1	1

Как и векторы \mathbf{f}_{e_i} , векторы \mathbf{g}_{e_i} линейно независимы. Нам осталось показать, что эти векторы образуют базис пространства \mathcal{B} . Как и в случае неориентированных графов, это легче всего сделать, рассматривая одновременно пространства \mathcal{B} и \mathcal{C} .

12.1.11. Покажем, прежде всего, что любая пара векторов $\mathbf{g} \in \mathcal{B}$ и $\mathbf{f} \in \mathcal{C}$ взаимно ортогональна. Действительно, согласно определению (17) напряжения \mathbf{g} , любой такой вектор можно представить в виде $\mathbf{g} = \mathbf{p} \cdot \mathbf{M}_i$. Но, согласно (15), произведение $\mathbf{M}_i \cdot \mathbf{f} = \mathbf{0}$ для любого вектора $\mathbf{f} \in \mathcal{C}$.

Следовательно, скалярное произведение

$$\mathbf{g} \cdot \mathbf{f} = \mathbf{p} \cdot \mathbf{M}_i \cdot \mathbf{f} = \mathbf{p} \cdot \mathbf{0} = 0.$$

Таким образом, нами доказано следующее

Утверждение 12.10. *Пространства \mathcal{C} циркуляций и \mathcal{B} напряжений взаимно ортогональны. При этом их пересечение состоит из единственного вектора $\mathbf{0}$ — тривиального цикла и тривиального разреза.*

Следствие 12.11. *Размерности пространства \mathcal{B} напряжений и \mathcal{C} циркуляций орграфа D равны*

$$\dim(\mathcal{B}) = n - 1, \quad \dim(\mathcal{C}) = m - n + 1,$$

а соответствующие фундаментальным разрезам и фундаментальным циклам графа G напряжения и циркуляции орграфа D образуют базисы пространств \mathcal{B} и \mathcal{C} .

Доказательство. Действительно, мы в пространстве \mathcal{B} предъявили $n - 1$ линейно независимых векторов — напряжений, связанных с $(n - 1)$ -м фундаментальным разрезом орграфа D . Аналогично, в пространстве \mathcal{C} мы предъявили $m - (n - 1)$ линейно независимых векторов — циркуляций, связанных с $(m - n + 1)$ -м фундаментальным циклом в этом орграфе. Следовательно, размерность пространства \mathcal{B} больше или равна $n - 1$, а размерность пространства \mathcal{C} больше или равна $m - (n - 1)$. Но так как пространства \mathcal{B} и \mathcal{C} взаимно ортогональны, то сумма размерностей этих подпространств не может превосходить размерности m всего пространства строк длины m . Поэтому размерности подпространств \mathcal{B} и \mathcal{C} равны, соответственно, $n - 1$ и $m - (n - 1)$, а описанные выше векторы образуют базисы в этих пространствах. При этом само пространство векторов длины m представляет собой прямую сумму подпространств \mathcal{B} и \mathcal{C} (см.[?], теорема 3, стр.27).

12.1.12. Заметим, что описанные результаты легко переносятся на случай линейного пространства над произвольным полем \mathbb{F} . Рассмотрим, в частности, случай $\mathbb{F} = \mathbb{F}_2$, то есть случай поля характеристики два. В этом случае ориентация ребер графа роли не играет. Например, закон сохранения (15) в этом случае просто означает, что

$$\sum_{e \in \partial(x)} f(e) = 0 \quad \text{для любой вершины } x \in V(D),$$

то есть что количество входящих в любую вершину ребер совпадает с количеством исходящих из нее ребер. Иными словами, \mathbf{f} есть циркуляция над полем \mathbb{F}_2 тогда и только тогда, когда носителем \mathbf{f} является четный подграф соответствующего D неориентированного графа G . Аналогично, \mathbf{g} есть напряжение над полем \mathbb{F}_2 тогда и только тогда, когда носитель \mathbf{g} есть некоторый реберный разрез неориентированного графа G . Иными словами, в данном случае пространства \mathcal{C} и \mathcal{B} представляют собой пространства циклов и разрезов соответствующего D неориентированного графа G .

12.2. Вернемся теперь к задаче подсчета остовных деревьев в графе G .

12.2.1. Напомним некоторые определения из линейной алгебры.

Определение 12.12. Прямоугольная матрица M размерами $n \times m$ называется унимодулярной, если определитель любой ее полной квадратной подматрицы $M_{n,n}$ принимает значения из множества $\{0, +1, -1\}$, и вполне унимодулярной, если определитель любой ее квадратной подматрицы принимает значения из этого множества.

Теорема 12.13 (Пуанкаре, 1901). Матрица инцидентности \mathbf{M}_i орграфа D является вполне унимодулярной матрицей.

Доказательство. Пусть A есть произвольная квадратная подматрица матрицы \mathbf{M}_i . Если каждый ее столбец содержит два отличных от нуля числа, то этими числами являются 1 и -1 . Следовательно, сумма всех строк этой подматрицы равна нулю, так что $\det(A) = 0$. В противном случае у матрицы A имеется столбец, в котором ровно одно число отлично от нуля. Раскладывая тогда $\det(A)$ по этому столбцу, мы получаем, что

$$\det(A) = \pm \det(A'),$$

где A' — матрица, полученная из A вычеркиванием строки и столбца. Продолжая этот процесс, мы получаем требуемый результат. \square

Аналогично показывается, что вполне унимодулярными являются также и матрицы \mathbf{B} и \mathbf{C} , составленные из базисных векторов пространств \mathcal{B} и \mathcal{C} . В частности, вполне унимодулярной является и матрица Кирхгофа \mathbf{K} орграфа D .

12.2.2. Следующее утверждение мы уже доказывали в параграфе, посвященном матричной теореме о деревьях.

Утверждение 12.14. Пусть S есть произвольный набор из $(n - 1)$ -го ребра орграфа D , а B_S есть подматрица базисной матрицы, столбцы которой отвечают ребрам из набора S . Определитель $\det(B_S) \neq 0$ тогда и только тогда, когда индуцированный S подграф T соответствующего D графа G представляет собой остовное дерево.

Покажем, как просто доказать это утверждение, используя полученные в данном параграфе результаты. Заметим, что определитель $\det(B_S)$ равен нулю тогда и только тогда, когда отвечающие S столбцы линейно зависимы:

$$\sum_{e \in S} \mathbf{b}_e \cdot f_e = \mathbf{0}.$$

Последнее равенство, в свою очередь, означает, что в орграфе D существует циркуляция \mathbf{f} , носитель которой содержится в S . Но тогда, согласно лемме 12.7, подмножество S содержит цикл, то есть деревом не является. Наоборот, пусть S содержит цикл C . Рассмотрим порожденную этим циклом циркуляцию \mathbf{f}_C . Так как

$$\mathbf{M}_i \cdot \mathbf{f}_C = \mathbf{0} \quad \Longleftrightarrow \quad \sum_{e \in C} \mathbf{b}_e \cdot f_C(e) = \mathbf{0},$$

а $C \subseteq S$, то столбцы B_S линейно зависимы и определитель $\det(B_S) = 0$.

Теорема 12.15. Пусть D есть связный орграф, B — базисная матрица пространства \mathcal{B} . Тогда количество $t(D)$ остовных деревьев рассчитывается по формуле

$$t(D) = \det(B \cdot B^T).$$

Доказательство. Воспользуемся формулой Бине-Коши для вычисления квадратной матрицы $B \cdot B^T$, являющейся произведением двух прямоугольных матриц:

$$\det(B \cdot B^T) = \sum_S (\det(B_S))^2,$$

где S — некоторое подмножество множества столбцов матрицы B размером $n - 1$. Так как матрица B унимодулярная, то каждое слагаемое в этой сумме равно 0 или 1. Количество же ненулевых слагаемых в этой сумме совпадает, согласно предыдущему утверждению, с количеством остовных деревьев графа G . \square

Следствие 12.16. Пусть K есть матрица Кирхгофа. Тогда

$$t(D) = \det(K \cdot K^T).$$

12.3. Перейдем теперь к использованию полученных выше результатов в задачах, связанных с расчетами электрических цепей.

12.3.1. Мы уже замечали ранее, что любой вектор \mathbf{f} циркуляции в орграфе D с физической точки зрения можно рассматривать как некоторый ток в сети, моделируемой орграфом D , а любой вектор $\mathbf{g} = \mathbf{p} \cdot \mathbf{V}_i$ напряжения в D с той же точки зрения можно трактовать как некоторое напряжение в этой сети, задаваемое вектором-строкой потенциалов \mathbf{p} . В реальной жизни напряжение и ток в сети связаны друг с другом — согласно закону Ома, падение напряжения U между двумя узлами цепи пропорционально силе тока I на этом участке цепи:

$$u = r \cdot i.$$

Коэффициент пропорциональности r носит название электрического сопротивления участка цепи. В случае, если между рассматриваемыми узлами электрической цепи имеется источник электродвижущей силы (ЭДС) — идеализированный генератор напряжения — мощностью h , то закон Ома переписывается в виде

$$u = r \cdot i + h.$$

В матричном виде это уравнение можно записать так:

$$\mathbf{u} = \mathbf{R} \cdot \mathbf{i} + \mathbf{h}.$$

Здесь \mathbf{R} — диагональная матрица сопротивлений, \mathbf{h} — вектор ЭДС, \mathbf{i} и \mathbf{u} — векторы силы тока и напряжения в рассматриваемой электрической цепи.

12.3.2. В 1847 году Кирхгоф сформулировал два закона, описывающих поведение токов и напряжений в электрической сети. Первый закон гласит, что электрическая сумма токов в каждом узле электрической цепи равна нулю. На языке теории графов это означает, что \mathbf{i} представляет собой циркуляцию в орграфе D , моделирующем электрическую сеть. Как следствие, первый закон Кирхгофа можно переписать в виде

$$\mathbf{M}_i \cdot \mathbf{i} = \mathbf{0},$$

где \mathbf{M}_i — матрица инцидентности орграфа D .

Так как размерность пространства \mathcal{B} столбцов матрицы \mathbf{M}_i равна $n - 1$, то тому же значению равен и ранг матрицы \mathbf{M}_i . Мы знаем, что сумма всех строк матрицы \mathbf{M}_i даст нам нулевую строку. Следовательно, любая строка этой матрицы есть линейная комбинация оставшихся $n - 1$ строк. Удаляя эту строку, мы получим так называемую *матрицу Кирхгофа* \mathbf{K} , ранг которой равен $n - 1$. Следовательно, строки этой матрицы мы также можем взять в качестве базисных строк пространства \mathcal{B} . Первый же закон Кирхгофа мы, таким образом, можем переписать так:

$$\mathbf{K} \cdot \mathbf{i} = \mathbf{0}.$$

Действительно, если вектор силы тока \mathbf{i} окажется ортогональным всем $n - 1$ строкам матрицы \mathbf{K} , то он автоматически окажется ортогональным и последней, n -й строке матрицы \mathbf{M}_1 .

12.3.3. Второй закон Кирхгофа утверждает, что алгебраическая сумма падения напряжения на любом замкнутом контуре электрической цепи равна нулю. Перебирать все циклы в моделирующей электрическую цепь орграфе нам не нужно — достаточно взять систему фундаментальных циклов соответствующего D неориентированного графа G . Предположим, что вектор \mathbf{u} напряжений электрической цепи удовлетворяет матричному уравнению

$$\mathbf{u} \cdot \mathbf{C} = \mathbf{0} \quad \Longleftrightarrow \quad \mathbf{C}^T \cdot \mathbf{u} = \mathbf{0},$$

где

$$\mathbf{C} = [\mathbf{f}_{e_1} \quad \mathbf{f}_{e_2} \quad \dots \quad \mathbf{f}_{e_{m-(n-1)}}]$$

есть базисная матрица пространства циклов, $m - (n - 1)$ столбцов которой состоят из векторов \mathbf{f}_{e_i} фундаментальных циркуляций, отвечающих фундаментальным циклам графа G . В этом случае падение напряжения на любом контуре электрической цепи также будет равняться нулю.

12.3.4. Закон Ома и законы Кирхгофа представляют собой замкнутую систему линейных алгебраических уравнений для определения силы тока и напряжения в электрической сети при заданных сопротивлениях и ЭДС. Покажем, как решить эту систему, используя изложенные в первой части параграфа сведения.

Пусть T есть какое-то остовное дерево соответствующего D графа G . Свяжем с этим деревом систему \mathbf{f}_{e_i} фундаментальных циркуляций и построим по этим векторам матрицу \mathbf{C} . Эту матрицу нам будет удобно записать в следующем блочном виде:

$$\mathbf{C} = \begin{bmatrix} \mathbf{C}_T \\ \mathbf{E}_{m-n+1} \end{bmatrix}.$$

Здесь \mathbf{E}_{m-n+1} есть единичная матрица размерами $(m - n + 1) \times (m - n + 1)$, а \mathbf{C}_T — неособая подматрица размерами $(n - 1) \times (n - 1)$, отвечающая ребрам дерева T .

В аналогичном блочном виде мы можем представить и матрицу Кирхгофа:

$$\mathbf{K} = [\mathbf{K}_T \quad \mathbf{K}_{m-n+1}].$$

Заметим, что так как \mathbf{K}_T порождена столбцами, соответствующими дереву T , то определитель этой подматрицы отличен от нуля, то есть матрица \mathbf{K}_T особой не является.

Так как любой цикл ортогонален любой строке матрицы \mathbf{K} , то

$$\mathbf{K} \cdot \mathbf{C} = \mathbf{0} \quad \Longleftrightarrow \quad \mathbf{K}_T \cdot \mathbf{C}_T + \mathbf{K}_{m-n+1} = \mathbf{0} \quad \Longleftrightarrow \quad \mathbf{C}_T = -\mathbf{K}_T^{-1} \cdot \mathbf{K}_{m-n+1}.$$

12.3.5. Вернемся к электрической цепи. Как и матрицы \mathbf{K} и \mathbf{C} , мы можем представить векторы \mathbf{i} и \mathbf{u} в виде

$$\mathbf{i} = \begin{bmatrix} \mathbf{i}_T \\ \mathbf{i}_{m-n+1} \end{bmatrix} \quad \text{и} \quad \mathbf{u} = \begin{bmatrix} \mathbf{u}_T \\ \mathbf{u}_{m-n+1} \end{bmatrix}.$$

Тогда из первого закона Кирхгофа $\mathbf{K} \cdot \mathbf{i} = \mathbf{0}$ мы получаем следующие равенства:

$$\mathbf{K}_T \cdot \mathbf{i}_T + \mathbf{K}_{m-n+1} \cdot \mathbf{i}_{m-n+1} = \mathbf{0} \quad \Leftrightarrow \quad \mathbf{i}_T = -\mathbf{K}_T^{-1} \cdot \mathbf{K}_{m-n+1} \cdot \mathbf{i}_{m-n+1} = \mathbf{C}_T \cdot \mathbf{i}_{m-n+1} \quad \Leftrightarrow \quad \mathbf{i} = \mathbf{C} \cdot \mathbf{i}_{m-n+1}.$$

Иными словами, мы показали, что значения силы тока полностью определяются значениями силы тока на ребрах, не принадлежащих остовному дереву T .

Подставим теперь полученное выражение для \mathbf{i} в закон Ома:

$$\mathbf{u} = \mathbf{R} \cdot \mathbf{C} \cdot \mathbf{i}_{m-n+1} + \mathbf{h}.$$

Домножая это уравнение слева на \mathbf{C}^T и учитывая второй закон Кирхгофа $\mathbf{C}^T \cdot \mathbf{u} = \mathbf{0}$, мы получаем систему линейных уравнений

$$(\mathbf{C}^T \cdot \mathbf{R} \cdot \mathbf{C}) \cdot \mathbf{i}_{m-n+1} = -\mathbf{C}^T \cdot \mathbf{h},$$

позволяющую определить элементы вектора \mathbf{i} , принадлежащие \mathbf{i}_{m-n+1} . Зная эти компоненты, мы вначале с помощью равенства $\mathbf{i} = \mathbf{C} \cdot \mathbf{i}_{m-n+1}$ определим оставшиеся значения вектора \mathbf{i} , а затем из закона Ома получим значения вектора \mathbf{u} напряжений электрической сети.