

Парадигма развития науки
Методологическое обеспечение

А. Е. Кононюк

ДИСКРЕТНО-НЕПРЕРЫВНАЯ
МАТЕМАТИКА

Книга 7

Графы

Часть 2

Киев
«Освіта України»
2015



Кононюк Анатолий Ефимович



УДК 51 (075.8)

ББК В161.я7

К213

Рецензенты:

В. В. Довгай — к-т физ.-мат. наук, доц. (Национальный тех—
нический университет «КПІ»);

В. В. Гавриленко — д-р физ.-мат. наук, проф.,

О. П. Будя — к-т техн. наук, доц. (Киевский университет эко—
номики, туризма и права);

Н. К. Печурин — д-р техн. наук, проф. (Национальный ави—
ационный университет).

Кононюк А. Е.

К213 Дискретно-непрерывная математика. (Графы. К.7 Ч.2). — В
15-и кн. Кн 7,— К.: Освіта України. 2015. — 512 с.

ISBN 978-966-373-693-8 (многотомное издание)

ISBN 978-966-373-694-5 (книга 7)

Многотомная работа содержит систематическое изложение математических дисциплин, используемых при моделировании и исследованиях математических моделей систем.

В работе излагаются основы теории множеств, отношений, поверхностей, пространств, алгебраических систем, матриц, графов, математической логики, теории вероятностей и массового обслуживания, теории формальных грамматик и автоматов, теории алгоритмов, которые в совокупности образуют единую методологически взаимосвязанную математическую систему «Дискретно-непрерывная математика».

Для бакалавров, специалистов, магистров, аспирантов, докторантов и просто ученых и специалистов всех специальностей.

УДК 51 (075.8)

ББК В161.я7

ISBN 978-966-373-693-8 (многотомное издание)

ISBN 978-966-373-694-5 (книга 7)

© Кононюк А. Е., 2015

© Освіта України, 2015

Структура
открытой развивающейся панмедийной системы математических наук (дисциплин)



Оглавление

1. Ориентированные графы.....	10
1.1. Вводные понятия представления графов.....	10
1.2. Маршруты, циклы и связность.....	16
1.3. Маршруты и связность в орграфах.....	21
1.4. Упорядоченные орграфы и обходы.....	26
1.5. Отношение включения и достижимые множества.....	33
1.6. Теорема о гомоморфизме.....	37
1.7. Транзитивные графы и погружения в отношения упорядочения.....	39
1.8. Базисные графы.....	41
1.9. Чередующиеся цепи.....	45
1.10. Суграфы первой степени в графе.....	47
1.11. Перечисление орграфов.....	50
1.12. Турниры.....	56
1.13. Ориентации графов.....	61
1.14. Смешанные графы.....	63
1.15. Правильная нумерация.....	68
1.16. Длины путей, протяженности и расстояния между вершинами графа.....	69
1.17. Графы с отмеченными ребрами и вершинами.....	70
1.18. Квадрат матрицы смежности.....	71
1.19. Лексикографическая нумерация многочленов.....	72
1.20. Новые свободные переменные.....	73
1.21. Старые и новые классы ребер и вершин.....	73

1.22. Группа автоморфизмов графа.....	74
1.23. О проблеме изоморфизма графов.....	75
1.24. Классы эквивалентных вершин.....	76
1.25. Разбиения классов вершин и ребер.....	76
2. Раскраски графов.....	78
2.1. Введение.....	78
2.2. Раскраска вершин и ребер графа.....	80
2.3. Характеризация раскраски графов.....	90
2.4. Оценки хроматических чисел.....	105
2.4.1. Нижние оценки для $\chi(G)$	105
2.4.2. Верхние оценки для $\chi(G)$	107
2.5. Точные алгоритмы раскраски.....	107
2.5.1. Метод динамического программирования.....	108
2.5.2. Формулировка задачи о раскраске на языке 0-1-программирования.....	113
2.5.3. Сведение задачи о раскраске к ЗНП.....	115
2.5.4. Алгоритм прямого неявного перебора, использующий дерево поиска.....	117
2.6. Приближенные алгоритмы раскрашивания.....	119
2.6.1. Последовательные методы, основанные на упорядочивании множества вершин.....	119
2.7. Обобщения и приложения.....	120
2.7.1. Простая задача размещения (загрузки).....	121
2.7.2. Составление-графиков осмотра (проверки).....	122
2.7.3. Распределение ресурсов.....	123
2.8. Перечисление раскрашенных и других графов.....	123
2.8.1. Число графов.....	123
2.8.2. Число связных графов.....	133
2.8.3. Число k -раскрашенных графов.....	136
2.8.4. Ациклические орграфы.....	138
2.8.5. Деревья.....	141
2.8.6. Эйлеровы контуры в орграфах.....	147
2.8.7. Раскрашенные графы.....	151
2.8.8. Корневые графы.....	159
2.8.9. Надграфы и раскрашенные графы.....	163
2.8.10. Булевы функции.....	171
2.8.11. Хроматическое число.....	175
2.8.12. Суммы хроматических графов.....	178
2.8.13. Критические графы.....	180
2.8.14. Полиномы раскрашиваний.....	186
2.8.15. Группы автоморфизмов.....	190
2.8.16. Цветные графы Кэли для групп.....	194

2.8.17. Графы с заданными группами.....	196
2.8.18. Реберные отображения.....	198
3. Размещение центров и медиан в графе.....	202
3.1. Разделения.....	203
3.2. Центр и радиус.....	205
3.3. Размещение аварийных служб и пунктов обслуживания.....	206
3.3.1 Абсолютный центр.....	207
3.4. Алгоритмы нахождения абсолютных центров.....	210
3.4.1. Метод Хаками.....	210
3.4.2. Размещение аварийных служб (общий случай).....	211
3.4.3. Модифицированный метод Хаками.....	216
3.4.4. Итерационный метод.....	218
3.5. Кратные центры (p -центры).....	219
3.5.1. Задача размещения нескольких пунктов обслуживания.....	220
3.6. Абсолютные p -центры.....	221
3.7. Алгоритм нахождения абсолютных p -центров.....	222
3.7.1. Описание алгоритма.....	224
3.7.2. Вычислительные аспекты.....	225
3.7.3. Пример.....	226
3.7.4. Результаты вычислений.....	230
3.7.5. Применение общего алгоритма для поиска p -центров.....	231
3.8. Медиана графа.....	232
3.8.1. Выбор места для склада.....	233
3.9. Кратные медианы (p -медианы) графа.....	234
3.9.1. Абсолютные p -медианы.....	235
3.10. Обобщенная p -медиана графа.....	236
3.11. Методы решения задачи о p -медиане.....	238
3.11.1. Формулировка задачи в терминах целочисленного программирования.....	238
3.11.2. Алгоритм направленного древовидного поиска.....	240
3.11.3. Другой алгоритм направленного поиска.....	243
3.11.4. Приближенный алгоритм.....	244
4. Частичная упорядоченность.....	247
4.1. Графы частичных упорядочений.....	247
4.2. Представления в виде сумм упорядоченных множеств.....	248
4.3. Структуры и структурные операции.....	253
4.4. Размерность в частичном упорядочении.....	257
4.5. Соответствия Галуа.....	260
4.6. Связи Галуа для бинарных отношений.....	264
4.7. Отношения чередующегося произведения.....	268
4.8. Отношения Феррерса.....	271
4.9. Матричный анализ связей в частично ориентированных графах.....	273

5. Кратчайшие пути.....	281
5.1. Кратчайший путь между двумя заданными вершинами s и t	283
5.1.1. Случай неотрицательной матрицы весов.....	283
5.1.2. Случай общей матрицы весов.....	290
5.2. Кратчайшие пути между всеми парами вершин.....	295
5.2.1. Алгоритм Флойда (для произвольной матрицы весов).....	296
5.3. Обнаружение циклов отрицательного веса.....	297
5.3.1. Оптимальные циклы в графах с двойными весами.....	299
5.4. Нахождение K кратчайших путей между двумя заданными вершинами.....	300
5.5. Кратчайший путь между двумя заданными вершинами в ориентированном ациклическом графе.....	304
5.5.1. Алгоритм нахождения самого длинного (критического) пути в ориентированном ациклическом графе.....	305
5.6. Задачи, близкие к задаче о кратчайшем пути.....	308
5.6.1. Наиболее надежный путь.....	308
5.6.2. Путь с наибольшей пропускной способностью.....	309
5.6.3. Путь с наибольшей приведенной пропускной способностью.....	314
5.6.4. Метод нахождения пути наибольшей приведенной пропускной способности.....	316
5.7. Цикломатическое число и фундаментальные циклы.....	321
5.8. Деревья графа.....	326
5.8.1. Формирована дерева графа.....	326
5.8.2. Выявление всех деревьев графа.....	330
5.8.3. k -деревья.....	331
5.8.4. Деревья с корнем. Ветви.....	332
5.8.5. Типы вершин и центры деревьев.....	334
5.8.6. Характеристики графов.....	337
5.8.7. Максимальные графы исключения.....	339
5.8.8. Максимальные паросочетания и устойчивые множества вершин в графе.....	339
5.8.9. Максимальные деревья.....	341
5.8.10. Двудольные графы.....	343
5.9. Анатолия графов.....	347
5.9.1. Вводные замечания.....	347
5.9.2. Свойства матрицы инцидентности.....	347
5.9.3. Деревья и дополнения.....	350
5.9.4. Разрезы.....	354
5.9.5. Матрицы циклов и разрезов.....	360
5.9.6. Матрица сечений.....	362
5.9.7. Матрица контуров.....	365
5.9.8. Связь между топологическими матрицами.....	367

5.9.9. Пространство суграфов.....	368
5.9.10. Несвязные графы.....	369
5.9.11. Топология ориентированных графов.....	371
5.10. Эйлеровы циклы и задача китайского почтальона.....	373
5.10.1. Некоторые родственные задачи.....	376
5.10.2. Алгоритм для задачи китайского почтальона.....	378
5.10.3. Связь между эйлеровыми и гамильтоновыми циклами.....	384
6. Числа, функции, ядра графов.....	386
6.1.Основные числа теории графов.....	386
6.1.1. Цикломатическое число.....	386
6.1.2. Хроматическое число.....	390
6.1.3. Число внутренней устойчивости.....	395
6.1.4. Число внешней устойчивости.....	401
6.2. Ядра графа.....	406
6.2.1 Теоремы существования и единственности.....	
6.2.2. Приложение к функциям Гранди.....	412
6.3. Игры на графе.....	413
6.3.1. Игра Ним.....	413
6.3.2. Общее определение игры (с полной информацией).....	419
6.3.3. Стратегии.....	421
7. Гамильтоновы циклы и цепи.....	425
7.1. Введение.....	425
7.2. Гамильтоновы циклы в графе.....	429
7. 2.1. Алгебраический метод.....	429
7. 2.2. Метод перебора Робертса и Флореса.....	434
7.2.3. Мультицепной метод.....	439
7.3. Сравнение методов поиска гамильтоновых циклов.....	446
7.4. Простая задача планирования.....	450
7. 4.1. Вычислительные аспекты.....	451
7.5. Задача коммивояжера.....	452
7.5.1. Нижняя граница из задачи о назначениях.....	453
7.5.2. Нижняя граница из задачи о кратчайшем остове.....	453
7.5.3. Двойственность.....	454
7.6. Задача коммивояжера и задача о кратчайшем остове.....	456
7.6.1. Определения.....	457
7.6.2. Алгоритм поиска, использующий дерево решений.....	459
7.6.3. Алгоритм штрафования вершин.....	461
7.6.4. Задачи, родственные задаче коммивояжера.....	473
7.7. Задача коммивояжера и задача о назначениях.....	473
7.7.1. Алгоритм поиска, использующий дерево решений.....	474
7.7.2. Пример.....	478

7.7.3. Вычислительные комментарии и характеристики.....	485
7.7.4. Лучшие границы для дерева поиска.....	486
7.7.5. Пример из раздела.....	494
7.7.6. Приложение.....	496
8. Ациклические графы.....	497
8.1. Базисные графы.....	497
8.2. Деформации цепей.....	498
8.3. Графы воспроизведения.....	501
Приложение 1.....	505
Литература.....	512

1. Ориентированные графы

1.1. Вводные понятия представления графов

Многие отношения на конечных множествах могут быть изображены в виде рисунков, с которыми можно работать при помощи соответствующих матриц. Перед тем как определить конструкции этих рисунков, необходимо быть уверенными в том, что это не повлечет за собой никаких двусмысленностей. Введем необходимые понятия.

Пусть V — конечное множество и

$$I_V = \{(v, v): v \in V\}.$$

Положим

$$V^2_- = V^2 \setminus I_V = \{(v_1, v_2): v_1 \neq v_2\}$$

и определим на V^2_- отношение эквивалентности следующим образом:

$$\{(v_1, v_2) \sim (w_1, w_2), \text{ если } (v_1, v_2) = (w_1, w_2) \\ \text{или } (v_1, v_2) = (w_2, w_1)\}.$$

Важное свойство отношения \sim сформулировано в следующем утверждении.

Утверждение. *Отношение \sim является отношением эквивалентности на V^2_- .*

Доказательство оставляем в качестве упражнения.

Множество эквивалентных классов, определенное таким образом, обозначим через V^2_-/\sim . Каждый класс эквивалентности содержит ровно два элемента, так как если $(v_1, v_2) \in V^2_-$, то $[(v_1, v_2)] = \{(v_1, v_2), (v_2, v_1)\}$. Здесь $[(v_1, v_2)]$ — класс эквивалентности, содержащий (v_1, v_2) .

Дадим строгое определение графа.

Определение. *Графом G называется пара $G = (V, E)$, где V — непустое конечное множество вершин, а E — подмножество V^2_-/\sim*

Другими словами, можно сказать, что *граф G есть пара $G = (V, E)$, где V — непустое конечное множество вершин, а E — множество неупорядоченных пар различных вершин.*

Множество E называют множеством *ребер* графа, $|V|$ обозначает число вершин G , $|E|$ — число ребер G .

Следующий результат выражает связь между графами и классами отношений на конечных множествах.

Утверждение.

а) *Граф $G = (V, E)$ определяет нерефлексивное симметричное отношение на V .*

б) *Нерефлексивное симметричное отношение на конечном множестве V определяет граф.*

Доказательство.

а) Пусть $G=(V, E)$ — граф. Определим отношение $R(E)$ на V следующим образом: $v_1R(E)v_2$ тогда и только тогда, когда $[v_1, v_2] \in E$, Отношение $R(E)$ нерефлексивно, так как $vR(E)v$ тогда и только тогда, когда $[v, v] \in E$, но $[v, v] \notin E$, поскольку $(v, v) \notin V^2_-$. $R(E)$ симметрично для $v_1R(E)v_2$ тогда и только тогда, когда $[v_1, v_2] \in E$, однако $[v_1, v_2] = \{v_1, v_2\}, (v_2, v_1) \in [v_2, v_1]$. Следовательно, $v_1R(E)v_2$ тогда и только тогда, когда $v_2R(E)v_1$.

б) Если R — нерефлексивное симметричное отношение на V , то $R \subset V^2_-$.

Нерефлексивность R означает, что $(v, v) \notin R$ для любого $v \in V$, поэтому $R \subset V^2_-$.

Симметричность R означает, что $(v_1, v_2) \in R$ тогда и только тогда, когда $(v_2, v_1) \in R$. Определим E формулой $E = R/\sim$, тогда $G = (V, E)$ есть искомый граф.

Графы могут быть представлены матрицами с булевыми элементами. Многие из свойств графов могут быть определены из их матричных представлений путем алгебраических преобразований. Это станет понятным из последующего изложения.

Определение. Матрица смежности $A \in M(n, \mathbf{B})$ графа $G = (V, E)$, где $|V|=n$, определяется следующим образом:

$$A_{ij} = \begin{cases} 1, & \text{если } [v_i, v_j] \in E, \\ 0 & \text{в противном случае.} \end{cases}$$

Говорят, что вершины v_i и v_j являются смежными, если $A_{ij}= 1$. Ясно, что $A_{ii}=0$ ($i=1, \dots, n$) и $A=A^T$. Таким образом, A симметрична, и $A = A(R(E))$.

Изображение графа $G = (V, E)$ получается путем расположения различных точек на \mathbf{R}^2 для каждой $v \in V$, причем, если $[v, w] \in E$, мы проводим линию, соединяющую вершины v и w .

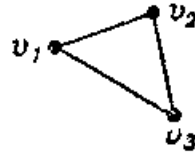
Пример 1. Пусть

$$1. \ V = \{v_1, v_2, v_3\}, \ E = \{[v_1, v_2], [v_2, v_3], [v_1, v_3]\}, \ |V| = 3, \ |E| = 3.$$

Этот граф изображен на рис. 1.

$$A = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$$

Матрица смежности



Изображение

Рис. 1.

$$2. V = \{v_1, v_2, v_3, v_4, v_5\},$$

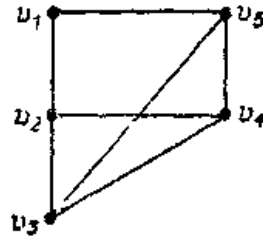
$$E = \{[v_1, v_2], [v_1, v_5], [v_2, v_3], [v_2, v_4], [v_3, v_5], [v_3, v_4], [v_4, v_5]\},$$

$$|V| = 5, |E| = 7.$$

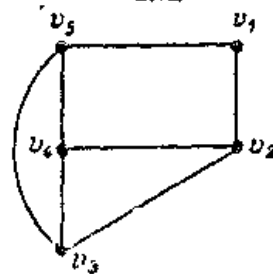
Этот граф изображен на рис. 2.

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 \end{bmatrix}$$

Матрица смежности



или

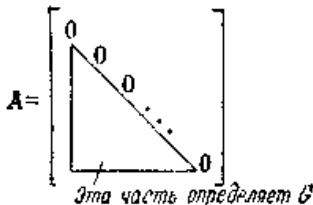


Изображение

Рис. 2.

Графы являются скорее «топологическими», чем «геометрическими» объектами, т. е. они выражают больше отношения между вершинами, чем расположение вершин и ребер в пространстве. Таким образом, граф может быть изображен бесконечным количеством разных, но

«эквивалентных» способов. Однако изображения графов могут вводить в заблуждение. Например, из пересечения двух ребер на рисунке не следует, что точка пересечения является вершиной (см. первую диаграмму на рис. 2). Ясно, что нижней (верхней) треугольной части матрицы смежности достаточно, чтобы определить граф.



Дадим следующие определения.

Определение. Говорят, что граф $H = (V_1, E_1)$ является *подграфом* графа $G = (V, E)$, если $V_1 \subseteq V$ и $E_1 \subseteq E$. Если $V_1 = V$, то говорят, что H является *остовным подграфом* G . Если V_1 — непустое подмножество вершин графа (V, E) , то подграф (V_1, E_1) , порожденный V_1 , определяют как

$$[v, w] \in E_1 \Leftrightarrow v, w \in V_1 \text{ и } [v, w] \in E.$$

Определение.

а) Пусть $G_1 = (V_1, E_1)$ и $G_2 = (V_2, E_2)$ — графы. Будем говорить, что G_1 и G_2 *эквивалентны*, если существует биекция $f: V_1 \rightarrow V_2$ такая, что

$$vR(E_1)w \Rightarrow f(v)R(E_2)f(w).$$

б) Пусть $G = (V, E)$ — произвольный граф. Определим отображение

$$\delta: V \rightarrow \mathbf{N} \cup \{0\}$$

следующим образом: величина $\delta(v)$ равна числу ребер, содержащих вершину $x \in V$. Назовем $\delta(v)$ *степенью* вершины v .

Следующее утверждение выражает два простых, но важных факта о свойствах графов.

Утверждение.

а) $\sum_{v \in V} \delta(v) = 2|E|.$

б) В любом графе число вершин нечетной степени четно.

Доказательство. Каждое ребро дважды входит в сумму, откуда и следует утверждение.

в) Пусть $V_e \subseteq V$ — множество вершин четной степени, а $V_o \subseteq V$ — множество вершин нечетной степени. Заметим, что

$$V = V_e \cup V_o \text{ и } V_e \cap V_o = \emptyset;$$

следовательно,

$$\sum_{v \in V} \delta(v) = \sum_{v \in V_0} \delta(v) + \sum_{v \in V_1} \delta(v),$$

$$2|E| = 2k + \sum_{v \in V_1} \delta(v).$$

(Ясно, что $\sum_{v \in V_0} \delta(v) = 2k$, где k — некоторое целое.) Таким образом,

$$\sum_{v \in V_1} \delta(v) = 2(|E| - k),$$

т. е. чётно, однако каждое $\delta(v)$ в левой части нечётно, поэтому $|V|$ чётно.

Во многих приложениях теории графов о топологии графа имеется дополнительная информация, относящаяся к V , или к E , или к обоим множествам одновременно. Чтобы конкретизировать вышесказанное, определим понятие помеченного графа и дадим несколько примеров.

Определение.

1) Пусть S_V и S_E — множества меток. *Пометкой* или *распределением меток графа* $G = (V, E)$ называется пара функций

$f: V \rightarrow S_V$ — распределение меток вершин,

$g: E \rightarrow S_E$ — распределение меток ребер.

2) Пусть граф $G = (V, E)$ помечен с помощью функций f и g , а

$G_1 = (V_1, E_1)$ помечен с помощью f_1 и g_1 . Графы G и G_1 называются *эквивалентно помеченными*, если существует биекция $h: V \rightarrow V_1$, такая что

а) G и G_1 эквивалентны как непомеченные графы;

б) $f(v) = f_1(h(v))$ для всех $v \in V$, поэтому соответствующие вершины имеют одну и ту же пометку;

в) $g([v, w]) = g_1([h(v), h(w)])$ для всех $v, w \in V$, т. е. соответствующие ребра имеют одну и ту же пометку.

Часто бывают помеченными только ребра или же только вершины. Вышесказанное применимо и в этом случае. Тогда

$$\left. \begin{array}{l} f: V \rightarrow S_V, \\ g = \text{const}, \end{array} \right\} \text{помечены только вершины;}$$

$$\left. \begin{array}{l} f = \text{const}, \\ g: E \rightarrow S_E, \end{array} \right\} \text{помечены только ребра.}$$

Ребра или вершины (или те и другие вместе) помеченного графа несут информацию, которая дополняет или заменяет обычную идентификацию с помощью имен.

Пример 2.

1. Пусть

$V = \{v_1, v_2, v_3, v_4\}$, $E = \{[v_1, v_3], [v_2, v_3], [v_3, v_4]\}$,
 $f: V \rightarrow \{\text{города Великобритании}\}$, $g: E \rightarrow \mathbb{N}$,
 $f(v_1)$ — Лондон, $g([v_1, v_3]) = 105$,
 $f(v_2)$ — Кардифф, $g([v_2, v_3]) = 196$,
 $f(v_3)$ — Бирмингем, $g([v_3, v_4]) = 292$,
 $f(v_4)$ — Эдинбург.
 Этот граф изображен на рис. 3.

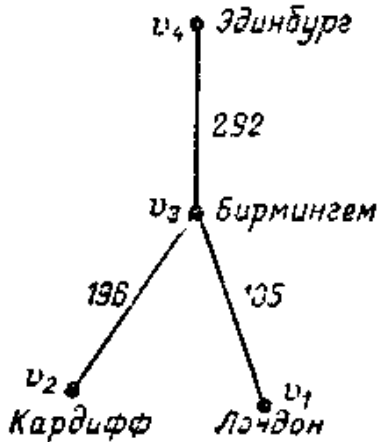


Рис. 3.

2. Пусть графы

$$\begin{aligned}
 G_1 &= (\{v_1, v_2, v_3\}, \{[v_1, v_2], \\
 &\quad [v_1, v_3], [v_2, v_3]\}, \\
 G_2 &= (\{w_1, w_2, w_3\}, \{[w_1, w_2], \\
 &\quad [w_1, w_3], [w_2, w_3]\})
 \end{aligned}$$

помечены так же, как указано на рис. 4; G_1 и G_2 являются эквивалентно помеченными графами (вершины не помечены).

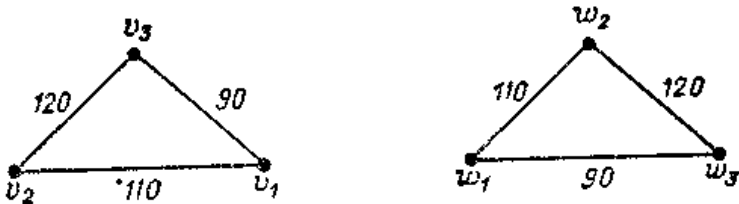


Рис. 4

Определение.

- а) Граф $G = (V, E)$ называется *полным*, если для всех $v_1, v_2 \in V$ имеем $[v_1, v_2] \in E$. Полный граф с вершинами обозначается через K_n .
- б) Граф $G = (V, E)$ называется *двудольным*, если существует разбиение $V = (V_1, V_2)$ такое, что никакие две вершины из V_1 или из V_2 не являются смежными. Двудольный граф называется *полным*, если для любой пары $v_1 \in V_1$ и $v_2 \in V_2$ имеем $[v_1, v_2] \in E$. Если $|V_1|=m$ и $|V_2|=n$, то полный двудольный граф (V, E) обозначается через $K_{m,n}$.

1.2. Маршруты, циклы и связность

Обратим сейчас внимание на понятие маршрута в графе. Значительная часть теории графов и ее приложений занимается вопросами существования и свойств маршрутов. Некоторые важные свойства вытекают из следующих определений.

Определение.

- а) Пусть $G = (V, E)$ — граф. *Маршрутом* длины k в графе G из v в w называется последовательность $\langle v_0, v_1, \dots, v_k \rangle$ вершин (необязательно различных) $v_i \in V$ таких, что $v_0 = v$, $v_k = w$, а $[v_{i-1}, v_i] \in E$ для всех $i = 1, \dots, k$. Маршрут называется *замкнутым*, если $v_0 = v_k$. Маршрут называется *цепью*, если все его вершины различны. Замкнутая цепь называется *циклом*. Цикл называется *простым циклом*, если только $v_0 = v_k$, а остальные v_i различны.
- б) Если существует маршрут из v в w , $v, w \in V$, то говорят, что w *достижима* из v .
- в) Граф без циклов называется *ациклическим*.

Заметим, что понятие замкнутости, в сущности, соответствует своему названию.

Определение.

- а) Граф $G = (V, E)$ называется *связным*, если каждая пара различных вершин может быть соединена маршрутом.
- б) *Деревом* называется связный ациклический граф.
- в) *Корневым деревом* называется дерево с выделенной вершиной, называемой *корнем*.
- г) *Остовным деревом* для $G = (V, E)$ называется остовный подграф, являющийся деревом.

В 1.1 мы отметили, что вычисления с матрицей смежности обнаруживают важную информацию о природе графа. Следующие результаты являются примерами внутренних связей между алгеброй и топологией в теории графов.

В приведенной ниже теореме и ее следствиях степени A^k вычисляются в $M(n, \mathbf{Z})$, а не в $M(n, \mathbf{B})$. Следовательно, в A^k могут возникать числа, большие чем 1.

Теорема. Пусть A — матрица смежности графа $G = (V, E)$ и $|V| = n$. Тогда $(A^k)_{ij}$ есть число маршрутов длины k от v_i к v_j .

Доказательство. Будем использовать индукцию по k . Для $k = 1$ маршрут длины 1 как раз является ребром G . Следовательно, результат теоремы при $k = 1$ вытекает из определения A . Пусть

$$(A^{k-1})_{ij} = \alpha_{ij}, \quad A_{ij} = a_{ij},$$

тогда

$$(A^k)_{ij} = (A^{k-1}A)_{ij} = \sum_{q=1}^n \alpha_{iq}a_{qj}.$$

Пусть результат имеет место для $k - 1$. Тогда, если α_{iq} — элементы матрицы A^{k-1} , то α_{iq} — число маршрутов длины $k - 1$ от v_i к v_q ; по определению a_{qj} — число маршрутов длины 1 от v_q к v_j . Следовательно, $\alpha_{iq}a_{qj}$ — число маршрутов длины k из v_i к v_j , где v_q есть предпоследняя вершина маршрута.

Отсюда следует, что

$$\sum_{q=1}^n \alpha_{iq}a_{qj}$$

есть число маршрутов длины k от v_i к v_j . Это завершает доказательство.

Следствие.

а) Маршрут от v_i к v_j ($i \neq j$) в $G = (V, E)$ существует тогда и только тогда, когда (i, j) -й элемент матрицы порядка $n \times n$ ($n = |V|$):

$$A + A^2 + \dots + A^{n-1}$$

не равен нулю.

б) Если не использовать условие $i \neq j$, то требуемая матрица имеет вид

$$A + A^2 + \dots + A^{n-1} + A^n.$$

Доказательство.

а) Пусть $\langle v_i, v_1, \dots, v_j \rangle$ — маршрут из v_i в v_j в G . Если не существует повторяющихся вершин, то (так как $|V|=n$) маршрут содержит не более $n - 1$ ребер, и необходимое утверждение следует из теоремы.

Пусть существует повторяющаяся вершина. Тогда маршрут имеет вид

$$\langle v_{i_1}, \dots, \underbrace{v_{r_1}, \dots, v_{r_1}}_{\text{замкнутый маршрут}}, \dots, v_{j_1} \rangle.$$

Если мы удалим все такие замкнутые маршруты, то задача сведется к предыдущему случаю, когда вершины не повторялись. Таким образом, в одну сторону требуемый результат получен. В обратную сторону рассуждения очевидны.

б) Если разрешается $i=j$, то существование маршрута из v_i в v_j влечет то, что существует последовательность $\langle v_i, v_1, \dots, v_j \rangle$. Если не существует повторяющихся вершин (за исключением, возможно, случая $v_i = v_j$), тогда маршруты состоят из более чем $n + 1$ вершин (не более n ребер). Следовательно, (i, j) -й элемент матрицы $\sum_{k=1}^n A^k$ не равен нулю. Тогда при $|V| = n$ отсюда следует

$$\begin{aligned} A(R^+(E)) &= A(R(E)) \vee A(R^2(E)) \vee \dots \vee A(R^n(E)) = \\ &= \bigvee_{k=1}^n A(R^k(E)), \\ A(R^*(E)) &= I \vee A(R(E)) \vee \dots \vee A(R^{n-1}(E)) = \\ &= \bigvee_{k=1}^{n-1} A(R^k(E)). \end{aligned}$$

Напомним, что для произвольного бинарного отношения R величина R^+ определялась как

$$R^+ = \bigcup_{k=1}^{\infty} R^k,$$

и если $R \subseteq V \times V$ при $|V| = n$, то отсюда следует, что

$$A(R^+) = A^+(R) = \bigvee_{k=1}^n A(R^k).$$

Аналогично

$$A(R^*) = A^*(R) = \bigvee_{k=0}^{n-1} A(R^k).$$

В этом параграфе для упрощения обозначений будем теперь обозначать $A(R^k(E))$ через $A(R^k)$, $A(R^+(E))$ через $A(R^+)$, а $A(R^*(E))$ через $A(R^*)$. Алгоритм Уоршола требует $4n^3$ операций для определения $A(R^+)$, тогда как при помощи приведенных выше соотношений требуется $4n^4 - 7n^3$ операций. Можно получить и другие еще более эффективные алгоритмы для больших значений n .

Матрицу $C = A(R^*)$ называют *матрицей связи, связности* или *достижимости* графа $G = (V, E)$. Маршрут из v_i к v_j ($i \neq j$) существует в G тогда и только тогда, когда (i, j) -й элемент из C равен 1. Граф G является связным тогда и только тогда, когда $C_{ij} = 1$ для всех $1 \leq i, j \leq n$. Важные свойства отношения R^* могут быть сформулированы следующим образом.

Утверждение. R^* — отношение эквивалентности на V .

Доказательство. Так как по определению R^* является рефлексивным замыканием R , то необходимо только проверить симметричность R^* .

Выполнение vR^*w влечет существование маршрута $\langle v, v_1, \dots, v_k, w \rangle$ от v к w в G , т. е.

$$[v, v_1] \in E, [v_1, v_2] \in E, \dots, [v_k, w] \in E.$$

Следовательно,

$$[w, v_k] \in E, [v_k, v_{k-1}] \in E, \dots, [v_2, v_1] \in E, [v_1, v] \in E.$$

Таким образом,

$$\langle w, v_k, v_{k-1}, \dots, v_2, v_1, v \rangle$$

есть маршрут из w в v в G , откуда следует wR^*v .

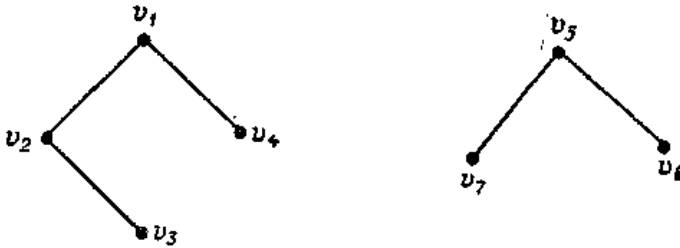
Отношение R^* определяет важный класс подграфов, который сейчас будет определен. Будут также даны некоторые сопутствующие понятия; они будут важны при обсуждении «пересечения» графа.

Определение. Пусть $\{V_i: 1 \leq i \leq p\}$ — разбиение графа, определяемое отношением R^* . Тогда говорят, что p — *число связности* G . Подграфы (V_i, E_i) , порожденные классами эквивалентности, называют *компонентами связности* графа G .

Лесом называется граф, в котором каждая связная компонента является деревом. *Остовный лес* для графа $G=(V, E)$ — это совокупность вершин разъединенные деревьев $T_i = (V_i, E_i)$ таких, что $V = \bigcup_i V_i$ и

$E_i \subset E$ для всех i . (Разъединенность вершин означает, что $V_i \cap V_j = \emptyset$ при $i \neq j$).

Рисунок 5 иллюстрирует вышеупомянутые понятия для графа при $p = 2$.



Остовный лес для графа

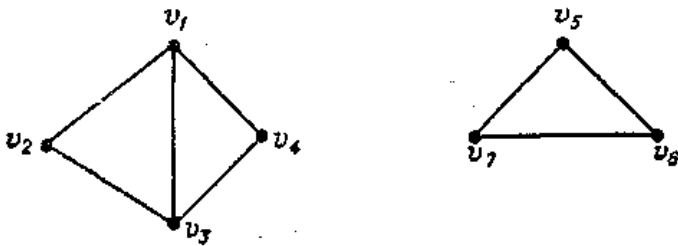


Рис. 5.

Во многих приложениях теории графов требуется, чтобы ребра графа имели направление. Например, поток данных проходит через программу.

Определение. *Ориентированный граф (орграф) G есть пара $G = (V, E)$, где V — конечное множество вершин, а E — произвольное подмножество $V \times V$.*

Утверждение.

- Ориентированный граф $G = (V, E)$ определяет отношение на V .*
- Пусть V — конечное множество. Тогда отношение на V определяет ориентированный граф, у которого множество вершин — V .*

Доказательство.

а) Определим $R(E)$ следующим образом: $vR(E)w$ тогда и только тогда, когда $(v, w) \in E$. Очевидно, что $R(E)$ — отношение.

б) Если R — отношение на V , то ориентированный граф $G=(V, E)$, определяемый R , имеет множество ребер E , где $(v, w) \in E$, тогда и только тогда, когда vRw .

Направление ребра обозначают порядком в $V \times V$; например, если

$(v, w) \in E$, то говорят, что ребро *выходит* из v и *входит* в w . На диаграмме в этом случае для указания направления используют стрелки.

Пример 1. Пусть $V = \{v_1, v_2, v_3\}$, а $E_1 = \{(v_1, v_2), (v_2, v_3), (v_3, v_1)\}$. Тогда матрица смежности и изображение орграфа $G_1 = (V, E_1)$ будут такими, как на рис. 6.

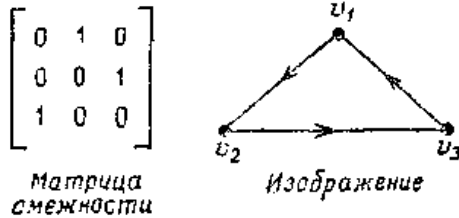


Рис. 6.

Аналогично на рис. 7 приведена матрица смежности и изображение графа $G_2 = (V, E_2)$, где

$$E_2 = \{(v_1, v_1), (v_1, v_2), (v_1, v_3), (v_2, v_3), (v_3, v_1)\}.$$

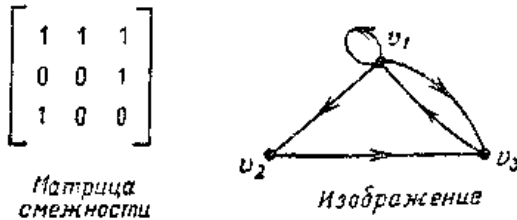


Рис. 7.

Поскольку реберное отношение для орграфа не обязательно симметрично или нерефлексивно, то, вообще говоря, не обязательно, чтобы $A = A^T$ или $A_{ii} = 0$. Ребра типа (v, v) называют *петлей*. Степень $\delta(v)$ вершины $v \in V$ может быть записана в виде суммы $\delta(v) = \delta^-(v) + \delta^+(v)$, где $\delta^-(v)$ — число ребер, входящих в v , а $\delta^+(v)$ — число ребер, выходящих из v . Множества $\{w: (w, v) \in E\}$ и $\{w: (v, w) \in E\}$ называют соответственно *входящим узлом* и *выходящим узлом* вершины $v \in V$. Понятия эквивалентности и пометки обобщаются на орграфы естественным образом.

1.3. Маршруты и связность в орграфах

Определение. *Маршрутом* длины k из v в w в орграфе $G = (V, E)$ называется последовательность ребер вида

$(v, w_1), (w_1, w_2), (w_2, w_3), \dots, (w_{k-2}, w_{k-1}), (w_{k-1}, w),$

т. е. вторая вершина каждого ребра совпадает с первой вершиной следующего ребра. Часто удобно представлять маршрут последовательностью вершин

$v, w_1, w_2, \dots, w_{k-2}, w_{k-1}, w,$

которые его определяют. Если $v = w$, то маршрут называют *замкнутым маршрутом* или *циклом*. Орграф без циклов называется *ациклическим*.

Теоремы п.1.2 также справедливы с аналогичными доказательствами для орграфов. Определим связность или матрицу достижимости тем же самым способом. Заметим, однако, что для орграфов отношение R^* не является отношением эквивалентности на V и, следовательно, не осуществляет разбиения V .

Пусть \mathcal{D} обозначает множество всех орграфов, а \mathcal{G} — множество всех графов. Мы можем определить отображение $\mathcal{F}: \mathcal{D} \rightarrow \mathcal{G}$ следующим образом.

Определение. Пусть $G = (V, E) \in \mathcal{D}$. Тогда множество вершин $F(G) \in \mathcal{G}$ совпадает с V , а множество ребер $F(G)$ определяется применением следующих операций на E :

- а) удаляются все петли из E ;
- б) (v, w) заменяются на $[v, w]$ для всех $(v, w) \in E$. Тогда $F(G)$ является графом, *связанным* с орграфом G .

Для орграфов понятие связности является более содержательным, чем для графов, и имеет отношение к проблеме обхода. Определим три важных типа связности орграфа.

Определение. Если $G = (V, E)$ — орграф, то будем говорить, что:

- а) G *слабо связный*, если граф $F(G)$ связный;
- б) G *односторонне связный*, если для каждой пары различных вершин $v, w \in V$ существует маршрут из v в w или обратно,
- в) G *сильно связный*, если для каждой пары различных вершин $v, w \in V$ существует маршрут из v в w и обратно.

Очевидно, что G сильно связный $\Rightarrow G$ односторонне связный $\Rightarrow G$ слабо связный.

Пример 3. Из рис. 8 мы видим, что орграф:

- а) только слабо связный (рис. 8,а);
- б) односторонне связный, но не сильно связный (рис. 8,б);
- в) сильно связный (рис. 8,с).

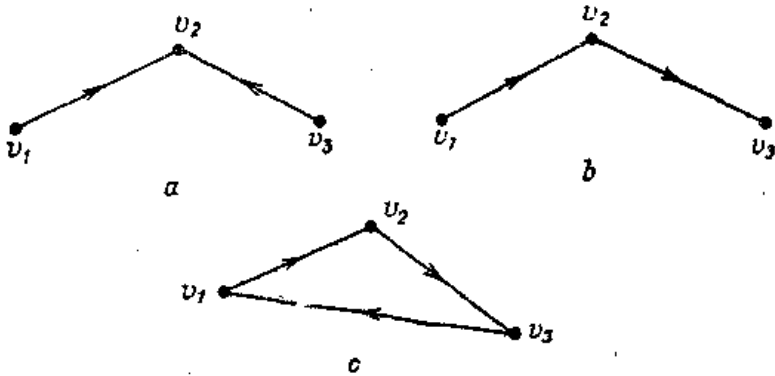


Рис. 8.

В терминах связности матрицы $C = A(R^*)$ оргграф G сильно связный тогда и только тогда, когда $C_{ij} = 1$ для всех $1 \leq i, j \leq n$; G односторонне связный тогда и только тогда, когда $C_{ij} \vee C_{ji} = 1$ для всех $1 \leq i, j \leq n$.

Пример 4. Рассмотрим оргграф, представленный диаграммой на рис. 9. Для этого оргграфа

$$A(R) = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad A(R^2) = \begin{bmatrix} 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{bmatrix},$$

$$A(R^3) = \begin{bmatrix} 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{bmatrix}, \quad A(R^4) = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 \end{bmatrix};$$

поэтому для

$$C = \bigvee_{h=0}^4 A(R^h) = I \vee A(R) \vee A(R^2) \vee A(R^3) \vee A(R^4)$$

имеем $C_{ij} = 1$ для всех $1 \leq i, j \leq 5$ и, следовательно, граф является сильно связным.

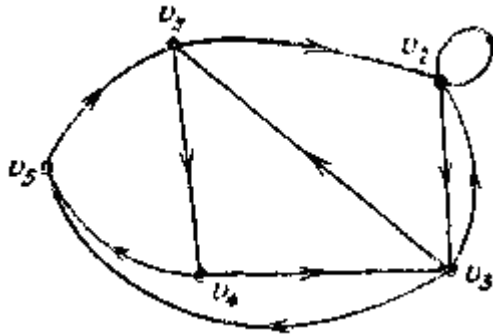


Рис. 9.

Для более эффективного вычисления C можно использовать алгоритм Уоршола.

Если $G = (V, E)$ — орграф, то можно разбить V путем определения отношения эквивалентности ρ следующим образом: $v \rho w$, если $v = w$ или существуют маршруты из v в w и обратно. Если $\{V_i: 1 \leq i \leq p\}$ — разбиение V и $\{E_i: 1 \leq i \leq p, \text{ а } E_i = (V_i \times V_i) \cap E\}$ являются соответствующими множествами ребер, то подграфы $G_i = (V_i, E_i)$ ($1 \leq i \leq p$) называются *сильно связными компонентами* G .

Очевидно, что $\rho \subseteq R^*$ и $A(\rho)$ может быть определено из $A(R^*)$ как $A(\rho)_{ij} = A(R^*)_{ij} \wedge A(R^*)_{ji}$; граф G сильно связный тогда и только тогда, когда G имеет только одну сильно связную компоненту, т. е. если $p=1$.

Пример 6. Для орграфа на рис. 10 имеем

$$A(R^*) = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}; \quad A(\rho) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

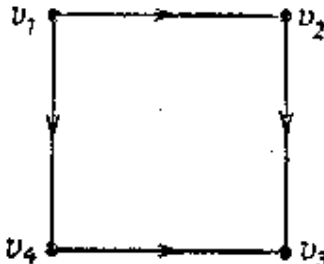


Рис. 10

Таким образом, $G_i = (\{v_i\}, \emptyset)$ ($1 \leq i \leq 4$) являются сильно связными компонентами графа.

Пусть $G = (V, E)$ — ациклический орграф. Вершину $v \in V$ называют *листом*, если $\delta^+(v)=0$. Если $(v, w) \in E$, то v является *непосредственным предком* w , а w — *непосредственным потомком* v . Если существует маршрут из v в w , то говорят, что v является *предком* w , а w — *потомком* v .

Эти понятия не имеют смысла для орграфов, имеющих циклы, так как для таких графов вершина может исходить сама из себя.

Пример 7. Для ациклического орграфа, изображенного на рис. 11, из вершин v_2, v_4 и v_5 ребра не выходят, v_1 является предком v_5 , v_5 является прямым потомком v_3 и т. д.

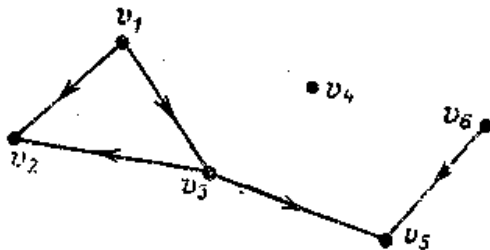


Рис. 11.

Существует тесная связь между ациклическими орграфами и частично упорядоченными отношениями. В частности, имеет место следующий результат, доказательство которого мы оставляем в качестве упражнения. Заметим, что для сокращения некоторых доказательств, приведенных ниже, частичные порядки будут основаны скорее на отношении $<$, чем на отношении \leq , и, следовательно, являются транзитивными и нерефлексивными.

Утверждение.

а) Пусть отношение $<$ является частичным отношением порядка на конечном множестве V , Тогда, если

$$E = \{(v, w): v < w\},$$

то пара $G=(V, E)$ является ациклическим графом.

б) Пусть $G=(V, E)$ — ациклический орграф и отношение $<$ определяется следующим образом; $v < w$, если v является предком w . Тогда отношение $<$ является частичным отношением порядка на V .

В терминах орграфов можно дать точное определение структурам данных, известным как *ориентированное дерево*.

Ориентированное дерево $T = (V, E)$ — это ациклический орграф, в котором одна вершина $v_r \in V$ не имеет предков, а каждая другая

вершина имеет только одного непосредственного предка; v_r называется *корнем* дерева. *Бинарное дерево* — это ориентированное дерево, в котором каждая вершина имеет не более двух непосредственных потомков, т. е. $\delta^+(v) \leq 2$ для всех $v \in V$. Говорят, что бинарное дерево является *полным*, если каждая вершина, не являющаяся листом, имеет ровно два непосредственных потомка.

Утверждение. Следующие утверждения эквивалентны по отношению к орграфу $G = (V, E)$:

- а) G является деревом.
- б) Граф $F(G)$ связный, и существует вершина v_r , которая не имеет предков, а все другие вершины имеют только по одному непосредственному предку.
- в) G имеет вершину v_r , которая соединяется с любой другой вершиной единственным маршрутом.
- г) G имеет вершину v_r , которая не имеет предков; все другие вершины имеют только одного непосредственного предка; существует маршрут к каждой вершине из v_r .

Доказательство оставляем в качестве упражнения.

1.4. Упорядоченные орграфы и обходы

Списки смежности являются альтернативной по отношению к матрице смежности формой представления орграфов. Заданное списком смежности представление определяет порядок ребер, выходящих из каждой вершины.

Определение. Упорядоченным орграфом называется пара $G = (V, E)$, где V — копечное множество вершин, а E — множество упорядоченных списков ориентированных ребер. Элементы E имеют вид

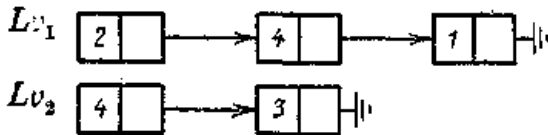
$$L_v = ((v, w_1), \dots, (v, w_k)),$$

где $v, w_i \in V$.

Пример 8. Упорядоченный орграф

$$G = (\{v_1, v_2, v_3, v_4\}, \{((v_1, v_2), (v_1, v_4), (v_1, v_1)), ((v_2, v_4), (v_2, v_3))\})$$

может быть представлен списками смежности



λ может быть изображен диаграммой (рис. 12).

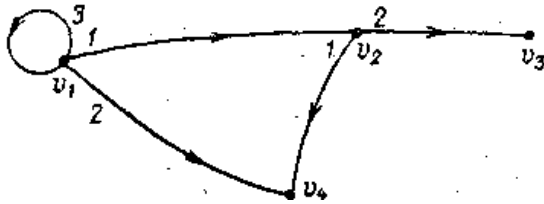


Рис. 12.

Упорядоченный орграф G определяет единственный неупорядоченный орграф; мы только заменяем каждый список $((v, w_1), \dots, (v, w_k))$ множеством $\{(v, w_1), \dots, (v, w_k)\}$. Орграф, определенный таким образом, называют оргграфом, *подчиненным* G . Упорядоченный ациклический орграф является упорядоченным графом, чьим подчиненным оргграфом является ациклический оргграф. Упорядоченное ориентированное дерево является упорядоченным оргграфом, чей подчиненный оргграф является ориентированным деревом.

Пример 9.

$$T = ((v_1, \dots, v_6), \{((v_1, v_2), (v_1, v_3)), ((v_3, v_4), (v_3, v_5), (v_3, v_6))\})$$

является упорядоченным ориентированным деревом, где v_1 — корень. Оно может быть изображено, как показано на рис. 13.

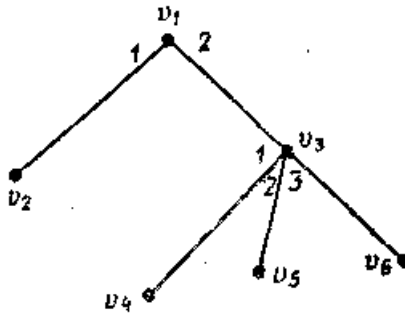


Рис. 13.

Упорядоченные ориентированные деревья будем изображать спуском вершин слева направо (рис. 13).

Если принять такое соглашение, то номера ребер можно опускать.

Определение.

1) Пусть S_V и S_E — множества. *Пометкой* упорядоченного орграфа $G = (V, E)$ называется пара отображений (f, g) , где

$f: V \rightarrow S_V$ — пометка вершин,

$g: E \rightarrow \bigcup_{k=1}^{\infty} S_E^k$ — пометка ребер.

Отображение g имеет вид

$$g(((v, w_1), \dots, (v, w_k))) = (\alpha_1, \dots, \alpha_k) \in S_E^k.$$

2) Говорят, что два помеченных графа $G_1=(V_1, E_1)$ и $G_2=(V_2, E_2)$ с функциями пометок (f_1, g_1) и (f_2, g_2) соответственно эквивалентны, если существует биекция $h: V_1 \rightarrow V_2$ такая, что

а) $((v, w_1), \dots, (v, w_k)) \in E_1$ тогда и только тогда, когда

$$((h(v), h(w_1)), \dots, (h(v), h(w_k))) \in E_2$$

(эквивалентны как упорядоченные графы);

б) $f_1(v) = f_2(h(v))$ для всех $v \in V$ (метки вершин совпадают);

в) для всех $((v, w_1), \dots, (v, w_k)) \in E_1$ имеем

$$\begin{aligned} g_1(((v, w_1), \dots, (v, w_k))) = \\ = g_2(((h(v), h(w_1)), \dots, (h(v), h(w_k)))) \end{aligned}$$

(метки ребер совпадают).

Определим обход орграфа как перестановку или полное упорядочивание вершин. Для упорядоченных орграфов все делается точно так же. Для упорядоченных ориентированных деревьев часто полезны другие обходы. Некоторые из них будут описаны ниже.

Определение. Пусть $T = (\{v_1, \dots, v_n\}, E)$ — упорядоченное ориентированное дерево и $L_v = ((v, w_1), \dots, (v, w_k)) \in E$. Определим отношение $<$ на множестве $\{w_1, \dots, w_k\}$ следующим образом: $w_i < w_j$ тогда и только тогда, когда $i < j$. Определим таким образом отношение $<$ для каждого списка E .

Утверждение. Отношение $<$ является отношением частичного порядка на V .

Доказательство.

Из соотношения $v < v$ следует список вида

$$((q, w_1), \dots, (q, v), \dots, (q, v), \dots, (q, w_k)),$$

который невозможен, так как в дереве не существует циклов.

Следовательно, $v \not< v$ для любого $v \in V$.

Из соотношений $v < w$ и $w < u$ следует, что существуют $x, y \in V$ такие, что

$$\begin{aligned} L_x = ((x, w_1), \dots, (x, v), \dots, (x, w), \dots, (x, w_k)), \\ L_y = ((y, u_1), \dots, (y, w), \dots, (y, u), \dots, (y, u_1)). \end{aligned}$$

Если $x \neq y$, то $\delta(w)=2$, что невозможно, так как T — дерево. Следовательно, $x = y$

$$L_x = \{(x, w_1), \dots, (x, v), \dots, (x, w), \dots, (x, u), \dots, (x, w_k)\},$$

т. е. $v < u$. Поэтому отношение $<$ есть частично упорядоченное отношение на V .

Отношение $<$ сравнивает только вершины, выходящие из одной вершины.

Пример 10. Для упорядоченного ориентированного дерева на рис. 14 имеем

$$< = \{(v_2, v_3), (v_4, v_5), (v_4, v_6), (v_5, v_6)\} \text{ или } v_2 < v_3, v_4 < v_5, v_4 < v_6 \text{ и } v_5 < v_6.$$

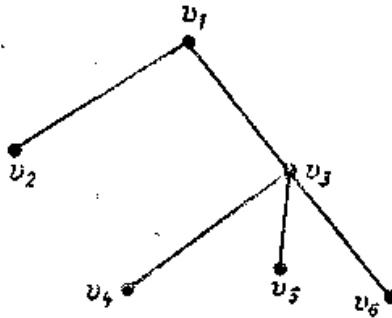


Рис. 14.

Замечание. Обозначим множество всех спусков из вершины $v \in V$ через $\Gamma^+(v)$; аналогично через $\Gamma^-(v)$ обозначим множество входов в v .

Определение. Отношение $<$ называют *трансверсальным порядком* вершин упорядоченного направленного дерева $T=(V, E)$.

Наша цель в оставшейся части раздела — вывести различные полезные методы обхода для дерева с использованием симметрии путем расширения трансверсального порядка. Перед точным определением обходов необходимы два результата. Пусть $<$ определяет отношение $<_e$ на V следующим образом: если $w_i < w_j$, то $w'_i <_e w'_j$ для всех

$w'_i \in \Gamma^+(w_i) \cup \{w_i\}$ и для всех $w'_j \in \Gamma^+(w_j) \cup \{w_j\}$.

Утверждение.

- 1) $< \subseteq <_e$;
- 2) $<_e$ — частично упорядоченное отношение на V .

Доказательство.

1) Утверждение очевидно.

2) а) Из $v <_e v$ следует, что или $v < v$, или существуют $x, y \in V$ такие, что $x < y$ и $v \in \Gamma^+(x) \cup \{x\}$ и $v \in \Gamma^+(y) \cup \{y\}$. Однако $v \not< v$, так как $<$ — отношение частичного порядка. Аналогично из $x < y$ следует

$x \neq y$; следовательно, $v \in \Gamma^+(x) \cup \{x\}$ и $v \in \Gamma^+(y) \cup \{y\}$. Однако это означает, что $\delta^-(v) = 2$ или $\delta^-(w) = 2$ для некоторого $w \in \Gamma^-(v)$. Это невозможно, потому что $T = (V, E)$ — дерево. Таким образом, $v <_e v$ для всех $v \in V$.

б) $v <_e w$ означает, что $v < w$ или что существуют $x, y \in V$ такие, что $x < y$ и $v \in \Gamma^+(x) \cup \{x\}$ и $w \in \Gamma^+(y) \cup \{y\}$; $w <_e u$ означает, что $w < u$ или что существуют $r, s \in V$ такие, что $r < s$ и $w \in \Gamma^+(r) \cup \{r\}$ и $u \in \Gamma^+(s) \cup \{s\}$; $w \in \Gamma^+(r) \cup \{r\}$ и $w \in \Gamma^+(y) \cup \{y\}$ дают, что или $r \in \Gamma^+(y)$, или $y \in \Gamma^+(r)$.

Следовательно, дерево имеет одну из форм, изображенных на рис. 15.

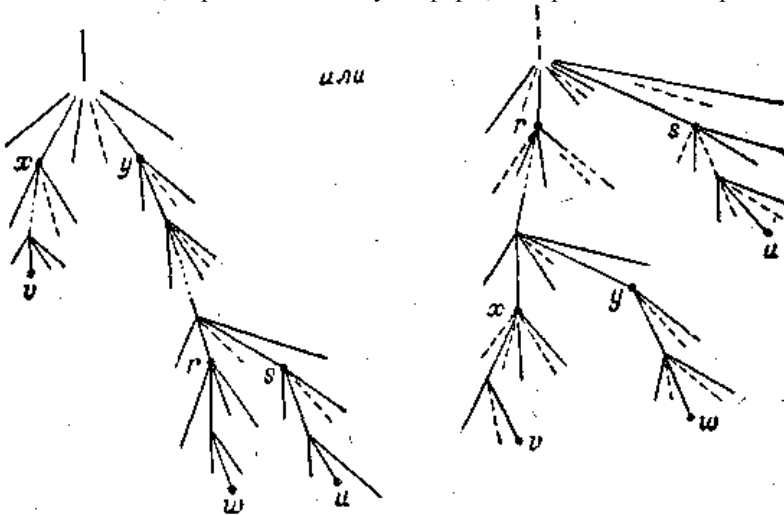


Рис. 15.

Если $r \in \Gamma^+(y)$, то $s \in \Gamma^+(y)$ и $x < y$ дает $x <_e s$. Однако $u \in \Gamma^+(s)$, следовательно, $x <_e u$, и так как $v \in \Gamma^+(x)$, то, следовательно, $v <_e u$. Если $y \in \Gamma^+(r)$, то $x \in \Gamma^+(r)$ и $r < s$ дает $x <_e s$. Однако $u \in \Gamma^+(s) \cup \{s\}$; следовательно, $x <_e u$, а $v \in \Gamma^+(x)$ дает $v <_e u$.

Следовательно, $<_e$ является отношением частичного порядка на V .

Иногда $v <_e w$ читают как « v находится слева от w ».

Утверждение. Пусть $T = (V, E)$ — упорядоченное ориентированное дерево. Тогда для $v_i, v_j \in V$ ($i \neq j$) или $v_i <_e v_j$ или $v_j <_e v_i$, или же v_i и v_j находятся на маршруте.

Доказательство. Пусть $v_i, v_j \in V$ и $v_i \neq v_j$. Тогда существует вершина v_a , такая, что $v_i \in \Gamma^+(v_a) \cup \{v_a\}$ и $v_j \in \Gamma^+(v_a) \cup \{v_a\}$. Если $v_i = v_a$ или

$v_j = v_a$ то v_i и v_j находятся на маршруте; в противном случае рассмотрим прямые спуски w_1, \dots, w_k из v_a . Тогда или

$$v_i \in \Gamma^+(w_1) \text{ и } v_j \in \Gamma^+(w_m) \text{ для } w_1 \neq w_m,$$

или

$$v_i, v_j \in \Gamma^+(w_h).$$

В первом случае имеем или $v_i <_e v_j$ или $v_j <_e v_i$ в зависимости от того, $w_1 < w_m$ или $w_m < w_1$. Во втором случае повторяем процесс из w_k , пока не будут выполнены условия первого случая, или получаем $v_i = v_a$ или $v_j = v_a$. В этом случае v_i и v_j находятся на маршруте.

Много полезных обходов дерева определяют посредством расширения отношения $<_e$, до полного упорядочивания V . Используя приведенный выше результат, надо только расширить $<_e$ чтобы сравнивать вершины, которые находятся на маршруте, для определения полного порядка на V , для которого $<_e$ является подпорядком.

Определение. Пусть $T=(V, E)$ - упорядоченное направленное дерево. Определим полную упорядоченность $<_1$ на V следующим образом: если v_j спускается из v_i , то $v_i <_1 v_j$; в противном случае $v_i <_2 v_j$, если $v_i <_e v_j$. Отношение $<_1$ называют *предпорядком* на V .

Очевидно, что $<_e \subseteq <_1$.

Пример 11. Пусть $T = (V, E)$ — упорядоченное направленное дерево, изображенное на рис. 16.

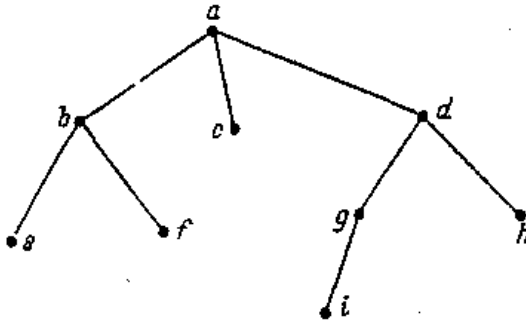


Рис. 16

Тогда предпорядок на $V = \{a, b, c, d, e, f, g, h, i\}$ можно записать как

$$a <_1 b <_1 e <_1 f <_1 c <_1 d <_1 g <_1 i <_1 h.$$

Соответствующий обход вершины имеет вид $a, b, e, f, c, d, g, i, h$.

Определение. Пусть $T = (V, E)$ — упорядоченное ориентированное дерево. Определим отношение $<_2$ полного порядка на V следующим образом: если v_i спускается из v_j , то $v_i <_2 v_j$; в противном случае $v_i <_2 v_j$, если $v_i <_e v_j$. Отношение $<_2$ называют *постпорядком* на V .

Очевидно, что $\prec_e \subseteq \prec_2$.

Пример 12. Пусть T и V такие же, как и в предыдущем примере. Тогда постпорядок на V будет иметь вид

$$e \prec_2 f \prec_2 b \prec_2 c \prec_2 i \prec_2 g \prec_2 h \prec_2 d \prec_2 a$$

с обходом

$$e, f, b, c, i, g, h, d, a.$$

Определение. Пусть $T = (V, E)$ — полное бинарное дерево. Определим симметричный порядок \prec_s на V следующим образом: для каждой вершины, не являющейся листом и имеющей прямые спуски w_1 и w_2 ($w_1 < w_2$), положим

$$w'_1 \prec_s v \text{ для всех } w'_1 \in \Gamma^+(w_1) \cup \{w_1\},$$

$$v \prec_s w'_2 \text{ для всех } w'_2 \in \Gamma^+(w_2) \cup \{w_2\}.$$

Порядок \prec_s также является расширением \prec_e .

Пример 13. Пусть T — дерево, изображенное на рис. 17. Тогда \prec_s определено, как показано на рис 17.

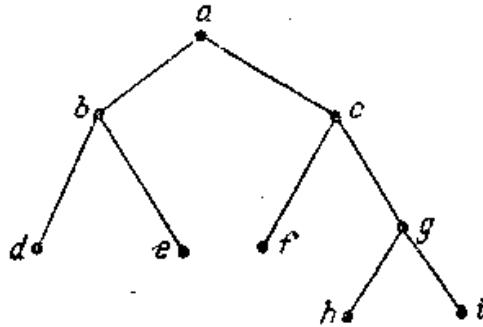


Рис. 17.

Обход по предпорядку упорядоченного ориентированного дерева соответствует обходу по глубине с корнем \prec_s качестве начальной вершины.

Если $T = (V, E)$ — упорядоченное ориентированное дерево, то вершины T могут быть расположены в предпорядок с применением следующего алгоритма, начинающегося с корня:

```

procedure  $pre(v)$ 
process  $v$ 
if  $L_v \neq \emptyset$  do  $pre(w)$  for each  $w \in L_v$ 
endproc
  
```

Соответствующий алгоритм для постпорядка имеет вид

```

procedure  $post(v)$ 
  
```

```

if  $L_v = \emptyset$  then process  $v$ 
    else begin
        post( $w$ ) for each  $w \in L_v$ 
        process  $v$ 
    end

```

Определение. Поддеревом $T' = (V', E')$ ориентированного дерева $T=(V, E)$ называется такое ориентированное дерево, что:

- а) $\emptyset = V' \subseteq V$;
- б) $E' = V' \times V' \cap E$;
- в) ни одна из вершин $\forall V'$ не является спуском вершины в V' .

Определение. Если T — ориентированное дерево, то *уровень* вершины определяют как максимальную длину маршрута от этой вершины до листа. *Глубина* вершины — это длина пути от корня до этой вершины. *Глубиной* T называют длину самого длинного маршрута в T . *Высотой вершины* T называют глубину T за вычетом глубины вершины. *Высота* T является высотой корня.

1.5. Отношение включения и достижимые множества

Исследуем некоторые основные свойства *ориентированных графов*. Пусть G — такой граф с множеством вершин V . Если a и b — некоторые вершины, для которых существует ориентированная цепь $P(a, b)$ от a к b , то будем говорить, что вершина b *достижима* из a , и писать $a > b$. Как и для неориентированных графов, очевидно, что ориентированную цепь $P(a, b)$ можно свести к ориентированной простой цепи $P_I(a, b)$, в которой каждая вершина встречается лишь один раз.

Теорема 1. Если b достижима из a и c достижима из b , то c достижима из a .

Доказательство. Пусть $P(a, b)$ и $Q(b, c)$ — ориентированные простые цепи и d — первая вершина в P , принадлежащая также Q . Тогда

$$P(a, d) \cup Q(d, c)$$

является ориентированной простой цепью из a к c .

Бинарное отношение $a > b$, которое таким образом определяется, будем называть *отношением включения*, соответствующим графу G . Теорема 1 показывает, что это отношение транзитивно. Обратное отношение обозначается через $b < a$.

Обозначим соответственно через

$$D(a), \quad D^*(a)$$

множества всех достижимых и обратно достижимых вершин из вершины a . Очевидно, $a > b$ тогда и только тогда, когда

$$D(a) \supseteq D(b).$$

Две вершины a и b называем *эквивалентными по достижимости*, если

$$D(a) = D(b), \tag{1}$$

т. е. если

$$a > b, \quad b > a \text{ или просто } a \sim b.$$

Для выполнения (1) необходимо и достаточно, чтобы вершины a и b были в G *взаимно связаны* (*сильно связаны*, *бисвязаны*), т. е. чтобы в G существовали цепи $P(a, b)$ и $Q(b, a)$.

Теорема 2. *Две вершины эквивалентны по достижимости тогда и только тогда, когда они взаимно связаны.*

Вообще, для любого множества A можно определить достижимое множество

$$D(A) = \bigcup_{a \in A} D(a)$$

как множество всех вершин, достижимых из тех или иных $a \in A$. Два множества A и B эквивалентны по достижимости, если

$$DU = D(B).$$

Это может случиться только тогда, когда каждое $b \in B$ достижимо из некоторого $a \in A$, и наоборот.

Множество A , из которого достижима каждая вершина, т. е.

$$D(A) = V, \tag{2}$$

можно назвать *порождающим множеством*. *Минимальное порождающее множество* есть минимальное среди множеств A , обладающих свойством (2); такие минимальные множества могут как существовать, так и не существовать.

Мы будем рассматривать некоторые вопросы, аналогичные изученным для неориентированных графов и связанные с листами и блоками. Ребро в G называем *ориентированно-циклическим ребром*, если оно принадлежит некоторому ориентированному циклу, и *ациклическим* в противном случае. Две вершины a_0 и b_0 *ориентированно-циклически-реберно связаны*, если существует такая последовательность ориентированных простых циклов

$$C_1, C_2, \dots, C_k, \tag{3}$$

что a_0 принадлежит C_1 , b_0 принадлежит C_k и любая пара соседних циклов C_i и C_{i+1} имеет хотя бы одну общую вершину.

Теорема 3. *Две вершины ориентированно-циклически-реберно связаны тогда и только тогда, когда они взаимно связаны.*

Доказательство. Предположим сначала, что a_0 и b_0 ориентированно-циклически-реберно связаны. Если в (3) $k=1$, то утверждение очевидно.

Поэтому можно провести доказательство индукцией по числу k ориентированных простых циклов. Пусть c — такая ближайшая к b_0 вершина на C_k , что c принадлежит одному из предыдущих простых циклов, и существует ориентированный участок $C_k(c, b_0)$ от c до b_0 . По индуктивному предположению существует ориентированная цепь из a_0 к c , и она может быть продолжена до b_0 . Аналогично находим ориентированную цепь от b_0 к a_0 .

Пусть теперь a_0 и b_0 взаимно связаны ориентированными простыми цепями $P(a_0, b_0)$ и $Q(b_0, a_0)$. Если $a_1 \neq a_0$ есть первая вершина на P , принадлежащая Q , то существует ориентированный простой цикл

$$C_1 = P(a_0, b_0) \cup Q(a_1, a_0),$$

и вершины a_0 и b_0 взаимно связаны простыми цепями

$$P(a_1, b_0), \quad Q(b_0, a_1).$$

Повторяя такую операцию, мы получим последовательность (3) ориентированных простых циклов, связывающую a_0 и b_0 .

Из этих рассуждений вытекает также следующее.

Теорема 4. *Две вершины a_0 и b_0 в ориентированном графе G взаимно связаны тогда и только тогда, когда они связаны маршрутом из ориентированно-циклических ребер.*

Ориентированный граф *бисвязен* (взаимно связей, сильно связан), если все его вершины взаимно связаны. Такой граф характеризуется также тем свойством, что каждая его вершина является порождающим множеством.

Теорема 5. *Ориентированный граф бисвязен тогда и только тогда, когда он связан и не имеет ациклических ребер.*

Доказательство. Бисвязный граф G , очевидно, имеет связный соотнесенный неориентированный граф G_u . Никакое ребро

$E_0 = (a_0, b_0)$ в G не может быть ациклическим, так как существует ориентированная простая цепь $Q(b_0, a_0)$, которая вместе с E_0 дает ориентированный простой цикл, содержащий E_0 . С другой стороны, если граф G_u связан и все ребра в G ориентированно-циклические, то из теоремы 4 следует, что граф G бисвязен.

Можно поставить вопрос: при каких условиях данный неориентированный граф G_0 можно рассматривать как соотнесенный неориентированный граф G_u для бисвязного графа G ? Этот вопрос можно сформулировать как *задачу о движении транспорта*. Граф G_0 рассматривается как план города или местности, а его ребра — как улицы или дороги. В каких случаях можно ввести направления движения на всех улицах так, чтобы любые два пункта были соединены в обоих направлениях дорогами с односторонним движением? Эту задачу решил Роббинс для конечных графов, а затем Эджид для произвольных графов. Результат состоит в следующем.

Теорема 6. Неориентированный граф G_0 является графом G_u бисвязного ориентированного графа G тогда и только тогда, когда граф G_0 связан и не имеет разделяющих ребер.

Доказательство. Разделяющие ребра соответствуют в нашей интерпретации единственным мостам через реку или тупикам. Поэтому предположим, что в связном графе G_0 нет разделяющих ребер. Можно построить бисвязные ориентированные части графа, вводя подходящие направления; например, из простого цикла возникают два таких графа. Эти ориентированные части графа образуют частично упорядоченное множество, если считать $H_1 \supset H_2$, когда ребра H_2 принадлежат H_1 и имеют те же направления. Рассмотрим упорядоченное по включению семейство $\{ H_i \}$ бисвязных частей. Их сумма

$$H = \bigcup H_i$$

также бисвязна; следовательно, по принципу суммы цепи существуют максимальные бисвязные части H .

Покажем теперь, что такая максимальная часть H содержит все вершины G_0 . Если бы это было не так, то нашлось бы некоторое неориентированное ребро $E = (b, c)$, где b принадлежит H , а c не принадлежит H . В G_0 существует простой цикл C , содержащий E . Обозначим через $C(b, c, d)$ участок C от b через E до следующей вершины d из H . В H существуют ориентированные цепи $P(c, d)$ и $Q(d, c)$, и, выбирая любое из двух возможных направлений на $C(b, c, d)$, мы получим больший бисвязный граф H' , что противоречит максимальнойности H . Наконец, проверяем, что в максимальном H содержатся все ребра G_0 .

В связи с задачей о движении транспорта упомянем более трудный вопрос: в каких случаях можно так ввести направления на некоторых или на всех улицах города, чтобы любые два перекрестка были взаимно связаны ориентированными дорогами, не имеющими пересечений? Другими словами, когда можно ввести направления на ребрах неориентированного графа G_0 так, чтобы каждая пара вершин принадлежала ориентированному простому циклу? Очевидно, это возможно, если G_0 имеет гамильтонов цикл.

Еще более ограничительным является условие, чтобы любые два дома в городе соединялись непересекающимися ориентированными дорогами. Тогда граф должен быть ориентирован так, чтобы любые два ребра принадлежали ориентированному простому циклу.

В связи с ориентированными цепями приведем следующий результат, который получил Редди при изучении квадратичных полей.

Теорема 7. Пусть G — конечный ориентированный граф, в котором каждая пара вершин соединена ребром. Тогда существует

ориентированная простая цепь, проходящая через все вершины графа.

Доказательство. Теорема справедлива для случая, когда число вершин $n = 2$. Пусть G_n есть граф с n вершинами, для которого теорема верна; построим граф G_{n+1} , добавив некоторую вершину v_{n+1} , в которой имеются ребра ко всем вершинам v_i из G_n . По предположению в G_n существует ориентированная простая цепь

$$P_n = (v_1, v_2) \dots (v_{n-1}, v_n)$$

Если есть ребро $E = (v_n, v_{n+1})$ или (v_{n+1}, v_n) , то эта простая цепь может быть продолжена через v_{n+1} . Поэтому можно считать, что в v_{n+1} есть и входящие, и выходящие ребра. Допустим, что все ребра E_1, \dots, E_k входящие, а ребро E_{k+1} выходящее. Тогда

$$P_{n+1} =$$

$$= (v_1, v_2) \dots (v_{k-1}, v_k) (v_k, v_{n+1}) (v_{n+1}, v_{k+1}) \dots (v_{n-1}, v_n)$$

является искомой простой цепью.

1.6. Теорема о гомоморфизме

В ориентированном графе G обозначим через $L(a_0)$ множество всех вершин, которые взаимно связаны с данной вершиной a_0 . Назовем $L(a_0)$ (*ориентированным*) *листовым множеством*, которому принадлежит a_0 . Оно состоит более чем из одной вершины, если a_0 лежит на ориентированном простом цикле, не являющемся петлей. Подграф $G(L)$ назовем (*ориентированным*) *листом*, определяемым листовым множеством L . Из предыдущих рассуждений следует, что все ребра в $G(L)$ ориентированно-циклические и что листы являются максимальными бисвязными частями графа G . Граф $G(L)$ *ориентированно-циклически замкнут* в том смысле, что если C — любой ориентированный простой цикл, имеющий общую с L вершину, то весь цикл C содержится в $G(L)$. Отсюда получаем теорему:

Теорема 8. *Все ребра, касающиеся листа $G(L)$, ациклические.*

Назовем ориентированный граф *ациклическим*, если все его ребра ациклические, т. е. если G не имеет ориентированных циклов. Для ориентированных графов ациклические графы играют роль, подобную роли деревьев для неориентированных графов. Например, справедлива

Теорема 9. *Любой ориентированный граф G имеет максимальные ациклические части H , обладающие хотя бы одним входящим или выходящим ребром в каждой неизолированной вершине G . Если граф G связан, то граф H также связан.*

Для ориентированного графа G можно построить ориентированный граф G' , вершинами которого являются листовые множества G . Два листовых множества L_1 и L_2 соединяются ориентированным ребром (L_1, L_2) в G' если имеются ориентированные ребра от L_1 к L_2 в G . Этот граф называется *листовой композицией* графа G . G' является ациклическим графом, так как любой ориентированный цикл в нем дал бы ориентированный цикл в G , проходящий через несколько листов. Таким образом, мы можем установить *теорему о гомоморфизме* для ориентированных графов, аналогичную теореме для неориентированных графов:

Теорема 10. *Ориентированный граф G имеет ациклический граф листовой композиции G' , и G гомоморфен G' при отображении τ , для которого множества образов элементов*

$$\tau^{-1}(a'_0) = L(a_0)$$

являются ориентированными листовыми множествами графа G .

В ориентированном графе можно также искать аналог блокам, введенным для неориентированных графов. Там они были получены при помощи отношения эквивалентности для ребер графа, которое основывалось на понятии сильной циклической связности. Для ориентированного графа будем говорить, что два ребра E_1 и E_2 в G *сильно ориентированно-циклически-реберно связаны*, если существует такая последовательность ориентированных простых циклов (3) , что E_1 лежит на C_1 , E_2 лежит на C_2 и любая пара соседних циклов C_i, C_{i+1} имеет по крайней мере одно общее ребро. Соответственно назовем ориентированный граф G *сильно ориентированно-циклически-реберно связным*, если все ребра сильно ориентированно-циклически-реберно связаны. Тогда справедлив следующий аналог теоремы 6.

Теорема 11. *Неориентированный граф G_0 является графом G_u для сильно ориентированно-циклически-реберно связного ориентированного графа G тогда и только тогда, когда G_0 не имеет разделяющих вершин.*

Доказательство подобно доказательству теоремы 6 и предоставляется читателю.

Далее, пусть E_0 - некоторое ребро в ориентированном графе G . Часть графа, состоящая из всех ребер, сильно ориентированно-циклически-реберно связанных с E_0 , назовем (*ориентированным*) *блоком*, которому принадлежит E_0 . Его множество вершин $M(E_0)$ есть *блоковое множество*, определенное ребром E_0 , и блок является, очевидно, подграфом $G(M)$. Можно проверить, что его соотнесенный неориентированный граф $G_u(M)$ не имеет разделяющихся вершин. Блок *сильно ориентированно-циклически замкнут*, т. е. любой

ориентированный простой цикл C , имеющий хотя бы две общие с $M(E_0)$ вершины, содержится целиком в $G(M)$.

1.7. Транзитивные графы и погружения в отношения упорядочения

Ориентированный граф G называется *транзитивным*, если для любых двух его ребер

$$E_0 = (a_0, a_1), \quad E_1 = (a_1, a_2)$$

существует замыкающее ребро $E_2 = (a_0, a_2)$ (и вообще, если $a > b$, то существует ребро $E = (a, b)$). Каждый ориентированный граф G имеет транзитивное замыкание, т. о. наименьший содержащий его транзитивный граф G_t . Граф G_t получается из ориентированных цепей $P(a, b)$ в G добавлением ребра $E = (a, b)$, когда $a > b$ и такого ребра нет в G .

Если G — бисвязный граф, то $a > b$ и $b > a$ для любой пары вершин, и, следовательно, в G будут все ребра (a, b) и (b, a) . Поэтому если G имеет однократные ребра (и петли), то его транзитивное замыкание будет ориентированным полным графом $U^{(d)}$.

Если граф G ациклический, то добавление ребер $E = (a, b)$, $a > b$, не может дать ориентированного цикла в G_t , так как он соответствовал бы некоторому циклическому ориентированному маршруту в G . Согласно определению частичного упорядочения мы видим, что G_t является графом частичного упорядочения, при условии, что он не имеет кратных ребер.

Из п. 1.6 получаем следующие общие утверждения.

Теорема 12. Пусть G — ориентированный граф с однократными ребрами. Транзитивное замыкание G_t графа G является графом, у которого листы суть ориентированные полные графы $U^{(d)}(L)$, определенные на листовых множествах L графа G , а листовая композиция G' для G_t является графом частичного упорядочения, изоморфным транзитивному замыканию листовой композиции G' для G .

Ориентированный граф будет называться *квазиупорядочением*, если его транзитивное замыкание является упорядоченным графом (т.е. графом упорядочения). Из определения упорядоченного множества видно, что квазиупорядочение характеризуется двумя свойствами:

1. Граф G ациклический: из $a > b$, $b > a$ следует $a = b$.
2. Одно из соотношений $a > b$ или $b > a$ всегда выполняется для любой пары вершин.

Представляет интерес также несколько менее ограничительное понятие упорядоченности. Ориентированный граф O назовем *слабым упорядочением*, если он удовлетворяет условию: для любой пары вершин выполняется хотя бы одно из соотношений $a > b$, $b > a$. Листовая композиция для слабо упорядоченного множества будет квазиупорядочением.

Теорема 13. *Любой ориентированный граф G является частью некоторого слабого упорядочения O с теми же листами, что и G .*

Будем говорить, что две вершины в G *сравнимы*, если выполняется хотя бы одно из соотношений $a > b$, $b > a$;

если ни одно из них не выполнено, то вершины a и b называются *несравнимыми*. Таким образом, граф является слабым упорядочением тогда и только тогда, когда все его вершины сравнимы.

Чтобы доказать теорему 13, докажем сначала вспомогательное утверждение:

Теорема 14. *Если a и b — несравнимые вершины в графе G и если присоединить к G ребро $E = (a, b)$, то $G \cup E$ будут иметь одни и те же ориентированно-циклические ребра.*

Доказательство. Во-первых, новое ребро E не может быть ориентированно-циклическим в $G \cup E$, так как иначе существовала бы ориентированная цепь $P(b, a)$ в G . Во-вторых, если $F = (c, d)$ есть ациклическое ребро в G , то оно не может стать ориентированно-циклическим в $G \cup E$, так как иначе ориентированной цепи

$$Q(d, a) \cup E \cup Q_1(b, c)$$

отвечала бы ориентированная цепь

$$Q_1(b, c) \cup E \cup Q(d, a)$$

связывающая b с a .

Чтобы теперь получить теорему 13, рассмотрим семейство всех графов $H \supset G$, которые имеют те же ориентированно-циклические ребра, что и граф G . Пусть $\{H_i\}$ — упорядоченное по включению семейство таких графов и

$$H_0 = \bigcup H_i$$

— их сумма. Тогда H_i есть граф того же типа, так как если бы H_0 содержал ориентированный цикл C , не содержащийся в G , то ребра из C должны были бы принадлежать одному из графов H_i . Из принципа максимальной заключаем, что должны существовать максимальные графы $H_0 \supset G$ с теми же ориентированно-циклическими ребрами. Из теоремы 14 следует, что в H_0 все вершины должны быть сравнимы; следовательно, H_0 является слабым упорядочением.

Вместо присоединения к графу G ребра $E = (a, b)$ в теореме 14, можно использовать и обратное ребро $E' = (b, a)$. Тогда мы получаем, что для любого ребра E между несравнимыми вершинами a и b существует слабоупорядоченный граф $O \supset G$, удовлетворяющий условиям теоремы 14 и не содержащий E . Это рассуждение дает следующее представление для ориентированных графов:

Теорема 15. Любой ориентированный граф G может быть представлен как пересечение слабых упорядочений $O_i \supset G$, имеющих те же ориентированно-циклические ребра, что и G .

В случае ациклического графа получается

Теорема 16. Каждый ациклический граф есть пересечение квазиупорядочений, в которых он содержится.

1.8. Базисные графы

Мы будем изучать достижимость вершин ориентированными цепями. Можно предполагать, что все ребра в графе однократные и петель нет. В п. 1.7 мы строили транзитивное замыкание ориентированного графа G , добавляя ребро $E = (a_0, a_n)$, если существовала ориентированная цепь $P(a_0, a_n)$. Теперь рассмотрим противоположную операцию, состоящую в удалении излишних ребер $E = (a_0, a_n)$, т. е. таких ребер E , для которых существует не содержащая E ориентированная цепь $P(a_0, a_n)$. Такая операция не может изменить достижимые множества ни для какой вершины.

Часть H графа G называется порождающей частью, если достижимые множества для всех вершин одни и те же для H и для G :

$$D_G(a) = D_H(a), \quad a \in V. \quad (4)$$

Для того чтобы граф H обладал этим свойством, необходимо и достаточно, чтобы для каждого ребра $E = (a, b)$ в G существовала ориентированная цепь $P_H(a, b)$ в H . Условие (4) приводит нас к следующему утверждению:

Теорема 17. Любая порождающая часть H графа G имеет те же листовые множества, что и G , и листовая композиция H' для H является порождающей частью для G' .

Минимальный порождающий граф для G назовем базисным графом. Таким образом, порождающий граф H будет базисным графом тогда и только тогда, когда удаление любого ребра из H изменяет достижимость в H , т. е. если $E = (a, b)$ — произвольное ребро в H , то оно не может быть излишним в том смысле, что существует некоторая ориентированная цепь $P_H(a, b)$ в H , не содержащая E .

Если G — конечный граф, то существуют базисные графы, и они могут быть получены при последовательном удалении ребер, как это указано

выше. Если G имеет ориентированный гамильтонов цикл, то он может быть взят в качестве базисного графа. Если существует базисный граф, он не обязательно единственный. На рис. 18 любое радиальное ребро и ориентированный многоугольный цикл определяют базисный граф.

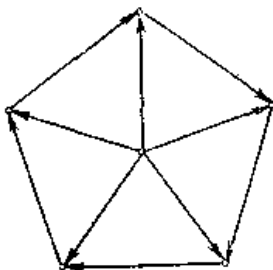


Рис. 18.

На рис. 19 пять ребер $(1, 0)$, $(3, 0)$, $(5, 0)$, $(0, 2)$, $(0, 4)$ составляют единственный базисный граф.

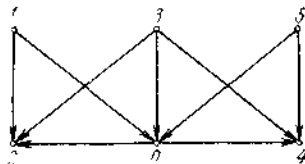


Рис. 19.

Заметим, что граф, полученный после удаления ребер $(0,2)$ и $(0,4)$, имеет единственный базис, состоящий из семи ребер.

В качестве примера можно также взять ориентированный полный граф $U_0^{(d)}$, определенный на множестве вершин из n элементов. Ориентированные гамильтоновы циклы дают один тип базисных графов; каждый имеет n ребер. Другой тип дают звездные графы $S(a_0)$, состоящие из всех выходящих из a_0 и входящих в вершину a_0 ребер; эти графы имеют $2(n-1)$ ребер.

Эта конструкция является частным случаем следующего результата.

Теорема 18. Пусть G_d — ориентированный граф, полученный удвоением связного неориентированного графа G . Тогда удвоением любого максимального дерева в G можно получить базисный граф для G_d .

Доказательство предоставляется читателю.

В конечном упорядоченном множестве

$$a_0 < a_1 < \dots < a_n$$

базисный граф единственный и состоит из всех ребер $E_b = (a_i, a_{i+1})$. Если бесконечное упорядоченное множество будет иметь базисный граф B , то, очевидно, для любого ребра $E = (a, b)$ в B вершины a и b должны быть непосредственно следующими друг за другом в смысле этого упорядочения. Кроме того, никакие две вершины не могут иметь бесконечного числа лежащих между ними вершин. Отсюда мы получаем следующую теорему.

Теорема 19. *Упорядоченное множество может иметь базисный граф только тогда, когда оно изоморфно естественно упорядоченному множеству целых чисел.*

Сформулируем некоторые общие условия, при которых существуют базисные графы. Будем говорить, что ориентированный граф G имеет *компактное реберное разделение*, если он удовлетворяет следующему условию: для любого ребра $E = (a_0, b_0)$ пусть H есть такая часть (содержащая E), что после удаления ребер из H вершина b_0 перестает быть достижимой из a_0 ; тогда существует конечная часть H_0 графа H с тем же свойством.

Теорема 20. *Ориентированный граф с компактным реберным разделением имеет базисный граф.*

Доказательство. Чтобы показать, что существуют минимальные порождающие графы, рассмотрим упорядоченное семейство $\{K^{(i)}\}$ таких графов и установим, что их пересечение

$$K_0 = \bigcap_i K^{(i)} \quad (5)$$

также является порождающим графом. Это имеет место тогда и только тогда, когда для каждого ребра $E = (a_0, b_0)$ в G вершина b_0 достижима из a_0 в K_0 . Так как G имеет компактное реберное разделение, для каждого E существует такое семейство конечных частей $\{H^{(j)}_0(E)\}$, что b_0 достижима из a_0 в части K , только если дополнение \bar{K} не содержит никакого из этих графов $H^{(j)}_0$. Если бы часть K_0 не содержала никаких ребер некоторого $H^{(j)}_0$, то в (5) был бы граф $K^{(j)}$ с тем же свойством, в противоречии с предположением, что все $K^{(i)}$ являются порождающими графами.

Как частный случай теоремы 20 получается

Теорема 21. *Если в G любые два конца a_0 и b_0 ребра связаны конечным числом ориентированных цепей, то G имеет базисный граф.*

Эти условия выполняются, если граф G локально конечен и цепи от a_0 к b_0 имеют ограниченные длины.

Установим также другой результат:

Теорема 22. *Ориентированный граф G имеет базисный граф тогда и только тогда, когда каждый его лист $G(L)$ и его листовая композиция G' имеют базисные графы.*

Доказательство. Предположим сначала, что G имеет базисный граф B . Для любой вершины a_0 в листовом множестве L каждая другая вершина $b_0 \in L$ достижима из a_0 только цепями, лежащими целиком в $G(L)$. Следовательно, в B вершина b_0 достижима из a_0 в подграфе $B(L)$. Никакое ребро в $B(L)$ не может быть излишним, так как иначе оно приводило бы к уменьшению B . В B существует только одно ребро, соединяющее пару листовых множеств L_1 и L_2 в G (и в B), так как если одно такое ребро содержится в B , то другие, очевидно, являются излишними. Ребра из B' образуют порождающий граф для G' , и никакие ребра из B' не могут быть излишними, так как это снова приводило бы к уменьшению B .

Обратно, предположим, что существует базисный граф $B(L)$ для каждого листового множества L в G , а также базисный граф для листовой композиции G' . Каждое ребро в G' является образом всех ребер из G , связывающих пару листовых множеств. Каждому ребру в G' поставим в соответствие одно такое ребро из G . Тогда легко видеть, что такие ребра, отвечающие ребрам из B' , вместе со всеми ребрами базисных графов $B(L)$ образуют порождающий граф для G , и он действительно является базисным.

Теорема 22 сводит проблему существования базисного графа к случаю ациклических графов и бисвязных графов.

В заключение сделаем несколько замечаний о том случае, когда базисный граф единственный. Ребро E графа G назовем *существенным*, если оно принадлежит каждому порождающему графу для G . Очевидно, это равносильно тому, что E не является излишним в G . Таким образом, существенные ребра принадлежат каждому базисному графу, если таковые существуют.

Теорема 23. *Пусть G — такой граф, что каждый его порождающий граф содержит базисный граф для G . Для того чтобы G имел единственный базисный граф, необходимо и достаточно, чтобы для каждого ребра $E = (a, b)$ существовала цепь $P(a, b)$, состоящая из существенных ребер.*

Доказательство. Согласно предыдущему единственный базисный граф существует тогда и только тогда, когда существенные ребра образуют базисный граф, а это имеет место, только если выполняется условие теоремы. Теорема 22 применима, в частности, к конечным графам.

1.9. Чередующиеся цепи

Цепь

$$P = (a_0, a_1)(a_1, a_2) \dots (a_{n-1}, a_n) \quad (6)$$

в ориентированном графе G называется *чередующейся*, если ребра $E_i = (a_i, a_{i+1})$ поочередно принадлежат G и его обратному графу G^* . Такие цепи можно классифицировать в соответствии с характером начального ребра E_0 : она будет α -цепью, если E_0 есть α -ребро, т. е. принадлежит G , и α^* -цепью, если E_0 есть α^* -ребро, т. е. принадлежит G^* . Часто бывает желательно принимать во внимание также характер $\beta = \alpha$ или $\beta = \alpha^*$ последнего ребра E_{n-1} . Например, P есть (α, α^*) -цепь, если E_0 принадлежит G и E_{n-1} принадлежит G^* . Мы получаем четыре типа цепей, характеризуемых символами

$$(\beta, \gamma) = (\alpha, \alpha^*); (\alpha^*, \alpha); (\alpha, \alpha); (\alpha^*, \alpha^*).$$

Цепи двух первых типов имеют четную длину, двух других — нечетную. Чередующаяся цепь будет циклом, если $a_0 = a_n$. Понятие чередующихся цепей можно распространить также на односторонне- и двусторонне-бесконечные цепи. Если P в (6) есть (β, γ) -цепь, то будем говорить, что a_n (β, γ) -достижима из a_0 .

Теорема 24. Если вершина b (β, γ) -достижима из a и вершина c (γ^*, δ) -достижима из b , то c (β, δ) -достижима из a .

Доказательство. Если две связывающие цепи $P(a, b)$ и $Q(b, c)$ не имеют общих ребер, то они могут быть объединены в одну (β, δ) -цепь

$$P(a, b) \cup Q(b, c).$$

Если они имеют общие ребра, то пусть $E = (e, f)$ — первое такое ребро в P . Тогда (β, δ) -цепью будет одна из цепей

$$P(a, e, f) \cup Q(f, c), \quad P(a, e) \cup Q(e, c),$$

в зависимости от того, в одном или в противоположных направлениях входит E в P и в Q .

Множество всех вершин, (β, γ) -достижимых из вершины a , обозначим через

$$V(a, \beta, \gamma). \quad (7)$$

Будем включать a в это множество при $\beta = \gamma$; в вырожденном случае, когда $\rho(a) = 0$, мы имеем

$$V(a, \alpha, \alpha^*) = a.$$

Это определение достижимых множеств можно распространить на произвольные множества A , полагая

$$V = (A, \beta, \gamma) = \bigcup_{a \in A} V(a, \beta, \gamma). \quad (8).$$

Как и ранее, обозначим через $G[A]$ множество конечных вершин тех ребер, которые имеют начальную вершину в A .

Теорема 25. *Множество A является (α, α^*) -достижимым множеством тогда и только тогда, когда все входящие в $G[A]$ ребра имеют свою начальную вершину в A , т. е.*

$$G^*[G[A]] \subseteq A, \quad (9)$$

Доказательство. Очевидно, что из этого свойства следует

$$A = V(A, \alpha, \alpha^*). \quad (10)$$

С другой стороны, если бы имелось некоторое ребро

$$F = (c, b), \quad b \in G[A], \quad c \notin A,$$

то можно было бы построить (α, α^*) -цепь $E \cup F^*$ от A к c ; следовательно, (10) не удовлетворялось бы.

Из теоремы 25, как частный случай, получается

Теорема 26. *Конечное множество A с конечными локальными степенями $\rho(a)$ является (α, α^*) -достижимым множеством тогда и только тогда, когда*

$$\rho(A) = \rho^*(G[A]).$$

Далее может быть доказана

Теорема 27. *Достижимые множества (10) образуют два полных поля множеств.*

Доказательство. Определение (8) показывает, что любая сумма множеств $V(A)$ будет множеством того же типа. Если

$$d \in D = \cap V_i$$

есть вершина, принадлежащая пересечению достижимых множеств, то $V(d)$ также должно содержаться в D . Наконец, нужно показать, что разность

$$V_1 = V(A) - V(B), \quad V(A) \supset V(B),$$

является достижимым множеством. Для любого ребра

$$E = (a, a'), \quad a \in V_1,$$

должно быть

$$a' \in G[V(A)],$$

так что для любого другого ребра $F = (a'', a')$, входящего в a , будет также $a'' \in V(a)$. Но $a'' \in V(B)$ не может быть, так как цепь $F \cup E$ приводила бы от a'' к вершине a , но принадлежащей $V(B)$.

Две вершины a и b называются *эквивалентными по α -достижимости*, если любая вершина v , (α, β) -достижимая из a ($\beta = \alpha$ или $\beta = \alpha^*$), будет также (α, β) -достижимой из b , и наоборот. Будем считать, что a эквивалентна по α -достижимости сама себе.

Теорема 28. *Достижимое множество $V(a)$ состоит из всех вершин, эквивалентных a по α -достижимости.*

Доказательство. Заметим сначала, что если $b \in V(a)$, то существует (α, α^*) -цепь $P(a, b)$ и обратная (α, α^*) -цепь $P(b, a)$; следовательно, a и b должны быть эквивалентны по α -достижимости согласно теореме 24. Предположим теперь, что a и b эквивалентны по α -достижимости. Так как a должна быть α -достижима из b , существует α -цепь $P(b, a)$. Если P есть (α, α^*) -цепь, то все доказано. Предположим поэтому, что P есть (α, α) -цепь. Пусть c — последняя перед a вершина в $P(b, a)$, для которой c (α, α^*) -достижима из b (может быть и $c = b$). Тогда вершина c должна быть (α, α^*) -достижимой из a цепью $Q(a, c)$, и

$$Q(a, c) \cup P(c, b)$$

будет (α, α^*) -маршрутом от a к b , который может быть сведен к (α, α^*) -цепи между этими двумя вершинами; следовательно, $b \in V(a)$. Таким же образом найдем, что множество $V^*(a)$ состоит из всех вершин, которые эквивалентны a по α^* -достижимости. Можно также сказать, что вершина b эквивалентна a по (α, α^*) -достижимости, если любая вершина, достижимая из a (α, β) -цепью, будет достижима из b (α^*, β) -цепью, и наоборот. Легко проверить, что все такие вершины образуют множество

$$V(G[a]; \alpha^*, \alpha) = G[V(a)],$$

В связи с чередующимися цепями введем *граф чередующейся композиции* $G(\alpha, \alpha^*)$, важный во многих приложениях. Он определяется как произведение

$$G(\alpha, \alpha^*) = G \cdot G^*$$

графа G на его обратный граф G^* . Это означает, что $G(\alpha, \alpha^*)$ и G имеют одно и то же множество вершин V , и две вершины a и c соединены ребром $E_a = (a, c)$ в $G(\alpha, \alpha^*)$ тогда и только тогда, когда a и c соединены чередующейся (α, α^*) -цепью длины 2 в G , т. е. существуют ребра

$$E_1 = (a, b), \quad E_2 = (b, c)$$

соответственно в G и в G^* . Граф $G(\alpha, \alpha^*)$ можно считать неориентированным. Его связными компонентами являются определенные выше достижимые множества $V(a)$.

1.10. Суграфы первой степени в графе

Любой ориентированный граф имеет адекватное представление как двудольный граф. Для множества вершин V данного графа G строится копия V' , которая находится во взаимно однозначном соответствии

$a \rightleftarrows a'$ с V . Графу G соотносим двудольный граф $G(V, V')$, построенный так, что $G(V, V')$ имеет ребро (a, b') тогда и только тогда, когда существует ребро (a, b) в G . Обратно, очевидно, если в двудольном графе $C(V, V')$ существует взаимно однозначное соответствие между множествами вершин, то он может быть представлен как ориентированный граф на V .

Это соответствие между двудольными графами и ориентированными графами позволяет перенести результаты, полученные выше для двудольных графов, на ориентированные графы. Если двудольный граф $G(V, V')$ имеет совершенное паросочетание M , то существует единственная последовательность ребер

$$\dots (a_0, a'_1)(a_1, a'_2) \dots$$

в M , начинающаяся в любой вершине c_0 из V . В соответствующем ему ориентированном графе G этой последовательности отвечает ориентированная цепь

$$\dots (a_0, a_1)(a_1, a_2) \dots$$

Через каждую вершину G проходит ровно одна цепь, так что совершенному паросочетанию в $G(V, V')$ соответствует суграф первой степени в G , или граф подстановок для множества вершин V .

Теоремы о паросочетаниях для локально конечных графов зависят, как мы видели ранее, от понятия дефицита множества, и это понятие также легко переносится на ориентированные графы. *Дефицит* конечного множества A есть величина

$$\delta(A) = v(A) - v(G[A]), \quad (11)$$

где $G[A]$ означает множество всех конечных вершин ребер, имеющих свои начальные вершины в A . Можно также говорить об *обратном дефиците*

$$\delta^*(A) = v(A) - v(G^*[A]), \quad (12)$$

где $G^*[A]$ — множество всех начальных вершин ребер, имеющих свои конечные вершины в A . Легко видеть, что эти дефициты имеют те же свойства, что и дефициты, определенные ранее для двудольных графов.

Как и ранее, обозначим максимальные дефициты через

$$\delta_0 \geq 0, \quad \delta_0^* \geq 0; \quad (13)$$

Теорема 29. *Для того чтобы локально конечный ориентированный граф имел суграф первой степени, необходимо и достаточно, чтобы его максимальные дефициты были равны нулю:*

$$\delta_0 = \delta_0^* = 0. \quad (14)$$

Как и ранее, мы замечаем, что однородный ориентированный граф не имеет дефицита; следовательно, он имеет суграф первой степени.

Если этот суграф удалить из G , то остающийся граф также будет однородным. Это приводит к теореме:

Теорема 30. *Однородный ориентированный граф с множеством вершин V и локальными степенями*

$$\rho(v) = \rho^*(v) = m, \quad v \in V,$$

является прямой по ребрам суммой t графов подстановок

$$G = \bigcup P_i$$

на V .

Так же, как и ранее, мы заключаем, что эта теорема справедлива и в том случае, когда G имеет счетное множество вершин и каждая вершина соединена со счетным числом различных вершин входящими и выходящими ребрами.

Все остальные теоремы о паросочетаниях имеют соответствующие им теоремы для ориентированных графов, и мы предоставляем читателю их сформулировать. Паросочетанию вершин для двудольного графа будет соответствовать часть H ориентированного графа G , которая имеет не более одного выходящего и одного входящего ребра в каждой вершине V . Назовем H частью *не более чем первой степени*. Вершины, в которых H не имеет ни входящего, ни выходящего ребра, являются *дефицитными вершинами*. Если множества дефицитных вершин минимальны, то H есть *максимальная часть не более чем первой степени*. Этот граф состоит из непересекающихся ориентированных простых цепей, покрывающих вершины графа G . Эти цепи могут быть конечными, а также односторонне- или двусторонне-бесконечными. Конечные цепи могут быть циклическими и нециклическими, в том числе нулевой длины.

Максимальным частям не более чем первой степени соответствуют максимальные паросочетания в соответствующем двудольном графе. Если они существуют, то, очевидно, найдутся такие подмножества N и N' , что вершины из N являются дефицитными для выходящих ребер, а вершины из N' — дефицитными для входящих ребер в некотором максимальном H . Как и ранее, мы получаем непересекающиеся разложения

$$V = N \cup G^*[N'] \cup R = N' \cup G[N] \cup R',$$

Здесь множества

$$G[N] \cup R \cup G^*[N'], \quad G[N] \cup R' \cup G^*[N']$$

являются разделяющими множествами для G , т. е. каждое ребро имеет вершину в одном из них; эти особые разделяющие множества имеют сильные минимальные свойства, соответствующие свойствам, указанным ранее.

Максимальные графы не более чем первой степени будут существовать в том частном случае, когда G имеет конечное разделяющее множество. То же справедливо, если граф G локально конечен. В последнем случае, если G имеет конечные максимальные дефициты (13), то число вершин, в которых нет выходящих ребер в H , равно бо, а число вершин в которых пет входящих ребер, равно δ_0 . Так как такие дефицитные вершины являются соответственно начальными и конечными вершинами простых цепей в H , мы получаем, что для конечного G

$$\delta_0 = \delta_0^*.$$

1.11. Перечисление орграфов

В приведенном ниже материале, состоящем из четырех разделов, перечисляются орграфы, турниры, ориентации данного графа и «смешанные графы» (т. е. графы, в которых могут встречаться и дуги, и неориентированные ребра). Кроме всех орграфов, мы перечисляем в этом разделе отношения и связные орграфы. Из решения задачи перечисления турниров выводится в качестве следствия теорема о перечислении сильных турниров. Формула для числа смешанных графов включает как частные случаи формулы, подсчитывающие числа графов, орграфов, направленных графов, полных орграфов и турниров (эти формулы легко могут быть представлены в явной форме).

Несколько слов о методах перечисления, используемых ниже. Подсчет орграфов осуществляется с помощью несложной модификации метода Поля, применявшегося при перечислении графов. Это достигается путем введения новой унарной операции на группах подстановок. Результатом операции является группа, называемая редуцированной упорядоченной парной группой, которая предназначена для того, чтобы действовать на множестве упорядоченных пар различных объектов, тех самых объектов, на множестве которых действует первоначальная группа подстановок.

Цикловой индекс редуцированной упорядоченной парной группы, порожденной группой S_n , используется для подсчета орграфов. Затем, чтобы дать возможность перечислить турниры, этот цикловой индекс модифицируется путем введения ограничения на область действия соответствующей степенной группы. Применяя эту идею к редуцированной упорядоченной парной группе произвольного графа и используя в результирующем цикловом индексе два множества переменных, мы далее получаем возможность подсчитать число ориентации графа. Кульминацией в приложении этих методов является общая теорема перечисления смешанных графов. Она доказывается с

использованием того же циклового индекса, который применяется для перечисления орграфов, снова с двумя множествами переменных и со специальным перечисляющим рядом для фигур.

В этом разделе продолжена аналогия между выводом многочлена $g_p(x)$, перечисляющего графы, и выводом соответствующего циклового индекса парной группы, порожденной группой S_p . В то время как существует только 11 графов четвертого порядка, число всех орграфов четвертого порядка равно 218.

Пусть $d_p(x)$ — перечисляющий многочлен для орграфов порядка p , так что, например,

$$d_3(x) = 1 + x + 4x^2 + 4x^3 + 4x^4 + x^5 + x^6. \quad (15)$$

Значения этих коэффициентов подтверждаются рис. 20.

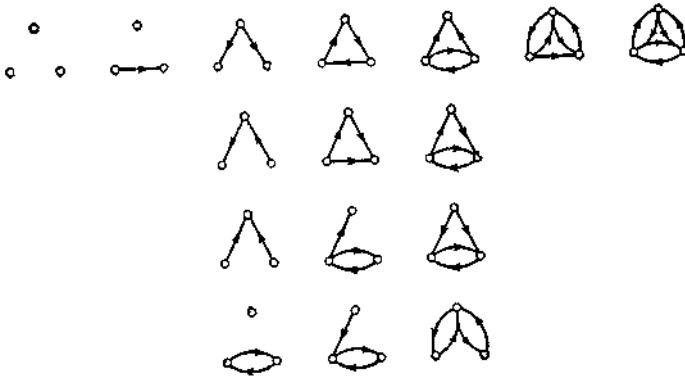


Рис. 20. 16 орграфов третьего порядка,

Мы покажем, следуя Харари, как можно использовать теорему перечисления Пойа для нахождения $d_p(x)$.

Пусть $X = \{1, 2, \dots, p\}$, и пусть $X^{[2]}$ обозначает все упорядоченные пары различных элементов из X . Если $Y = \{0, 1\}$, то функции из $X^{[2]}$ в Y представляют помеченные оргграфы порядка p . Каждая функция f соответствует тому оргграфу $D(f)$ с множеством вершин X , в котором существует дуга, идущая из вершины i в вершину j , тогда и только тогда, когда $f(i, j) = 1$. Таким образом, две функции f и g представляют один и тот же оргграф, если существует подстановка α множества X , такая, что из смежности вершины i к вершине j в оргграфе $D(f)$ следует смежность вершины αi к вершине αj в оргграфе $D(g)$. Значит, оргграфы $D(f)$ и $D(g)$ изоморфны тогда и только тогда, когда для всех пар $(i, j) \in X^{[2]}$ выполняется равенство

$$f(i, j) = g(\alpha i, \alpha j). \quad (16)$$

Это соотношение подсказывает мысль о введении следующей унарной операции на группах подстановок. Если A — группа подстановок с множеством объектов X , то *редуцированная упорядоченная парная группа* группы A , обозначаемая $A^{[2]}$, имеет в качестве множества объектов множество $X^{[2]}$ и индуцируется группой A . То есть для всякой подстановки α из A найдется подстановка α' из $A^{[2]}$, такая, что образ любой пары $(i, j) \in X^{[2]}$ при воздействии на нее подстановкой α' дается формулой

$$\alpha'(i, j) = (\alpha i, \alpha j). \tag{17}$$

Мы теперь можем вывести формулу для числа орграфов.

Теорема. *Перечисляющий многочлен $d_p(x)$ для орграфов порядка p дается следующим выражением:*

$$d_p(x) = Z(S_p^{[2]}, 1+x), \tag{18}$$

где

$$Z(S_p^{[2]}) = \frac{1}{p!} \sum_{(f)} \frac{p!}{\prod k^{j_k} j_k!} \prod_{k=1}^p s_k^{(k-1)j_k+2k} \binom{j_k}{2} \prod_{r < t} s_{[r, t]}^{2(r, t)j_r j_t}. \tag{19}$$

Доказательство. Уравнение (18) является вариантом следствия из теоремы перечисления Пойа. Мы покажем, что редуцированная упорядоченная парная группа, порожденная группой S_p , как раз и является требуемой группой конфигураций. Две функции f и g , действующие из $X^{[2]}$ в $\{0, 1\}$, представляют один и тот же орграф тогда и только тогда, когда существует подстановка α в S_p , такая, что

$$f(z) = g(\alpha'z) \tag{20}$$

для каждой пары z из $X^{[2]}$. Следовательно, орбиты степенной группы $E_2^{S_p^{[2]}}$ соответствуют разным орграфам.

Перечисляющим рядом для фигур является ряд $1+x$, ибо, как и в случае графов, дуга (i, j) либо присутствует в орграфе, либо нет. Таким образом, используя обычные определения весов функций и орбит,

убеждаемся, что вес орбиты группы $E_2^{S_p^{[2]}}$ равен числу дуг в соответствующем орграфе. Это доказывает равенство (18).

Теперь нашей целью является вывод формулы (19) для циклового индекса $Z(S_p^{[2]})$, но сначала мы проиллюстрируем соответствующий прием для случая $p = 4$, чтобы сделать ясным вывод более общего циклового индекса. Каждая из подстановок группы S_4 индуцирует некоторую подстановку из редуцированной упорядоченной парной группы $S_4^{[2]}$. В табл. 1 приведены слагаемые циклового индекса группы $S_4^{[2]}$, индуцированные соответствующими слагаемыми из $Z(S_4)$.

Таблица 1

Слагаемое из $Z(S_4)$	Индукцированное слагаемое из $Z(S_4^{[2]})$
s_1^4 $s_1^2 s_2$ $s_1 s_3$ s_2^2 s_4	s_1^{12} $s_1^2 s_2^5$ s_3^4 s_2^6 s_4^3

Одно из этих слагаемых будет сейчас проиллюстрировано. На рис. 21 упорядоченные пары указываются для краткости следующим образом: $(1,2) = 12$ и т. д. Собирая все слагаемые, получаем

$$Z(S_4^{[2]}) = \frac{1}{24} (s_1^{12} + 6s_1^2 s_2^5 + 8s_3^4 + 3s_2^6 + 6s_4^3), \quad (21)$$

а поэтому

$$Z(S_4^{[2]}, 1+x) = d_4(x) = 1 + x + 5x^2 + 13x^3 + 27x^4 + 38x^5 + 48x^6 + 38x^7 + 27x^8 + 13x^9 + 5x^{10} + x^{11} + x^{12}. \quad (22)$$

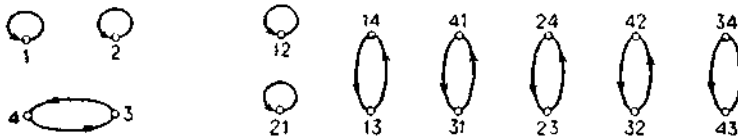


Рис. 21. Подстановка (1) (2) (34) и индуцируемая ею подстановка из $S_4^{[2]}$.

Мы указываем соответствие между слагаемыми цикловых индексов групп S_p и $S_p^{[2]}$, используя следующую запись (для определенности положили $p = 4$):

$$s_1^4 \rightarrow s_1^{12}, \quad s_1^2 s_2 \rightarrow s_1^2 s_2^5$$

и т. д. Чтобы получить выражение для $Z(S_p^{[2]})$, нам нужно найти недостающий член в правой части соотношения

$$s_1^{i_1} s_2^{i_2} \dots s_p^{i_p} \rightarrow ? \quad (23)$$

Точно так же, как при выводе выражения для $Z(S_p^{[2]})$, в случае, когда рассматривается более чем один цикл в одной и той же подстановке α из S_p , существуют два различных вклада в соответствующее слагаемое из $Z(S_p^{[2]})$. Первый вклад вносится такими парами элементов из X , в которых оба элемента принадлежат одному и тому же циклу подстановки α ; второй вклад дают пары элементов, лежащих в двух различных циклах подстановки α (каждый элемент принадлежит «своему» циклу).

Для нахождения первого из этих вкладов рассмотрим цикл z_k длины k , содержащийся в подстановке α . В множестве $X^{[2]}$ существует $k(k-1)$ пар, составленных из элементов, принадлежащих циклу z_k . Эти пары переставляются подстановкой α' в циклах, длина которых равна k . Следовательно, подстановка α' переставляет их в $k-1$ циклах и длина каждого из этих циклов равна k . Таким образом, если существует j_k циклов длины k , то пары элементов, лежащих в «общих» циклах, дают вклад

$$s_k^{j_k} \rightarrow s_k^{(k-1)j_k}. \quad (24)$$

Далее рассмотрим два произвольных цикла подстановки α , скажем z_r и z_t длины r и t соответственно. В множестве $X^{[2]}$ существует $2rt$ пар (i, j) , в которых i переставляется циклом z_r , а j — циклом z_t . Эти пары переставляются подстановкой α' в циклах, длина которых равна $[r, t]$. Следовательно, подстановка α' переставляет их в $2(r, t)$ циклах и длина каждого из этих циклов равна $[r, t]$. Таким образом, вклад, вносимый парами различных циклов, длина одного из которых равна r , а другого $t \neq r$, описывается следующим соотношением:

$$s_r^{j_r} s_t^{j_t} \rightarrow s_{[r, t]}^{2(r, t)j_r j_t}. \quad (25)$$

Наконец, вклад, вносимый парами циклов, имеющими одинаковую длину k , будет таким:

$$s_k^{j_k} \rightarrow s_k^{2j_k \binom{j_k}{2}}. \quad (26)$$

Теперь правая часть соотношения (23), а следовательно и правая часть формулы (19), получается путем перемножения правых частей выражений (24), (25) и (26) по всем подходящим индексам.

Петли в орграфах не допускаются, ибо орграфы определяются как иррефлексивные отношения. Для подсчета числа p -вершинных отношений, которые могут содержать петли, нам необходимо расширить множество объектов группы $S_p^{[2]}$ путем введения *всех* упорядоченных пар из множества $X \times X$. Получаемая при этом группа подстановок называется *упорядоченной парной группой* группы S_p и

обозначается через S_p^2 . Ее цикловой индекс получается видоизменением формулы (19) для $Z(S_p^{[2]})$: нужно умножить каждое слагаемое на $\prod S_k^{j_k}$. Тогда многочлен $r_p(x)$, перечисляющий отношения, дается формулой

$$r_p(x) = Z(S_p^2, 1+x). \quad (27)$$

Из этого уравнения при $p = 2$ находим

$$r_2(x) = 1 + 2x + 4x^2 + 2x^3 + x^4. \quad (28)$$

Правильность коэффициентов многочлена $r_2(x)$ проверяется по рис. 22.

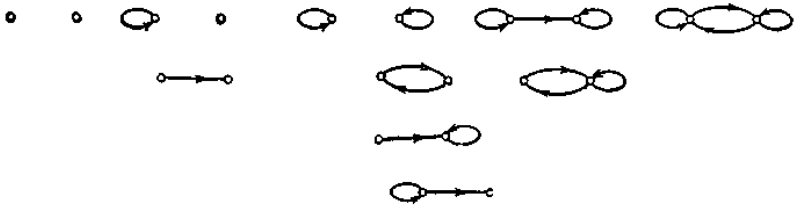


Рис. 22. Отношения на множестве из двух элементов.

Орграф называется *связным*, если, игнорируя все ориентации на его дугах, получаем связный мультиграф. Харари, Норманом и Картрайтом такие орграфы названы «слабо связными». Связные орграфы перечисляются «на языке» всех орграфов с помощью метода, идентичного методу, основанному на формуле

$$1 + g(x) = \exp \left\{ \sum_{k=1}^{\infty} c(x^k)/k \right\}$$

и использованному при перечислении связных графов. Мы завершим наши рассуждения замечанием, что если в формуле

$$p a_p = p g_p - \sum_{k=1}^{p-1} k a_k g_{p-k}$$

заменить g_p числом d_p орграфов порядка p , то уравнение

$$c_p = \sum_{d|p} \frac{\mu(d)}{d} a_{p/d}$$

, включающее параметры a_p , можно использовать для подсчета числа связных орграфов. Результаты такого подсчета при $p \leq 8$ приводятся в табл. 2.

Таблица 2

Связные орграфы

p	d_p	pa_p	c_p
1	1	1	1
2	3	5	2
3	16	40	13
4	218	801	199
5	9 608	46 821	9 364
6	1 540 944	9 185 102	1 530 843
7	882 033 440	6 163 297 995	880 471 142
8	1 793 359 192 848	14 339 791 693 249	1 792 473 955 306

1.12. Турниры

Число помеченных турниров (см. рис. 23) равно числу помеченных графов; для непомеченных турниров это совсем не так.

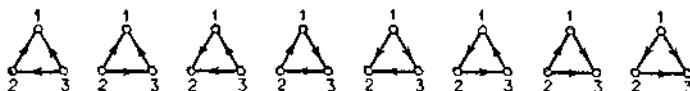


Рис. 23. Восемь помеченных турниров третьего порядка.

Все турниры с 2, 3 и 4 вершинами показаны на рис. 24.

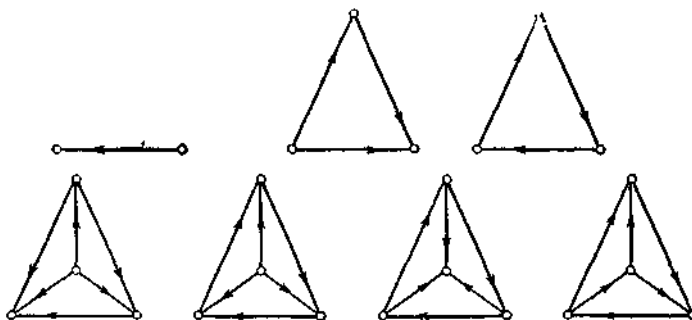


Рис. 24. Наименьшие турниры.

Первый на этом рисунке турнир с тремя вершинами называется *транзитивной тройкой*, а второй — *циклической тройкой*.

Перечисление турниров осуществлено Дэвисом.

Теорема. Число $T(p)$ турниров порядка p дается формулой

$$T(p) = \frac{1}{p!} \sum_{(j)}^n \frac{p!}{\prod k! j_k!} 2^{t(j)}, \quad (29)$$

где звездочка у знака суммы указывает на то, что суммирование ведется только по таким разбиениям (j) , у которых $j_k = 0$ при всех четных k и где

$$t(j) = \frac{1}{2} \left(\sum_{m, n=1}^p (m, n) j_m j_n - \sum_{k=1}^p j_k \right). \quad (30)$$

Доказательство. Сначала убедимся в правильности формулы (29). Как и в доказательстве теоремы, связанной с формулой (18), замечаем, что орбиты степенной группы

$$E_2 S_p^{[2]}$$

соответствуют орграфам порядка p . Ограничивая эту группу на множество F функций f , представляющих турниры, а именно тех функций f , для которых $f(i, j) \neq f(j, i)$, можем применить вариант леммы Бернсайда. Тогда $T(p)$ может быть выражено через число функций из множества F , которые неподвижны относительно подстановок, принадлежащих той же самой степенной группе. Таким образом, для каждой подстановки α из группы S_p мы ищем число функций f из множества F , таких, что

$$f(i, j) = f(\alpha i, \alpha j) \quad (31)$$

для всех пар $(i, j) \in X^{[2]}$. Функции, удовлетворяющие соотношению (31), являются как раз теми функциями, которые неподвижны относительно подстановки, принадлежащей степенной группе и индуцируемой подстановкой α . Допуская незначительную вольность, будем говорить, что эти функции являются «неподвижными относительно α ». Следовательно, если циклы подстановки α определяют разбиение (j) числа p , то нам надо показать, что число функций из F , неподвижных относительно α , равно $2^{i(j)}$.

Пусть снова α' — подстановка группы $S_p^{[2]}$, индуцируемая подстановкой α . Определим *обращение* произвольного данного цикла z' , содержащегося в некотором разложении подстановки α' на непересекающиеся циклы: обращением цикла z^* является цикл, который переставляет все упорядоченные пары (i, j) , такие, что пары (j, i) переставляются циклом z' . Цикл z' подстановки α' называется *самообратным*, если пара (i, j) переставляется циклом z' всякий раз, когда переставляется пара (j, i) .

Чтобы найти вклад, вносимый в $t(j)$ подстановкой α , когда циклы подстановки α рассматриваются индивидуально, положим, что f

представляет турнир, неподвижный относительно α , так что выполняется соотношение (31). Следовательно, функция f на циклах подстановки α' постоянна.

Дадим обоснование «использованию звездочки» у знака суммы в формуле (29). Если $z_k = (123 \dots k)$ — произвольный четный цикл подстановки α , то подстановка α' имеет самообратный цикл z' а именно цикл, действующий и на пару $(1, (k/2) + 1)$, и на пару $((k/2) + 1, 1)$. Но если f постоянна на этом самообратном цикле z' , то f не является турниром. Итак, не существует турниров, неподвижных относительно α , если α содержит некоторый четный цикл. Формула (29) доказана. Для завершения доказательства теоремы осталось получить формулу (30). Если z_k — нечетный цикл, то он индуцирует $k - 1$ циклов на множестве пар, содержащихся в $X^{[2]}$. Эти $k - 1$ циклов образуют $(k - 1)/2$ пар взаимно обратных циклов. Напомним, что произвольный турнир f , неподвижный относительно α , является константой на циклах подстановки α' . Следовательно, в каждой из $(k - 1)/2$ пар взаимно обратных циклов должны быть выбраны только элементы одного цикла, чтобы получился турнир, неподвижный относительно α . Но это значит, что вклад в $t(j)$, вносимый всеми нечетными циклами подстановки α , равен $\sum_j k (k - 1)/2$, где суммирование ведется по всем нечетным k .

Теперь рассмотрим два цикла z_m и z_n подстановки α и такие пары из множества $X^{[2]}$, которые имеют по одной вершине в каждом из этих двух циклов. При вычислении циклового индекса $Z(S_p^{[2]})$ мы видели, что два таких цикла индуцируют $2(m, n)$ циклов, состоящих из объектов (упорядоченных пар) множества $X^{[2]}$. Эти последние циклы образуют (m, n) пар взаимно обратных циклов. В турнир, неподвижный относительно α , должны быть включены только такие объекты из $X^{[2]}$, которые порождают в точности один цикл в каждой из (m, n) пар взаимно обратных циклов. Значит, вклад в $t(j)$, вносимый всеми парами циклов z_m и z_n при $m \neq n$, равен $\sum_{m < n} j_m j_n (m, n)$. Вклад, вносимый парами циклов, имеющих одну и ту же длину, равен

$$\sum k \binom{j_k}{2}.$$

Суммируя эти три вклада, получаем формулу (30) для $t(j)$. Значения для $T(p)$ при $p \leq 12$, приведены в табл. 1.

Таблица 1

Число турниров

p	T_p
1	1
2	1
3	2
4	4
5	12
6	56
7	456
8	6 880
9	191 536
10	9 733 056
11	903 753 248
12	154 108 311 168

Орграф называется *сильно связным*, или *сильным*, если любая пара его вершин взаимно достижима с использованием ориентированных цепей (другими словами, всякую вершину орграфа можно соединить выходящей из нее ориентированной цепью с любой другой вершиной орграфа.) Известно, что турнир является сильным тогда и только тогда, когда он гамильтонов, т. е. содержит остовный ориентированный цикл. Приведем (см. рис. 25.) все сильные турниры с пятью вершинами, используя при их изображении искусный прием, предложенный Муном: всякий раз, когда дуга на диаграмме отсутствует, это означает, что дуга ориентирована в соответствии с направлением силы тяжести — из верхней вершины в нижнюю.

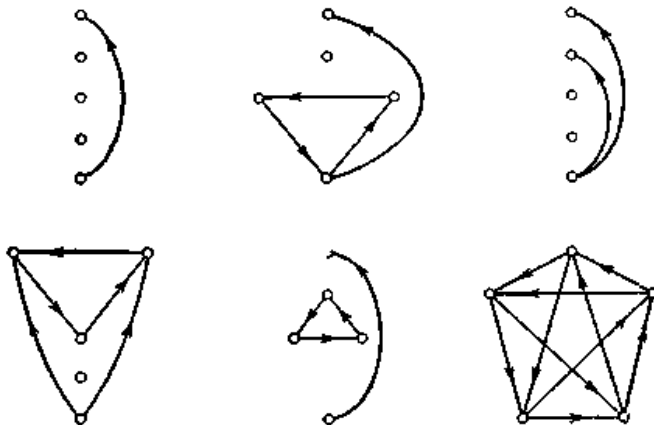


Рис. 25. Сильные турниры пятого порядка.

Число сильных турниров просто выражается через число всех турниров с помощью производящих функций. Для доказательства этого факта требуется ввести несколько определений. *Сильная компонента* орграфа D представляет собой максимальный сильный подграф. *Конденсация* D^* орграфа D имеет в качестве своих вершин сильные компоненты орграфа D , а дуги индуцируются дугами орграфа D (другими словами, дуги, соединяющие сильные компоненты орграфа, «полностью переносятся» в конденсацию.) Как отмечено Харари, Норманом и Картрайтом, конденсация любого турнира является транзитивным турниром, т. е. индуцирует на множестве своих вершин линейный порядок.

Теорема. Пусть $T(x)$ и $S(x)$ — производящие функции для турниров и сильных турниров соответственно. Тогда

$$S(x) = \frac{T(x)}{1+T(x)}. \quad (32)$$

Доказательство. Чтобы получить $S(x)$, обозначим через $T_n(x)$ производящую функцию для турниров, имеющих в точности n сильных компонент. Тогда, в силу приведенных выше рассуждений, $T_n(x) = S^n(x)$ и $T(x) = \sum T_n(x)$, что доказывает соотношение (32).

1.13. Ориентации графов

Ориентация графа G представляет собой приписывание стрелок всем ребрам графа G , т. е. упорядочение всех пар его смежных вершин. *Направленным графом* называется орграф, в котором никакая пара вершин не соединена симметричной парой дуг. Таким образом, всякая ориентация графа порождает направленный граф. Каждый граф G обладает инвариантом $o(G)$, представляющим собой число различных ориентации графа G . Легко проверить, что $o(C_3) = 2$, $o(C_4) = o(C_5) = 4$ и $o(C_6) = 9$.

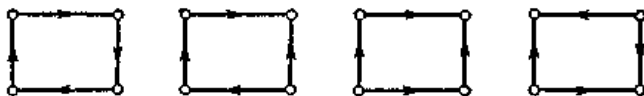


Рис. 26. Ориентации цикла четвертого порядка.

Нашей целью здесь является развитие перечислительного метода для определения числа $o(G)$ ориентации данного графа G . Чтобы достичь этой цели, нам нужно только обобщить некоторые рассуждения из предыдущего параграфа. Там мы нашли число $T(p)$ турниров порядка p , которое является просто $o(K_p)$ — числом ориентации полного графа. Как и в случае графа K_p , формула для $o(G)$ может быть получена путем применения леммы Бернсайда. Получающееся при этом выражение можно упростить, используя специальный цикловой индекс, зависящий от переменных двух типов. Положим, что группа $\Gamma(G)$ действует на множестве $X = \{1, 2, \dots, p\}$. Удобно определить *орграф* $D(G)$, *соответствующий графу* G . Вершины орграфа $D(G)$ совпадают с вершинами графа G , а множество U дуг орграфа $D(G)$ состоит из всех упорядоченных пар (i, j) , таких, что вершины i и j смежны в графе G . Таким образом, ориентация графа G получается путем выбора в точности одной дуги из каждой пары симметричных дуг, принадлежащих орграфу $D(G)$.

Редуцированная упорядоченная парная группа $\Gamma(G)^{[2]}$ действует на множестве *всех* упорядоченных пар (i, j) , у которых i и j принадлежат X , $i \neq j$ и вершины i и j являются смежными в графе G . Нам, таким образом, требуется ограничение группы подстановок $\Gamma(G)^{[2]}$, именно группа $\Gamma(G)^{[2]}|U$, которая действует только на дугах из U . Цикловой индекс этой группы обозначим через $Z(\Gamma(G)^{[2]}|U)$. Чтобы различать между собой разные виды встречающихся здесь циклов, удобно использовать переменные t_k для самообратных циклов и переменные s_k

для пар взаимно обратных циклов. Таким образом, это различие проявляется в самом определении многочлена $Z(\Gamma(G)^{[2]}|U; s_k, t_k)$. Заметим, что, как и в случае турниров, переменные t_k возникают только из четных циклов длины k , содержащихся в автоморфизмах графа G . Приводимая ниже теорема следует из ограниченной формы леммы Бернсайда.

Теорема. Число ориентации графа G дается формулой

$$o(G) = Z(\Gamma(G)^{[2]}|U; \sqrt{2}, 0). \quad (33)$$

Заметим, что радикал $\sqrt{2}$ после подстановки исчезает, ибо взаимно обратные циклы встречаются только попарно. Кроме того, довольно странный перечисляющий ряд для фигур, целиком состоящий из 0, служит вполне определенной цели: он указывает на отсутствие симметричных пар дуг.

Мы можем только в принципе вычислить число $o(G)$ для произвольного графа G , потому что цикловой индекс, указанный в формуле (33), найти непросто. Однако, если $G = C_p$ — цикл порядка p , то можно получить точную формулу для соответствующего циклового индекса. Так как $\Gamma(C_p) = \Gamma_1(C_p) = D_p$ — диэдральная группа, то мы можем видоизменить формулу для $Z(D_p)$, чтобы получить выражение для $Z(\Gamma(C_p)^{[2]}|U; s_k, t_k)$. Учитывая тот факт, что переменные t_k вносятся только обращениями, имеем

$$Z(\Gamma(C_p)^{[2]}|U; s_k, t_k) = \frac{1}{2^p} \sum_{d|p} \varphi(d) s_d^{3p/d} + \begin{cases} \frac{1}{2} s_2^{p-1} t_2, & \text{если } p \text{ — нечетное,} \\ \frac{1}{4} (s_2^p + s_2^{p-2} t_2^2), & \text{если } p \text{ — четное.} \end{cases} \quad (34)$$

Следовательно, мы можем написать

$$o(C_p) = \frac{1}{2^p} \sum_{d|p} \varphi(d) 2^{2p/d} + \begin{cases} 0, & \text{если } p \text{ — нечетное,} \\ 2^{(p-4)/2}, & \text{если } p \text{ — четное.} \end{cases} \quad (35)$$

Наш подход можно также приспособить к перечислению направленных графов, но этот результат будет представлен как частный случай перечисления смешанных графов, которое рассматривается в следующем параграфе.

1.14. Смешанные графы

Смешанный граф может содержать как неориентированные ребра, так и ориентированные. Например, граф, изображенный на рис. 27, является смешанным графом с двумя неориентированными и тремя ориентированными ребрами.

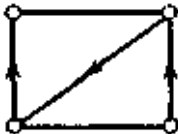


Рис. 27. Смешанный граф четвертого порядка.

Неориентированный граф можно трактовать как смешанный граф, не имеющий ориентированных ребер. Далее орграф можно рассматривать как смешанный граф, заменяя для этого каждую симметричную пару ориентированных ребер неориентированным ребром.

Нашей целью является получение формулы, перечисляющей смешанные графы с p вершинами в соответствии с числом неориентированных и ориентированных ребер в них.

Пусть m_{pqr} — число смешанных графов с p вершинами, q ориентированными ребрами и r неориентированными ребрами. Тогда многочлен $m_p(x, y)$, перечисляющий смешанные графы с p вершинами в соответствии как с числом неориентированных, так и числом ориентированных ребер, определяется формулой

$$m_p(x, y) = \sum_{q, r} m_{pqr} x^q y^r, \quad (36)$$

где $q + r \leq \binom{p}{2}$. Из рис. 28 следует, что при $p = 3$ эта формула имеет вид

$$m_3(x, y) = 1 + x + 3x^2 + 2x^3 + y + 2xy + 3x^2y + y^2 + xy^2 + y^3. \quad (37)$$

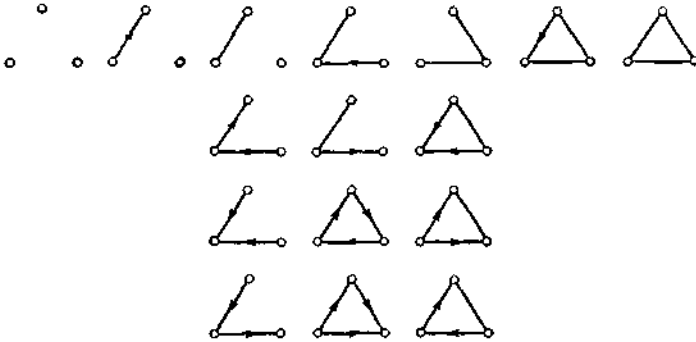


Рис. 28. 16 смешанных графов третьего порядка.

Чтобы вывести формулу для $m_p(x, y)$, мы воспользуемся слабо модифицированной формой теоремы Пойа. В этой модификации оба вида перечисляющих рядов для фигур применяются совместно со специальным цикловым индексом, зависящим от переменных двух типов и рассмотренным в предыдущем параграфе. Используя обозначения, введенные в п.1.13, получаем равенство

$$\Gamma(K_p)^{[2]} | U = S_p^{[2]}. \quad (38)$$

Таким образом, $Z(S_p^{[2]}; s_k, t_k)$ является цикловым индексом редуцированной упорядоченной парной группы $S_p^{[2]}$, в котором переменные s_k используются для пар взаимно обратных циклов, а переменные t_k — для самообратных циклов. Мы увидим, что, подставляя в это выражение вместо каждой переменной s_k функцию $(1 + 2x^k + y^k)^{1/2}$, а вместо каждой переменной t_k функцию $1 + y^{k/2}$, получим многочлен $m_p(x, y)$.

Теорема. Перечисляющий многочлен для смешанных графов порядка p дается соотношением

$$m_p(x, y) = Z(S_p^{[2]}; (1 + 2x + y)^{1/2}, 1 + y^{1/2}), \quad (39)$$

где

$$Z(S_p^{[2]}; s_k, t_k) = \frac{1}{p!} \sum_{(j)} \frac{p!}{\prod_k k^{j_k} j_k!} \prod_{k \text{ нечетное}} s_k^{(k-1)j_k} \times \\ \times \prod_{k \text{ четное}} (s_k^{k-2} t_k)^{j_k} \prod_k s_k^{2k} \binom{j_k}{2} \prod_{r < w} s_{[r, w]}^{2(r, w)} j_r j_w. \quad (40)$$

В качестве примера приведем некоторые детали для случая $p = 3$. Сначала получаем формулу для циклового индекса:

$$Z(S_3^{[2]}, s_k, t_k) = (1/3!) (s_1^6 + 3s_2^2 t_2 + 2s_3^2). \quad (41)$$

Подставляя перечисляющие ряды для фигур: $(1 + 2x+y)^{1/2}$ и $1+y^{1/2}$, приходим к выражению

$$m_3(x, y) = \frac{1}{6} ((1 + 2x + y)^3 + 3(1 + y)(1 + 2x^2 + y^2) + 2(1 + 2x^3 + y^3)), \quad (42)$$

которое полностью согласуется с формулой (37), а соответствующие ему смешанные графы показаны на рис. 28.

Соотношение (40) не требует комментариев, кроме замечания о том, что оно получается из формулы (19) с помощью ее модификации в соответствии с той частью доказательства формулы (29), где показывается, что каждый четный цикл подстановки из S_p индуцирует самообратный цикл.

Здесь представлен краткий набросок доказательства формулы (39). Как

обычно, степенная группа $E_2^{S_p^{[2]}}$, действует на функциях из множества

$Y^{X^{[2]}}$. Так как каждая такая функция f представляет оргграф, скажем, с q ориентированными ребрами и r симметричными парами дуг, то f можно также трактовать как смешанный граф с q ориентированными и r неориентированными ребрами. Очевидно, что любые две функции из

множества $Y^{X^{[2]}}$ принадлежат одной и той же орбите степенной группы тогда и только тогда, когда соответствующие им смешанные графы изоморфны. Наконец, функциям обычным образом

приписываются веса и, применяя взвешенный вариант леммы Бернсайда, получают соотношение (39).

Идея использования выражения $1 + 2x + y$ состоит в следующем: слагаемое 1 соответствует парам несмежных вершин, в то время как $2x$

указывает на две возможные ориентации, а y — на неориентированное ребро. Радикал в выражении $(1 + 2x + y)^{1/2}$ исчезает в $m_p(x, y)$, потому что каждая переменная s_k встречается только в четной степени, ибо

взаимно обратные циклы обязательно появляются парами. Аналогично двучлен $1 + y$ отражает, как обычно, отсутствие ребра и наличие

неориентированного ребра (ориентированные ребра просто запрещены для самообратных циклов). Радикал в $1 + y^{1/2}$ также исчезает в

$m_p(x, y)$, ибо, как показано в формуле (40), в единственном ее произведении, содержащем t_k , индексы k — четные. Это в свою очередь справедливо потому, что каждый самообратный цикл имеет четную

длину.

Перечисляющие многочлены $g_p(x)$ и $d_p(x)$ для графов и оргграфов были уже выведены раньше. Многочлен $o_p(x)$ для направленных графов

66

найден Харари. Заметим, что каждый из этих многочленов легко получается из $m_p(x, y)$, который является, таким образом, одновременным обобщением всех трех указанных перечислительных формул:

$$d_p(x) = m_p(x, x^2), \quad o_p(x) = m_p(x, 0), \quad g_p(y) = m_p(0, y). \quad (43)$$

Для $p = 3$ находим, используя соотношение (42):

$$\begin{aligned} d_3(x) &= m_3(x, x^2) = 1 + x + 4x^2 + 4x^3 + 4x^4 + x^5 + x^6, \\ o_3(x) &= m_3(x, 0) = 1 + x + 3x^2 + 2x^3, \\ g_3(y) &= m_3(0, y) = 1 + y + y^2 + y^3. \end{aligned} \quad (44)$$

Правильность этих выражений легко установить с помощью рис. 28. *Полный орграф* имеет для каждой пары вершин либо ориентированное ребро, либо симметричную пару ребер, соединяющих эти вершины. Орграф на рис. 29 является полным ориентированным графом с пятью вершинами, тремя парами симметричных дуг и семью ориентированными ребрами.

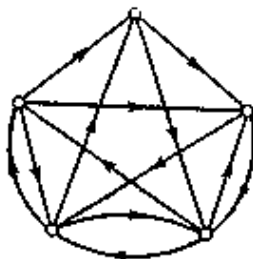


Рис. 29. Полный орграф пятого порядка.

Пусть c_{pqr} — число полных орграфов с p вершинами, имеющих в точности q ориентированных ребер и r симметричных пар. Многочлен $c_p(x, y)$, который перечисляет полные орграфы с p вершинами в соответствии и с числом ориентированных ребер, и с числом симметричных пар, определяется формулой

$$c_p(x, y) = \sum c_{pqr} x^q y^r, \quad (45)$$

где

$$q + r = \binom{p}{2}.$$

Из рис. 30 следует, что при $p = 3$ формула имеет вид

$$c_3(x, y) = 2x^3 + 3x^2y + xy^2 + y^3.$$

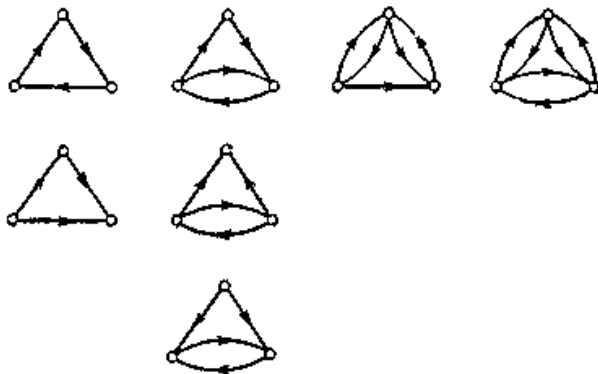


Рис. 30. Полные орграфы третьего порядка.

Перечислительная формула для $c_p(x, y)$ легко получается путем модификации формулы для смешанных графов. Число 1 в каждом из двух перечисляющих рядов для фигур — $(1 + 2x + y)^{1/2}$ и $1 + y^{1/2}$ — отражает возможность отсутствия ребра, соединяющего пару вершин. Но в полном орграфе всегда существует либо ориентированное ребро, либо симметричная пара, соединяющие пару вершин. Поэтому подходящими перечисляющими рядами для фигур являются ряды $(2x + y)^{1/2}$ и $y^{1/2}$.

Следствие. Перечисляющий многочлен для полных орграфов с p вершинами дается формулой

$$c_p(x, y) = Z(S_p^{21}; (2x + y)^{1/2}, y^{1/2}). \quad (46)$$

Из этого следствия немедленно выводим, что число турниров с p вершинами равно

$$T(p) = c_p(1, 0). \quad (47)$$

Используя (47) и (40), с помощью несложных преобразований получаем в явной форме формулы (29) и (30).

Общее число c_p полных орграфов, если не фиксировать число ориентированных ребер и симметричных пар, есть

$$c_p = c_p(1, 1). \quad (48)$$

Например, рис. 30 показывает, что $c_3 = 7$.

Используя формулу (40), получаем следующее выражение для c_p .

Теорема. Число полных орграфов порядка p есть

$$c_p = \frac{1}{p!} \sum_{(j)} \frac{p!}{\prod k! j_k!} 3^{a(j)}, \quad (49)$$

где

$$a(j) = \sum_{k=1}^p \left(\left[\frac{k-1}{2} \right] j_k + k \binom{j_k}{2} \right) + \sum_{r < s} (r, s) j_r j_s. \quad (50)$$

Первые пять значений величины c_p таковы:

p	1	2	3	4	5
c_p	1	2	7	42	382

1.15. Правильная нумерация

Пусть вершины конечного ориентированного графа занумерованы от 1 до n . Нумерация называется *правильной на ребре* $(v_i, v_j) \in G$, если $i < j$, и *правильной на графе* G , если она правильна на всех его ребрах.

Правильная нумерация вершины графа G может существовать лишь в том случае, когда G — ациклический граф. Правильную нумерацию можно рассматривать как расширение отношения частичной упорядоченности, заданной на множестве V вершин конечного ориентированного ациклического графа, до отношения полной упорядоченности. Строится это расширение следующим образом.

Пусть G — конечный ориентированный ациклический граф, $v' < v''$ — соответствующее отношение строгой частичной упорядоченности.

Если вершины v', v'' не удовлетворяют ни отношению $v' < v''$, ни обратному отношению $v'' < v'$, то к множеству истинных формул можно добавить одно из этих соотношений, например $v' < v''$, и замкнуть это расширение отношения $<$: $v_1 < v_2 \equiv v_1 < v_2 \vee (v_1 < v' \& v'' < v_2)$.

Когда выполняются отношения $v_1 < v_2$ и $v_2 < v_3$, возможны 3 случая:

- 1) $v_1 < v_2 \& v_2 < v_3$, тогда $v_1 < v_3$, т. е. $v_1 < v_3$;
- 2) $v_1 < v_2 \& v_2 < v' \& v'' < v_3$, тогда $v_1 < v' \& v'' < v_3$, т. е. $v_1 < v_3$;
- 3) $v_1 < v' \& v'' < v_2 \& v_2 < v_3$, тогда $v_1 < v' \& v'' < v_3$, т. е. $v_1 < v_3$.

Четвертый случай — $v_1 < v' \& v'' < v_2 \& v_2 < v' \& v'' < v_3$ невозможен. По транзитивности старого отношения $<$ в этом случае $v'' < v'$, что не выполняется по условию. Таким образом, новое отношение транзитивно.

Оно антирефлексивно, так как выполнение отношений $v_1 < v_2$ и $v_2 < v_1$ означает, что:

- 1) $v_1 < v_2 \& v_2 < v_1$ — сочетание условий, невозможное из-за антирефлексивности старого отношения $<$;
- 2) $v_1 < v_2, v_2 < v', v'' < v_1$, тогда по транзитивности старого отношения $v'' < v'$, а это по условию не так;
- 3) $v_1 < v', v'' < v_2, v_2 < v_1$ — так же, как и для второго случая, отсюда следует $v'' < v'$;
- 4) $v_1 < v', v'' < v_2, v_2 < v'$ и $v'' < v_1$ — но и в этом случае по транзитивности отношения $< v'' < v'$.

Продолжая, если нужно, этот процесс расширения отношения строгой частичной упорядоченности, мы вследствие конечности числа вершин графа G через конечное число шагов получим отношение строгой полной упорядоченности \mathcal{Q} . Для любых вершин v_1, v_2 , различных между собой, будет выполняться одно и только одно из условий $v_1 < v_2$ или $v_2 > v_1$. Для этого отношения есть единственная правильная нумерация. Номер 1 получает вершина v' , для которой выполняются все условия $v' < v$ ($v \neq v'$). Существование этой вершины легко доказать — это начальная вершина ациклического графа \mathcal{G}^0 , соответствующего нашему порядку. Номер 2 имеет вершина, для которой выполняются условия $v'' < v$ [$v \in G \setminus (v' \cup v'')$], начальная в подграфе $\mathcal{G}^0 \setminus v'$ и т.д. Такая нумерация правильна и для исходного отношения $<$.

1.16. Длины путей, протяженности и расстояния между вершинами графа

Длины путей, протяженности и расстояния между вершинами графа уже были определены ранее. Часто рассматриваются более общие определения этих понятий. Пусть каждому ребру $(v', v'') \in G$ поставлено в соответствие действительное число $l(v', v'')$ — его длина. Тогда длина любого пути $L(v_{i_0}, v_{i_1}, \dots, v_{i_p})$ определяется как сумма длин входящих в него ребер:

$$l(L) = \sum_{k=1}^p l(v_{i_{k-1}}, v_{i_k}).$$

Расстоянием $d(v_i, v_j)$ между двумя вершинами v_i и v_j графа называется нижняя грань длин путей $L(v_i, \dots, v_j)$ с началом в первой и концом во второй из этих вершин; протяженностью $g(v_i, v_j)$ — верхняя грань этих длин.

Считается, что длина пути $L_0(v_i)$, состоящего из одной вершины, равна 0; если вершины v_i и v_j в графе G не связаны, то $d(v_i, v_j) = +\infty$,

$g(v_i, v_j) = -\infty$. Приведенные ранее определения длины пути, расстояния и протяженности являются частными случаями этого определения, когда длины всех ребер равны единице. Так как при изменении знаков длин $l(v', v'')$ всех ребер (v', v'') на противоположные расстояния и протяженности меняются местами с заменой знаков на противоположные, достаточно исследовать свойства расстояний.

Если существует путь $L(v_i, \dots, v_h, \dots, v_j)$, в котором некоторая вершина v_h принадлежит циклу $Z(v_h, \dots, v_h)$ отрицательной длины $l' < 0$, то $d(v_i, v_j) = -\infty$. Действительно, в этом случае путь $L_q(v_i, \dots, v_h, \dots, v_h, \dots, v_j)$, состоящий из отрезка (v_i, \dots, v_h) пути L , q раз повторенного цикла Z и отрезка (v_h, \dots, v_j) пути L , имеет длину $l(L) + ql'$, т. е. может оказаться меньше любого отрицательного числа. Если же в любом связывающем рассматриваемые вершины пути нет вершин, принадлежащих циклам отрицательной длины, то, выбросив из таких путей циклы, мы получим простые пути не большей длины. В конечном графе G количество таких путей конечно и нижняя грань их длин достигается.

Теорема. Расстояние $d(v_i, v_j)$ между различными вершинами графа G удовлетворяет условию

$$d(v_i, v_j) = \inf_{(v_h, v_j) \in G} (d(v_i, v_h) + l(v_h, v_j)).$$

При доказательстве будем считать, что нижние грани достигаются (для конечных графов это всегда верно). Пусть $(v_h, v_j) \in G$ и $L(v_i, \dots, v_h)$ — путь минимальной длины, соединяющий v_i с v_h . Добавив к нему ребро (v_h, v_j) , мы получим путь $L'(v_i, \dots, v_h, v_j)$, соединяющий v_i с v_j . Следовательно, $d(v_i, v_j) \leq l(L') = l(L) + l(v_h, v_j) = d(v_i, v_h) + l(v_h, v_j)$, откуда

$$d(v_i, v_j) \leq \inf_{(v_h, v_j) \in G} (d(v_i, v_h) + l(v_h, v_j)).$$

Если же $L(v_i, \dots, v_g, v_j)$ — путь минимальной длины, соединяющий v_i с v_j (он не состоит из одной вершины, так как $v_i \neq v_j$), то $L'(v_i, \dots, v_g)$ — путь, соединяющий v_i и v_g , и $d(v_i, v_j) = l(L) = l(L') + l(v_g, v_j) \geq d(v_i, v_g) + l(v_g, v_j) \geq \inf_{(v_h, v_j) \in G} (d(v_i, v_h) + l(v_h, v_j))$.

Если вершина v_i принадлежит какому-либо циклу отрицательной длины, то $d(v_i, v_i) = -\infty$. В противном случае $L_0(v_i)$ — кратчайший путь с началом и концом v_i и $d(v_i, v_i) = 0$.

1.17. Графы с отмеченными ребрами и вершинами

В приложениях теории графов часто встречается более общее понятие, чем рассматриваемые выше понятия неориентированного и

ориентированного графа. Это граф, вершины и ребра которого различны по качеству. Можно определить множества $\{v^i\}$ и $\{v^j\}$ всех рассматриваемых типов вершин и ребер и каждой вершине $v \in G$ или ребру $e \in G$ отнести соответствующую метку, определяющую, к какому типу это ребро или вершина относится.

Далее приводится принадлежащий Б. Ю. Вейсфейлеру и А.А.Леману способ изображения графа G с отмеченными ребрами и вершинами. Пусть $1, 2, \dots, k$ — различные способы отметить ребра графа G , $1, 2, \dots, l$ — способы отметить его вершины. Заметим, что петля в любой вершине тоже является способом ее отметки. Поэтому можно считать, что по номеру отметки вершины v можно определить, имеется ли петля в этой вершине и как она отмечена.

Элементы матрицы смежности нашего графа G — это не числа, а свободные переменные. Если ребро (i, j) отмечено m -м способом (граф G считается полным, в противном случае не соединенные ребрами вершины соединяются ребрами нового типа), то соответствующий элемент матрицы смежности равен x_m : $\delta_{ij} = x_m$, $i \neq j$. Если же вершина h ($1 \leq h \leq n$) отмечена g -м способом ($1 \leq g \leq l$), то $\delta_{hh} = v_g$. Обычному графу G без петель, кратных ребер и разбиения вершин и ребер на типы тоже можно поставить в соответствие матрицу этого типа.

Вводятся переменные v, x, y и

$$\delta_{ij} = \begin{cases} x, & \text{если } (i, j) \in G; \\ y, & \text{если } (i, j) \notin G \quad (i \neq j); \\ v, & i = j. \end{cases}$$

1.18. Квадрат матрицы смежности

Свободные переменные, являющиеся элементами матрицы смежности рассматриваемого графа с помеченными ребрами, не связаны никакими соотношениями. В частности, они не коммутативны между собой. Элементы квадрата матрицы смежности — это формальные суммы попарных произведений свободных переменных:

$$\begin{aligned} \delta_{ij}^2 &= \sum_{k=1}^n \delta_{ik} \delta_{kj} = \\ &= \begin{cases} \sum_{k \neq i, k \neq j} x_{m(i, k)} x_{m(k, j)} + v_{g(i)} x_{m(i, j)} + x_{m(i, j)} v_{g(j)}, & i \neq j; \\ \sum_{k \neq i} x_{m(i, k)} x_{m(k, i)} + v_{g(i)} v_{g(i)}, & i = j, \end{cases} \end{aligned}$$

где $m(i, j)$ — номер недиагональной переменной, стоящей в i -й строке и j -м столбце; $g(i)$ — номер диагональной переменной, которой равен элемент δ_{ij} матрицы смежности. Таким образом, диагональные

элементы квадрата матрицы смежности отличаются от недиагональных.

Каждому слагаемому в этой сумме произведений соответствует путь в графе G из двух ребер с началом v_i и концом v_j . Так как граф G полный, это соответствие взаимно-однозначно. При $i \neq j$ выделенным произведениям $v_{g(i)}x_{m(i, j)}$ и $x_{m(i, j)}v_{g(j)}$, соответствуют двухзвенные пути с петлей в начале или конце, т. е. по существу однозвенный путь

$L(x_{m(i, j)})$.

Если элементы $\delta_i^{(2)j}$ и $\delta_i^{(2)j'}$ рассматриваемой матрицы состоят из одинаковых слагаемых, то между двухзвенными и однозвенными путями в графе G , связывающими i' с j' и i'' с j'' , можно установить взаимно-однозначное соответствие так, что первые и вторые ребра соответствующих путей будут помечены одинаковым образом.

1.19. Лексикографическая нумерация многочленов

Классификация элементов квадрата матрицы смежности не должна зависеть от нумерации его вершин. Поэтому многочлены с одинаковыми слагаемыми, но расположенными в разных порядках, нужно считать равными. Порядок же нумерации исходных свободных переменных будем считать существенным. Недиagonальные и диагональные переменные у нас занумерованы по отдельности; чтобы иметь возможность рассматривать их общий порядок, будем считать, что последние старше.

Попарные произведения свободных переменных легко упорядочить лексикографически:

$$\xi_{\alpha'} \xi_{\beta'} > \xi_{\alpha''} \xi_{\beta''} \equiv (\alpha' > \alpha'') \vee (\alpha' = \alpha'' \ \& \ \beta' > \beta'').$$

Если теперь переставить одночлены в рассматриваемых элементах квадрата матрицы смежности в порядке старшинства, то и их можно упорядочить лексикографически:

$$\sum_{k=1}^{n'} \mu'_k > \sum_{k=1}^n \mu''_k \equiv (\mu'_1 > \mu''_1) \vee (\mu'_1 = \mu''_1 \ \& \ \mu'_2 > \mu''_2) \vee \dots \\ \dots \vee (\mu'_1 = \mu''_1 \ \& \ \mu'_2 = \mu''_2 \ \& \ \dots \ \& \ \mu'_{n-1} = \mu''_{n-1} \ \& \ \mu'_n > \mu''_n).$$

Однако таким образом недиагональные и диагональные многочлены будут упорядочиваться по отдельности. Как и в случае свободных переменных, диагональные многочлены (ранее было отмечено, что они отличаются от недиагональных) считаются более старшими.

1.20. Новые свободные переменные

Соответствие вершинам и ребрам графа G одних и тех же или разных сумм произведений переменных является более важным обстоятельством, чем конкретный вид последних. Поэтому можно пользоваться более экономным способом пометки этих вершин и ребер. Рассматриваемые многочлены нумеруются в порядке старшинства (недиагональные и диагональные по отдельности) и заменяются в квадрате матрицы смежности свободными переменными с теми же номерами:

$$\sum_{k=1}^{n-2} v_k^m + \rho_1^m + \rho_2^m \rightarrow x_m^2, \quad m=1, 2, \dots, l^{(2)};$$

$$\sum_{k=1}^{n-1} \mu_k^g + \pi^g \rightarrow v_g^{(2)}, \quad g=1, 2, \dots, h^{(2)}.$$

Здесь v_k^m и μ_k^g — произведения двух недиагональных переменных; ρ_1^m и ρ_2^m — соответственно произведения диагональной переменной на недиагональную и наоборот, π^g — квадрат диагональной переменной; $l^{(2)}$ — количество различных недиагональных многочленов в квадрате матрицы смежности; $h^{(2)}$ — количество разных диагональных многочленов.

1.21. Старые и новые классы ребер и вершин

Систему различных пометок ребер и вершин полного графа G , описанную введенной ранее матрицей смежности $\Delta = \Delta_1$ можно рассматривать как разбиение множества ребер и вершин на классы. После замены многочленов в квадрате матрицы Δ на новые свободные переменные возникают новое разбиение этих множеств на классы и соответствующая ему матрица смежности Δ_2 :

$$\delta_{2, i, j} = \begin{cases} x_m^{(2)}(i, j), & i \neq j; \\ v_g^{(2)}(i), & i = j. \end{cases}$$

Каждый новый класс ребер или вершин целиком погружен в старый. Более того, если элементы

$$\delta_{2, i', j'} = \delta_{2, i'', j'} \quad (i' \neq i''),$$

то в старой классификации вершины i' и i'' относятся к одному и тому же классу; вершины j' и j'' также принадлежат одному классу.

Действительно, в квадрате исходной матрицы смежности на соответствующих местах стоят одни и те же многочлены:

$$\begin{aligned} \sum_{k=1}^{k-2} \nu_k^m + \rho_1^m + \rho_g^m &= \sum_{k \neq i', k \neq j'} x_m(i', k) x_m(k, j') + \\ + v_g(i') x_m(i', j') + x_m(i', j') v_g(j') &= \sum_{k \neq i'', k \neq j''} x_m(i'', k) x_m(k, j'') + \\ + v_g(i'') x_m(i'', j'') + x_m(i'', j'') v_g(j'') & \\ \sum_{k=1}^{n-1} \mu_k^g + \pi^g &= \sum_{k \neq i'} x_m(i', k) x_m(k, i') + v_g(i') v_g(i) = \\ = \sum_{k \neq i''} x_m(i'', k) x_m(k, i'') + v_g(i'') v_g(i''). & \end{aligned}$$

Следовательно,

$$\begin{aligned} \delta_{i'j'} &= \delta_{i''j''} = x_m(i', j') = x_m(i'', j'') = \delta_{i''j''} = \delta_{i'j'}, \\ \delta_{1i'i'} &= \delta_{1i''i''} = v_g(i') = v_g(i'') = \delta_{1i''i''} = \delta_{1i'i'}, \\ \delta_{1j'j'} &= \delta_{1j''j''} = v_g(j') = v_g(j'') = \delta_{1j''j''} = \delta_{1j'j'}, \end{aligned}$$

что и требовалось доказать.

1.22. Группа автоморфизмов графа

Автоморфизмом ориентированного графа G (без кратных дуг и петель) называется такое взаимно-однозначное отображение множества V его вершин на себя, при котором соединенные ребром вершины переходят в вершины, также соединенные ребром, причем начало — в начало, конец — в конец, а несоединенные — в несоединенные. Если G — конечный граф, то второе следует из первого. Это понятие автоморфизма естественным образом распространяется на графы с отмеченными вершинами и ребрами. Требуется еще, чтобы соответствующие друг другу вершины и соединяющие пары соответствующих вершин ребра принадлежали одним и тем же классам. Так как любой ориентированный граф можно сопоставить полному графу G с отмеченными вершинами и ребрами, в дальнейшем рассматриваются только последние.

Легко видеть, что отображение, обратное автоморфизму, а также произведение двух автоморфизмов являются автоморфизмами. Таким образом, все автоморфизмы графа G образуют группу $\text{Aut}(G)$. Для графа G с конечным числом вершин можно занумеровать последние. Тогда любое взаимно-однозначное отображение $\sigma : V \rightarrow V$ можно

рассматривать, как перестановку $(\mu_1, \mu_2, \dots, \mu_n)$ отрезка натуральных чисел $(1, 2, \dots, n)$ (n — количество вершин графа G). Произведению таких отображений соответствует произведение перестановок

$$(\mu'_1, \mu'_2, \dots, \mu'_n) \times (\mu''_1, \mu''_2, \dots, \mu''_n) = (\mu'_{\mu''_1}, \mu'_{\mu''_2}, \dots, \mu'_{\mu''_n}),$$

и $\text{Aut}(G)$ можно считать подгруппой группы S_n всех перестановок отрезка натуральных чисел $(1, 2, \dots, n)$.

1.23. О проблеме изоморфизма графов

Ранее был рассмотрен алгоритм проверки изоморфизма графов G' и G'' . Фиксируется нумерация вершин одного из графов, например G' . Тогда его матрица смежности Δ' имеет вполне определенный вид. Для другого графа G'' рассматриваются различные нумерации вершин и соответствующие матрицы смежности Δ'' . Если среди них встретится совпадающая Δ' , то рассматриваемые графы G' и G'' — изоморфные. В противном случае — нет.

Пусть при некоторой нумерации вершин матрица смежности графа G'' имеет вид $\bar{\Delta}''$. Тогда остальные ее виды можно получить при помощи одинаковых перестановок строк и столбцов. Каждая такая перестановка взаимно-однозначно соответствует нумерации вершин графа G'' . Если бы для идентификации графов G' и G'' пришлось произвести все $n!$ перестановок строк и столбцов матрицы смежности $\bar{\Delta}''$, то при сколько-нибудь большом n предложенный алгоритм оказался весьма трудоемким.

Однако во многих случаях, когда графы G' и G'' неизоморфны, последнее легко выяснить. Не говоря уже о том, что заведомо неизоморфны графы с разным количеством вершин, можно предложить и другие критерии различия графов. Пусть, например, в графе G' a_1 вершин степени 1, a_2 — степени 2 и т. д., а для графа G'' — соответствующие количества равны b_1, b_2, \dots . Если хотя бы одно из равенств $a_i = b_i$ ($i = 1, 2, \dots$) не выполняется, то графы G' и G'' различны. Более тонкие критерии различия связаны с уточнением описания положения вершин и ребер в графе. Рассмотренные ранее конструкции представляют наиболее эффективный из известных, но, к сожалению, тоже недостаточный язык такого описания.

1.24. Классы эквивалентных вершин

Траекторией вершины $i \in G$ называется множество вершин с номерами μ_i для всех перестановок $(\mu_1, \mu_2, \dots, \mu_n) \in \text{Aut}(G)$. Пусть для какого-нибудь автоморфизма σ $i \xrightarrow{\sigma} j$. Тогда $j \xrightarrow{\sigma^{-1}} i$ и для любого автоморфизма $\rho \in \text{Aut}(G)$ $j \xrightarrow{\rho\sigma^{-1}} \rho(i)$, $i \xrightarrow{\rho\sigma} \rho(j)$

(произведение автоморфизмов — это выполнение сначала отображения, соответствующего правому множителю, а затем — левому). Следовательно, у всех вершин $\sigma(i)$ ($\sigma \in \text{Aut}(G)$) та же траектория, что и i , т. е. траектории — это классы эквивалентности вершин $i \in G$.

Если разбиение множества V вершин графа G'' на траектории определено, при проверке изоморфизма графов G' и G'' можно ограничиться перестановками, не выводящими эти вершины из своих траекторий (на самом деле в этом случае легко обосновываются гораздо более сильные сокращения множества рассматриваемых перестановок). Однако найти автоморфизмы графа и траектории вершин нелегко. Вместе с тем, если известно какое-либо разбиение множества V на классы, целиком включающие траектории, множество рассматриваемых перестановок можно сократить, так как достаточно рассматривать только не выводящие из этих классов перестановки.

В частности, когда вершины графа G отмечены различными способами, достаточно переставлять между собой одинаково отмеченные вершины. Поэтому в общем задача идентификации графов тем проще, чем больше имеется классов вершин и ребер. В частности, полезно увеличить количество классов, целиком включающих траектории вершин.

1.25. Разбиения классов вершин и ребер

Условию включения траекторий удовлетворяют классы вершин, определенные матрицей смежности Δ_2 . Действительно, пусть $\sigma \in \text{Aut}(G)$ и $i \xrightarrow{\sigma} j$. При любом автоморфизме графа G ребра переходят в так же отмеченные ребра, пути — в пути, ребра которых в порядке следования отмечены одинаково, причем соответствующие при этом отображении пути одновременно являются или не являются циклами.

Двухзвенные циклы с началом и концом в вершине i при автоморфизме σ переходят в таким же образом отмеченные циклы с началом и концом в j . Этим определяется разбиение классов

двухзвенных циклов или, что то же самое, попарных произведений независимых переменных в многочленах $\delta_{ii}^{(2)}$ и $\delta_{ij}^{(2)}$ квадратов матрицы смежности $\Delta = \Delta_1$. Следственно, $v_{g(i)}^{(2)} = v_{g(i)}$, что и требовалось доказать.

Матрицу смежности Δ_2 тоже можно возвести в квадрат и построить матрицу Δ_3 со свободными переменными $x_m^{(3)}$ и $v_g^{(3)}$. Классы вершин, одинаково помеченных символами $v_g^{(3)}$ ($g = 1, 2, \dots, h^{(3)}$), целиком помещены в классы помеченных одними и теми же символами матрицы Δ_2 . Дело в том, что Δ_3 получается из Δ_2 так же, как Δ_2 из Δ_1 .

В дальнейшем $\alpha_{ij}^{(s)}$ обозначает множество всех s -звенных путей в графе G с началом i и концом j . Каждый четырехзвенный путь $L \in \alpha_{ij}^{(4)}$ взаимно-однозначным образом представим в виде композиции двухзвенных путей, причем у первой компоненты — начало i , у второй — конец j . Следовательно,

$$\alpha_{ij}^{(4)} = \bigcup_{k=1}^n \alpha_{ik}^{(2)} \times \alpha_{kj}^{(2)},$$

где \times — символ прямого произведения множеств.

Свободные переменные δ_{2ef} матрицы смежности Δ_2 взаимно-однозначно соответствуют наборам способов, которыми помечены соединяющие пару вершин графа G двухзвенные пути. Следовательно, если диагональные элементы $v_{g(i)}^{(3)}$ и $v_{g(i)}$ матрицы Δ_3 и тем самым стоящие в i -й и j -й строках диагональные элементы квадрата матрицы Δ_2 различны, то, по крайней мере, не совпадают разбиения наборов способов, которыми помечены четырехзвенные циклы с началом и концом в вершинах i и j , на поднаборы таких способов для циклов из подмножеств

$$\alpha_{ik'}^{(2)} \times \alpha_{k'j}^{(2)} \quad \text{и} \quad \alpha_{ik''}^{(2)} \times \alpha_{k''j}^{(2)} \quad (k', k'' = 1, 2, \dots, n).$$

Пусть $\sigma \in \text{Aut}(G)$ и $i \xrightarrow{\sigma} j$. Тогда четырехзвенные циклы, проходящие через вершину i , переходят в так же отмеченные циклы, проходящие через j , а каждое из множеств $\alpha_{ik'}^{(2)} \times \alpha_{k'i}^{(2)}$ — в какое-либо из множеств $\alpha_{jk''}^{(2)} \times \alpha_{k''j}^{(2)}$. Значит, рассмотренные ранее разбиения на подмножества способов, которыми помечены ребра этих циклов, одина-

ковые и переменные $v_{g(i)}^{(3)}$ и $v_{g(j)}$ должны совпадать. Таким образом, траектории вершин целиком лежат в классах эквивалентности, определяемых матрицей Δ_3 .

Аналогично можно определить матрицы Δ_4, Δ_5 , доказать, что траектории вершин графа G целиком включены в определенные ими классы эквивалентности, а каждая переменная матрицы Δ_k соответствует набору способов, которыми помечены ребра 2^{k-1} -звенных путей в графе G

с одинаковыми началом и концом. Так как разбиение ребер и вершин графа G , определяемое матрицей Δ_{h+1} , является в то же время разбиением соответствующих классов для матрицы Δ_h , а количества ребер и вершин графа G конечны, начиная с некоторого номера H , все Δ_h ($h \geq H$) определяют одни и те же классы ребер и вершин. К сожалению, и эти стабилизированные классы вершин могут включать по несколько траекторий.

Графы — язык дискретной математики. При формулировке задач дискретной математики или описании методов их решений часто употребляется язык теории графов. Ряд примеров этого будет приведен в следующих главах этой книги. Здесь же укажем лишь один достаточно общий способ постановки задач дискретной математики, при котором естественным образом возникает граф.

Пусть рассматривается множество V объектов v , каждый из которых может находиться в состояниях x_v из заданного для этого элемента v множества состояний X_v . Состояния различных объектов взаимозависимы так, что для каждого объекта v заданы множество Q_v непосредственно влияющих объектов и функция влияния $F_v(v_1, \dots, v_s)$, определяющая состояние x_v объекта v по состояниям x_{v_i} объектов $v_i \in Q_v$: $x_v = F_v(v_1, \dots, v_s)$, $\{v_1, \dots, v_s\} = Q_v$ (в частности, для некоторых v множества Q_v пусты, тогда $x_v = F_v$, где F_v — заданное состояние).

В данном случае объекты $v \in V$ можно рассматривать как вершины структурного графа определенной ранее системы соотношений между состояниями этих объектов. Это ориентированный граф с ребрами (v', v'') , $v' \in Q_{v''}$. Иногда рассматриваются неоднозначные функции $F_v(v_1, \dots, v_s)$ и ищутся системы допустимых состояний объектов, удовлетворяющие каким-либо дополнительным условиям, чаще всего некоторым условиям оптимальности.

2. Раскраски графов

2.1. Введение

Разнообразные задачи, возникающие при планировании производства, составлении графиков осмотра, хранения и транспортировке товаров и т. д., могут быть представлены часто как задачи теории графов, тесно связанные с так называемой «задачей раскраски».

Граф G называют *r-хроматическим*, если его вершины могут быть раскрашены с использованием r цветов (красок) так, что не найдется двух смежных вершин одного цвета. Наименьшее число r , такое, что граф G является r -хроматическим, называется *хроматическим числом*

графа G и обозначается $\gamma(G)$. Задача нахождения хроматического числа графа называется *задачей о раскраске* (или *задачей раскраски*) графа. Соответствующая этому числу раскраска вершин разбивает множество вершин графа на r подмножеств, каждое из которых содержит вершины одного цвета. Эти множества являются независимыми, поскольку в пределах одного множества нет двух смежных вершин. Вообще говоря, хроматическое число графа (так же как числа независимости и доминирования, рассмотренные в предшествующей главе) нельзя найти, зная только числа вершин и ребер графа. Недостаточно также знать степень каждой вершины, чтобы вычислить хроматическое число графа. В этом можно убедиться, рассматривая графы, приведенные на рис. 1(а) и 1(б).

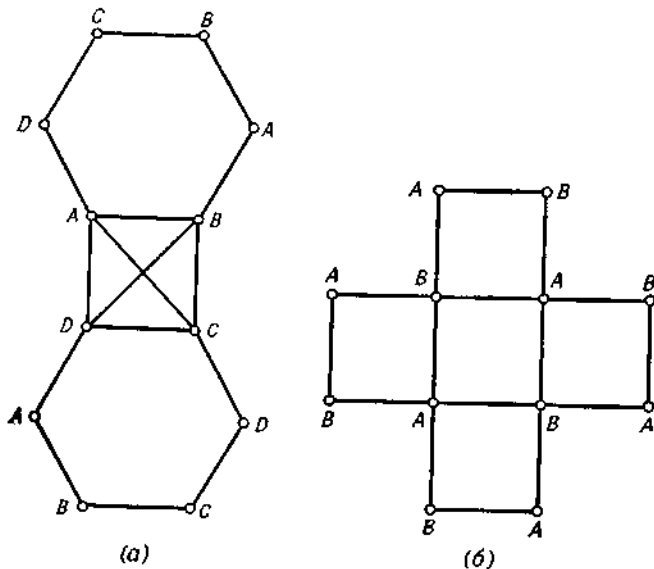


Рис. 1. Два графа с одинаковыми n , m и распределениями степеней вершин, но с различными хроматическими числами,

(а) $\gamma(G) = 4$. (б) $\gamma(G) = 2$.

Эти графы имеют по $n=12$ вершин, $m = 16$ ребер и одинаковые распределения степеней вершин d_i . Однако хроматические числа данных графов равны 4 и 2 соответственно. При известных величинах n (число вершин), m (число ребер) и d_1, \dots, d_n (степени вершин графа) можно получить верхнюю и нижнюю оценки для хроматического числа графа.

Задача нахождения хроматического числа произвольного графа явилась предметом многих исследований в конце XIX и все XX столетие. Сейчас по этому вопросу известно большое количество интересных результатов. В этом разделе мы попытаемся дать их краткий обзор. Мы вводим такие понятия, которые нужны для построения основ теории раскраски графов и алгоритмов решения задачи о раскраске графа. Здесь мы рассматриваем в основном алгоритмы (как точные, так и «приближенные»), позволяющие находить (точное или приближенное) значение хроматического числа произвольного графа и соответствующую этому значению раскраску вершин.

2.2. Раскраска вершин и ребер графа

Раскраской вершин графа $G = \langle V, U \rangle$ в k цветов называется разбиение носителя V графа G , при котором каждое подмножество

$$V_i \left(\bigcup_{i=1}^k V_i = V, V_{i_a} \cap V_{i_b} = \emptyset, i_a, i_b = 1, 2, \dots, k \right)$$

пары смежных вершин. Каждому подмножеству сопоставляется цвет, в который окрашивают все элементы этого подмножества. Вершины, окрашенные в один цвет, будем называть *соцветными*.

Хроматическим числом $\gamma(G)$ графа G называется минимальное число n , для которого граф имеет n -раскраску. Граф, хроматическое число которого равно n , называется n -хроматическим; если же $\gamma(G) \leq n$, то граф G называется n -раскрашиваемым.

Очевидно, что 1-хроматическим граф может быть тогда и только тогда, когда он является пустым графом.

Для 2-хроматических графов справедлива следующая теорема.

Теорема 1 (теорема Кенига). *Граф является 2-хроматическим тогда и только тогда, когда он не содержит циклов нечетной длины.*

Отметим, что аналогичная теорема задает и условие двудольности. Поэтому все 2-хроматические (двухцветные) графы являются двудольными. Любое дерево, поскольку оно является двудольным графом, двухцветно. Оценим хроматическое число графа через его параметры.

Теорема 2. *Если максимальная степень вершин графа G равна $s(G)$, то хроматическое число этого графа не превышает величины $s(G)+1$:*

$$\gamma(G) \leq s(G) + 1 \tag{1}$$

Для большинства графов эту оценку можно улучшить.

Теорема 3 (теорема Брукса). *Граф G со степенью $s(G)$ является s -раскрашиваемым, за исключением двух случаев:*

- 1) при $s(G) > 2$ граф G содержит компоненту, являющуюся полным графом плотности, равной $s(G) + 1$;
- 2) при $s(G) = 2$ граф G содержит компоненту, являющуюся циклом нечетной длины.

Оценки, полученные с помощью этих теорем, дают, однако, хорошее приближение лишь тогда, когда степени всех вершин графа имеют близкие значения. В противном случае оценка может быть значительно завышена. Например, звездный граф $K_{1,n}$ согласно теореме Брукса, является n -раскрашиваемым, в действительности же он двудольный и поэтому для его раскраски достаточно двух красок.

Теорема 4 (теорема Бержа, Оре, Харрари). Для любого графа $G = \langle V, U \rangle$

$$\frac{|V|}{\beta_0} \leq \gamma(G) \leq |V| - \beta_0 + 1, \quad (2)$$

где β_0 - вершинное число независимости графа.

Теорема 5. (теорема Гаддума, Нордхауза). Сумма и произведение хроматических чисел графа $G = \langle V, U \rangle$ и его дополнения $\bar{G} = \langle V, \bar{U} \rangle$ удовлетворяют неравенствам

$$\begin{aligned} &]2\sqrt{|V|}[\leq \gamma(G) + \gamma(\bar{G}) \leq |V| + 1, \\ &|V| \leq \gamma(G) \cdot \gamma(\bar{G}) \leq \left] \frac{(|V| + 1)^2}{4} \right[, \end{aligned} \quad (3)$$

где] [— целая часть.

Теорема 6. Хроматическое число $\gamma(G)$ графа G :

$$\gamma(G) \leq \gamma(G_a) \cdot \gamma(G_b), \text{ где } G = G_a \cup G_b; \quad (4)$$

$$\gamma(G) = \gamma(G_a) + \gamma(G_b), \text{ где } G = G_a + G_b; G_a \cap G_b = \emptyset; \quad (5)$$

$$\gamma(G) \leq \min(\gamma(G_a), \gamma(G_b)), \text{ где } G = G_a \times G_b. \quad (6)$$

Аналогично раскраске вершин графа определим раскраску его ребер. Раскраской ребер графа $G = \langle V, U \rangle$ называется разбиение сигнатуры U графа: $U = \bigcup_i U_i, U_{i_\alpha} \cap U_{i_\beta} = \emptyset, \alpha \neq \beta$, при котором каждое под-

множество U_i не содержит ни одной пары смежных ребер. Каждому подмножеству сопоставляется краска, при этом ребра, принадлежащие одному и тому же подмножеству, называются *соцветными*.

Граф G называется *реберно k -раскрашиваемым*, если его ребра можно раскрасить k красками так, чтобы никакие смежные ребра не были окрашены одним цветом.

Хроматическим классом (хроматическим индексом) графа называется число k такое, что граф является реберно k -раскрашиваемым, но его

ребра не могут быть раскрашены в $k - 1$ цветов. Хроматический класс графа G обозначается $H(G)$.

Например, хроматический класс для двудольных графов $H(K_{1,4}) = 4$, $H(K_{2,3}) = 3$.

Хроматический класс $H(G)$ графа G совпадает с хроматическим числом графа $G = \langle X', \Gamma' \rangle$, определяемого следующим образом: вершины X' его взаимно однозначно соответствуют ребрам графа G и $x'_a \in \Gamma x'_b$, когда соответствующие ребра графа G смежны. Следовательно, задача определения хроматического класса сводится к задаче определения хроматического числа.

В связи со сведением раскраски ребер графа к раскраске его вершин рассмотрим важное свойство графов — свойство реберности.

Граф $G = \langle V, U \rangle$ обладает свойством *реберности* тогда и только тогда, когда существует взаимно однозначное соответствие, при котором каждая вершина $v_i \in V$ графа G сопоставлена ребру $u \in U$ графа $G = \langle V, U \rangle$, при этом матрица смежности вершин графа G подобна матрице смежности ребер графа G . Две матрицы *подобны*, если они совпадают с точностью до перестановки строк (столбцов). В общем случае граф G может быть мультиграфом, т. е. графом с параллельными ребрами.

Теорема 7. *Граф G обладает свойством реберности тогда и только тогда, когда он разложим по аддитивной операции «объединение»*

$\bigcup_i F_i$ на полные подграфы $\{F_i\}$, в которые каждая вершина входит не более двух раз.

Пример 1 (рис. 2). Рассмотрим свойство реберности графа

$$\begin{aligned} G &= \langle V, U \rangle, V = \{a, b, c, d, e\}, \\ U &= \{\{a, b\}, \{a, c\}, \{a, d\}, \{a, e\}, \\ &\{b, c\}, \{b, d\}, \{c, d\}, \{c, e\}, \{d, e\}\}. \end{aligned}$$

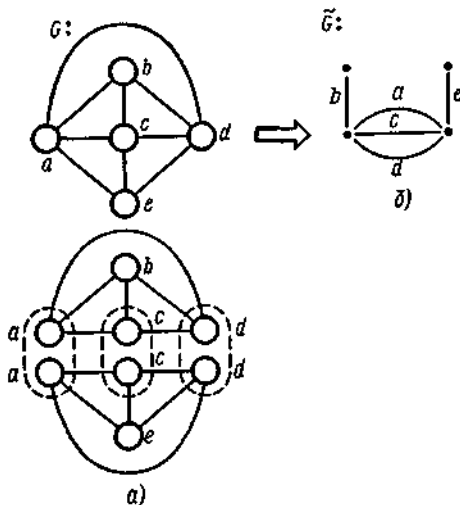


Рис . 2

Граф G разложим на два полных подграфа $G = F_a \cup F_b$ (рис. 2, а):

$$\begin{aligned}
 F_a &= \langle V_a, U_a \rangle, V_a = \{a, b, c, d\}, \\
 U_a &= \{\{a, b\}, \{a, c\}, \{a, d\}, \\
 &\{b, c\}, \{b, d\}, \{c, d\}\}, \\
 F_b &= \langle V_b, U_b \rangle, V_b = \{a, c, d, e\}, \\
 U_b &= \{\{a, c\}, \{a, d\}, \{a, e\}, \\
 &\{c, d\}, \{c, e\}, \{d, e\}\}, \\
 V_a \cap V_b &= \{a, c, d\}.
 \end{aligned}$$

Условия теоремы 7 выполнены, граф G обладает свойством реберности, соответствующий ему реберно производный граф G изображен на рис. 2, б.

Если наложить ограничения на декомпозицию исходного графа G , а именно: при разложении графа G на полные подграфы не допускать повторения ребер в полных подграфах, то реберно производный граф G^0 не будет содержать параллельных ребер.

Теорема 8. Граф G обладает свойством реберности и соответствующий ему реберно производный граф G является обыкновенным графом тогда и только тогда, когда существует

разбиение сигнатуры графа G на полные подграфы, в которые каждая вершина входит не более двух раз.

Пример 2. Определим, имеет ли граф $\mathcal{G}^0 = \langle V^0; U^0 \rangle$ (рис. 3) реберно производный граф.

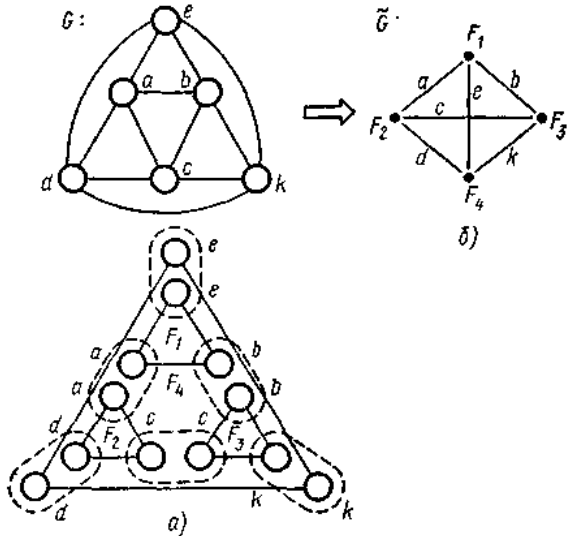


Рис.3

Граф G можно разложить на четыре полных подграфа:

$$G = \bigcup_{i=1}^4 F_i$$

$$F_1 = \langle V_1, U_1 \rangle, V_1 = \{a, b, e\},$$

$$U_1 = \{\{a, b\}, \{a, e\}, \{b, e\}\};$$

$$F_2 = \langle V_2, U_2 \rangle, V_2 = \{a, c, d\},$$

$$U_2 = \{\{a, c\}, \{a, d\}, \{c, d\}\};$$

$$F_3 = \langle V_3, U_3 \rangle, V_3 = \{b, c, k\},$$

$$U_3 = \{\{b, c\}, \{b, k\}, \{c, k\}\};$$

$$F_4 = \langle V_4, U_4 \rangle, V_4 = \{d, e, k\},$$

$$U_4 = \{\{d, e\}, \{d, k\}, \{e, k\}\}.$$

Каждая вершина $v_i, v_i \in V$, входит в полные подграфы $F_i, i = 1, \dots, 4$ два раза (рис. 3, а). Условия теоремы 8 выполнены, реберно производный граф \mathcal{G}^c изображен на рис. 3, б.

При построении реберно производного графа $\mathcal{G}^c = \langle V^c; U^c \rangle$ каждой вершине $v_i \in V$ со степенью $s(v_i) > 1$ сопоставляют выделенный в графе G полный граф $F_i \subset G$. Ребра, инцидентные вершине $v_i \in V^c$, взаимно однозначно соответствуют вершинам графа G , образующим этот полный подграф F_i . Вершина $v_i \in V^c$ со степенью $s(v_i)=1$ коинцидентна ребру ρ_i , которому соответствует вершина $v_j \in V$, входящая в выделенные полные подграфы один раз.

Используя свойства реберной раскраски гиперкуба, предложим оптимальный алгоритм вложения графа G в гиперкуб. Очевидно, что хроматический класс $H(H_n)$ n -мерного куба H_n равен его размерности $n, H(H_n)=n$. При этом соцветные ребра гиперкуба образуют его разрез. Например, при $n = 3$ каждое множество соцветных ребер $a = \{\{0, 4\}, \{1, 5\}, \{2, 6\}, \{3, 7\}\}; b = \{\{0, 2\}, \{1, 3\}, \{4, 6\}, \{5, 7\}\}; c = \{\{0, 1\}, \{2, 3\}, \{4, 5\}, \{6, 7\}\}$ образует разрез (рис. 4).

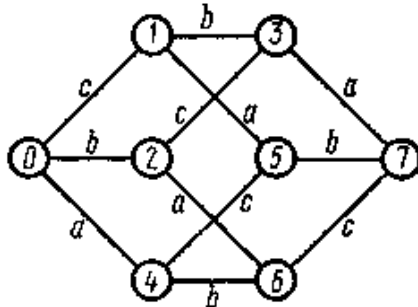


Рис. 4

Теорема 9 (теорема М. И. Смирнова). *Граф G вложим в гиперкуб тогда и только тогда, когда каждое множество соцветных ребер образует его разрез.*

На основании этой теоремы алгоритм вложения графа G в гиперкуб состоит из следующих преобразований:

1. Строя соответствующее дерево, выделяем реберно независимые подмножества (пустые реберные подграфы). Ребрам каждого из этих подграфов можно сопоставить одну и ту же краску.
2. Из полученного в п. 1 списка выделяем подмножества, каждое из которых образует разрез графа G .

3. Используя выделенные в п. 2 подмножества, раскрываем ребра заданного графа.
4. Проверяем, образует ли каждое множество соцветных ребер разрез графа. Если «нет», минимальным сужением сигнатуры или расширением носителя эквивалентуем заданный граф G графом G , удовлетворяющим условиям теоремы 9. Если «да», переходим к п. 5.
5. Производим фактическое вложение графа, полученного в п. 4, в булево пространство. При этом краска ребра взаимно однозначно соответствует разряду двоичного вектора, идентифицирующего вершину гиперкуба. При переходе через ребро $\{v_i, v_j\}$, окрашенное в i -ю краску, двоичные коды вершин v_i и v_j отличаются в i -м разряде.
6. Конец.

Пример 3. Вложим граф G (рис. 5, а) в булево пространство H , используя рассмотренный алгоритм.

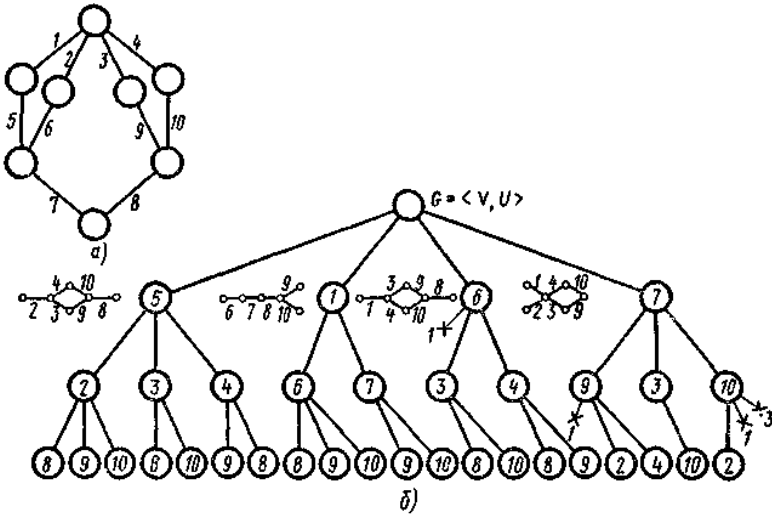


Рис. 5

1. Дерево, определяющее распределение пустых реберных подграфов, содержит 20 висячих вершин, каждая из которых соответствует пустому реберному подграфу (рис. 5, б).
2. Четыре пустых реберных подграфа образуют разрезы заданного графа: $E_1 = \{2, 5, 8\}$, $E_2 = \{1, 6, 8\}$, $E_3 = \{4, 7, 9\}$, $E_4 = \{3, 7, 10\}$.
3. Раскрашиваем ребра заданного графа G с помощью покрытия столбцов строками двоичной таблицы, в которой строке взаимно однозначно соответствует пустой реберный подграф, полученный в п.

2, столбцу — ребро, а на пересечении i -й строки с j -м столбцом стоит 1, если i -й пустой реберный подграф содержит j -е ребро, и 0 в противном случае (табл. 1).

Таблица 1

E_i	E_j									
	1	2	3	4	5	6	7	8	9	10
{2, 5, 8}		1			1			1		
{1, 6, 8}	1					1		1		
{4, 7, 9}				1			1		1	
{3, 7, 10}			1				1			1

Имеем четыре раскраски $R_i(G)$ графа G :

$$R_1(G) = \{\{2, 5, 8\}, \{1, 6\}, \{4, 7, 9\}, \{3, 10\}\}, R_2(G) = \{\{2, 5, 8\}, \{1, 6\}, \{4, 9\}, \{3, 7, 10\}\}, R_3(G) = \{\{2, 5\}, \{1, 6, 8\}, \{4, 7, 9\}, \{3, 10\}\}, R_4(G) = \{\{2, 5\}, \{1, 6, 8\}, \{4, 9\}, \{3, 7, 10\}\}.$$

4. Проверяем, образует ли каждое множество соцветных ребер разрез графа. В каждой раскраске имеются соцветные вершины, не являющиеся разрезом заданного графа: $\{1, 6\}, \{3, 10\}, \{4, 9\}, \{2, 5\}$. Для выполнения условий теоремы 9 необходимо удалить из графа G любой элемент множества $\{\{7\}, \{8\}, \{3, 4\}, \{9, 10\}, \{3, 10\}, \{4, 9\}\}$, выбрав соответствующую раскраску ребер графа G (рис. 6).

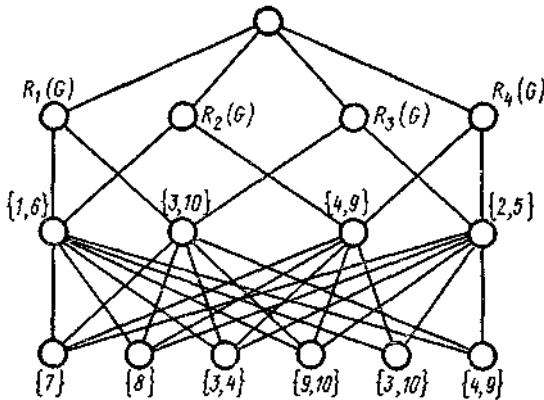


Рис 6

Если предметная интерпретация задачи требует реализовать это эквивалентирование не сужением сигнатуры, а расширением носителя, то соответствующие ребра заменяют цепью, при этом необходимо, чтобы не нарушился принцип четности. Задача определения

минимальных сужения сигнатуры или расширения носителя графа G сводится к покрытию соответствующей двоичной таблицы. В данном примере выбираем ребро 7 и, заменяя его цепью длины 3, получаем гомеоморфный граф G , эквивалентный заданный граф G и обладающий свойством сложимости в булево пространство. Используя первую раскраску $R_1(G)$, раскрашиваем ребра $\{2, 5, 8\}$ $\{1, 6, 11\}$, $\{4, 7, 9\}$, $\{3, 10, 12\}$ графа G (рис. 7, а).

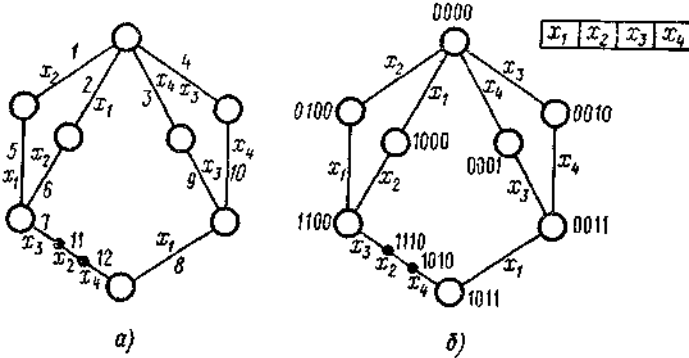


Рис. 7.

Сопоставим множества соцветных ребер разрядам двоичного вектора: $x_1 = \{2, 5, 8\}$, $x_2 = \{1, 6, 11\}$, $x_3 = \{4, 7, 9\}$, $x_4 = \{3, 10, 12\}$. Кодированием вершин графа G производим фактическое вложение графа G в булево пространство (рис. 7, б).

Теорема 10. Если G — двудольный граф и его степень равна $s(G)$, то
$$H(G) = s(G). \quad (7)$$

Теорема 11. Хроматический класс любого полного графа на n вершинах $H(K_n) = n$, если n — нечетное ($n \neq 1$), и $H(K_n) = n - 1$, если n — четное.

Приведем точный алгоритм минимальной раскраски вершин (ребер) графа G , т. е. алгоритм определения хроматического числа $\gamma(G)$ (хроматического класса $H(G)$):

1. Выделяем множество вершинно пустых (реберно пустых) подграфов графа G .
2. Строим двумерную таблицу, каждой строке которой сопоставим взаимно однозначно пустой подграф, столбцу - вершину (ребро): в клетке (i, j) записываем единицу, если j -я вершина (j -е ребро) содержится в i -м пустом подграфе, в противном случае клетку оставляем пустой.
3. Определяем покрытие столбцов строками. Каждое покрытие

порождает раскраску. Покрытие минимальной мощности определяет хроматическое число (хроматический класс) графа G .

В случае большого графа $G = \langle V, U \rangle$, когда мощность его носителя равна нескольким сотням и тысячам, рассмотрим следующий алгоритм, использующий частотные свойства графа:

1. Производя $|U|$ сравнений, определяем пару смежных вершин v_α и v_β графа $G = \langle V, U \rangle$, для которой функционал

$$\Phi(\{v_\alpha, v_\beta\}) = \frac{|Fv_\alpha \cap Fv_\beta|}{|Fv_\alpha| + |Fv_\beta|} \quad (8)$$

принимает максимальное значение. Равенство этого функционала нулю означает, что ребро $\{v_\alpha, v_\beta\}$ не входит в треугольник, и чем больше значение (8), тем в подграфе большей плотности это ребро может входить.

2. Найденную пару вершин раскрашиваем, и им взаимно однозначно сопоставляем столбцы в двумерной таблице, строкам которой взаимно однозначно сопоставляем вершины, смежные хотя бы с одной раскрашенной вершиной; в клетке (i, j) ставим 1, если i -я и j -я вершины смежны, и 0 в противном случае.

3. Выбираем строку с максимальным числом единиц.

4. Если для выделенной в п. 3 i -й строки найдется j -й столбец, на пересечении с которым находится 0, то соответствующую i -ю вершину раскрасим в j -ю краску и произведем склеивание соцветных по j -й краске вершин. В противном случае i -ю вершину раскрасим в новую краску, увеличивая количество столбцов в таблице на 1.

5. Если осталась хотя бы одна неокрашенная вершина, то переходим к п. 3, в противном случае — к п. 6.

6. Конец.

Число красок при раскраске вершин графа $G = \langle V, U \rangle$ равно количеству столбцов в итоговой таблице.

Пример 4. Раскрасим вершины графа $G = \langle V, U \rangle$ (рис. 8, а), используя рассмотренный алгоритм.

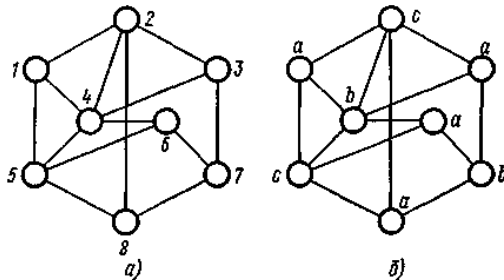


Рис. 8

1. Максимальное значение функционала $\varphi(\{v_\alpha, v_\beta\})$ согласно (8) равно 0,25 при рассмотрении вершин 1 и 4:

$$\max_{\alpha, \beta} \varphi(\{v_\alpha, v_\beta\}) = 0,25 \text{ при } v_\alpha = 1 \text{ и } v_\beta = 4.$$

2. Получаем таблицу (табл. 5).
3. Выбираем первую строку.
4. Раскрашиваем вершину 2 в новую краску, в результате получаем новую таблицу (табл. 6).
5. Неокрашенные вершины имеются; следовательно, переходим к п. 3.
3. Выбираем первую строку.
4. Вершина 3, соответствующая первой строке, соцветна вершине 1, раскрашиваем вершину 3 и склеиваем ее с вершиной 1. Получаем таблицу (табл. 7).

Таблица 5

$v_i \in \Gamma v_j$	v_j	
	1	4
2	1	1
3	0	1
5	1	1
6	0	1

Таблица 6

$v_i \in \Gamma v_j$	v_j		
	1	4	2
3	0	1	1
5	1	1	0
6	0	1	0
8	0	0	1

Таблица 7

$v_i \in \Gamma v_j$	v_j		
	{1,3}	4	2
5	1	1	0
6	0	1	0
7	1	0	0
8	0	0	1

Выполняя этот алгоритм последовательно, получаем таблицы (табл. 8, а, б, в).

Таблица 8, а

$v_i \in \Gamma v_j$	v_j		
	{1,3}	4	{2,5}
6	0	1	1
7	1	0	0
8	0	0	1

Таблица 8, б

$v_i \in \Gamma v_j$	v_j		
	{1, 3, 6}	4	{2, 5}
7	1	0	0
8	0	0	1

Таблица 8, в

$v_i \in \Gamma v_j$	v_j		
	{1, 3, 6}	{4, 7}	{2, 5}
8	0	1	1

Окончательно получена раскраска исходного графа $G = \langle V, U \rangle$ в три краски (рис. 8, б): $a = \{1, 3, 6, 8\}$, $b = \{4, 7\}$, $c = \{2, 5\}$.

Плотность графа G равна трем. Следовательно, полученная раскраска является минимальной.

2.3. Характеризация раскраски графов

Характеризационная проблема (важнейшая в дискретной математике) считается решенной, если найдены конструкции, определяющие смысл проводимого преобразования.

Рассмотрим структуру подграфов, определяющих хроматическое число графа.

Квазиполным графом $Q(q)$ называется граф Q , для минимальной раскраски которого необходимо q красок, а при раскраске любого собственного частичного графа Q' , $Q' \subset \subset Q$, достаточно q' красок, $q' < q$. Число красок q при этом называется квазиплотностью квазиполного графа Q .

Квазиплотность $q(G)$ графа G определяется выражением

$$q(G) = \max_i q_i(Q_i), Q_i \subset G. \quad (9)$$

Порядок $k(Q)$ квазиполного графа $Q(q)$

$$k(Q) = q(Q) - p(Q). \quad (10)$$

Порядок $k(G)$ графа G

$$k(G) = q(G) - p(G). \quad (11)$$

Условимся квазиполный граф Q плотности p и порядка k обозначать $Q(p, k)$, а граф с такими же характеристиками — $G(p, k)$.

Квазиполные графы $Q(2, 0)$, $Q(2, 1)$, $Q(2, 2)$, и $Q(2, 3)$ изображены на рис. 9, а, б, в, г.

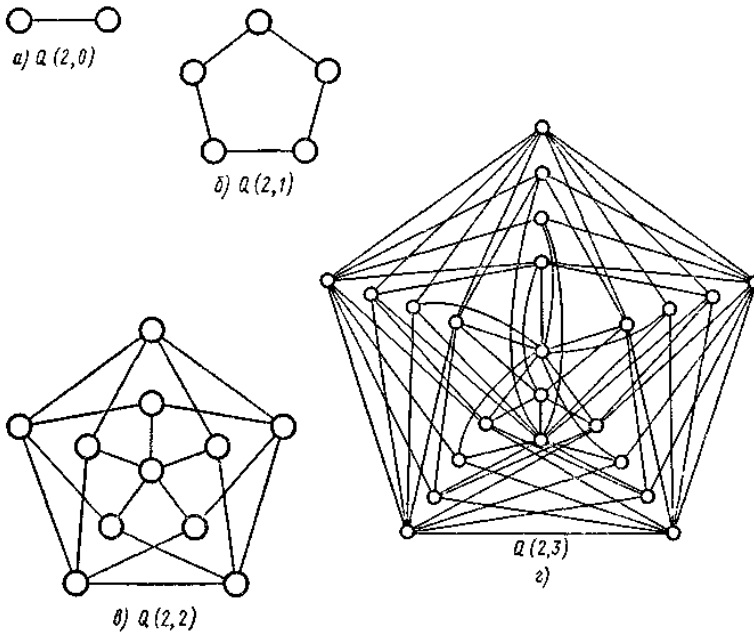


Рис. 9

Теорема 12 (основное свойство квазиполных графов). *Для любой вершины v_i квазиполного графа Q найдется такая раскраска его вершин, при которой не существует вершины v_α ($\alpha \neq i$), соцветной с v_i .* Доказательство проводим от противного. Пусть такая раскраска не существует, т. е. при каждой раскраске вершин графа Q найдется вершина v_α ($\alpha \neq i$), соцветная с v_i . Но тогда после удаления ребер, инцидентных вершине v_i , получаем, что число красок не изменяется, что противоречит определению квазиполного графа.

Следствие. При удалении любого ребра из квазиполного графа его квазиплотность уменьшается на 1.

Теорема 12 лежит в основе доказательства структуры квазиполного графа $Q(q)$ квазиплотности q ; квазиполный граф $Q(q)$ состоит из основания — квазиполного графа $Q(q-1)$, замещающего и замыкающего слоев. Вершины замещающего слоя согласно основному свойству квазиполных графов взаимно однозначно соответствуют вершинам основания, при этом соответствующие вершины смежны с одними и теми же вершинами, принадлежащими основанию $Q(q-1)$. Вершины замещающего слоя являются как бы «двойниками» вершин основания. Замыкающий слой представляет собой вершины, конусирующие вершины замещающего слоя. В пределе замыкающий слой представляет собой вершину, конусирующую весь замещающий слой.

Рассмотрим образование квазиполного графа $Q(2, k)$, обладающего экстремальными свойствами, т. е. мощности его носителя и сигнатуры являются минимальными по сравнению с другими квазиполными графами плотности 2 и порядка k . Согласно теореме 12, чем меньше мощность носителя или сигнатуры основания $Q(2, k-1)$, тем меньше соответствующие мощности квазиполного графа $Q(2, k)$. Основанием графа $Q(2, 1)$ является граф $Q(2, 0)$ (рис. 9, а), представляющий собой ребро $\{a, b\}$; замещающий слой — две вершины a' и b' , которые соответственно смежны с b и a . Замыкающий слой представляет собой вершину, конусирующую вершины a' и b' . В результате получен цикл длины 5 (рис. 9, б).

Аналогично, продолжая процесс построения графов и взяв в качестве основания $Q(2, 1)$ (рис. 9, б), получаем граф $Q(2, 2)$ (рис. 9, в); взяв граф $Q(2, 2)$ в качестве основания, получаем квазиполный граф $Q(2, 3)$ (рис. 9, г). Продолжая этот процесс, можно построить квазиполный граф $Q(2, k)$ с минимальными мощностями носителя и сигнатуры для любого k .

Определим зависимость минимальной мощности носителя графа $6(2, k)$

от его порядка k .

Имеем

$$\begin{aligned}
 k & |V_{\min}(Q(2, k))|, \\
 0 & 2, \\
 1 & 2 \cdot 2 + 1, \\
 2 & 2 \cdot (2 \cdot 2 + 1) + 1, \\
 3 & 2 \cdot (2 \cdot (2 \cdot 2 + 1) + 1) + 1, \\
 & \dots \\
 k & 2 \cdot \underbrace{(2 \cdot (\dots \cdot 2 \cdot (2 \cdot 2 + 1) + 1) \dots)}_{(k-1) \text{ скобка}} + 1.
 \end{aligned}$$

Следовательно,

$$\begin{aligned}
 |V_{\min}(Q(2, k))| &= 2 \cdot (2 \cdot (\dots \cdot 2 \cdot (2 \cdot 2 + 1) + 1) \dots) + 1 = 2^{k+1} + 2^{k-1} + \\
 &+ 2^{k-2} + 2^{k-3} + \dots + 2^1 + 2^0 = \sum_{i=0}^{k+1} 2^i - 2^k = \frac{2^{k+1} \cdot 2 - 2^0}{2-1} - 2^k = \\
 &\approx 4 \cdot 2^k - 2^k - 1 = 3 \cdot 2^k - 1.
 \end{aligned}$$

Таким образом, окончательно имеем

$$|V_{\min}(Q(2, k))| = 3 \cdot 2^k - 1. \tag{12}$$

Найдем зависимость минимальной мощности сигнатуры графа $Q(2, k)$ от его порядка. Согласно структуре квазиполных графов и их основному свойству, сумма мощностей сигнатуры основания $Q(2, k-1)$ и сигнатуры замещающего слоя равна утроенной мощности сигнатуры основания $Q(2, k-1)$ при построении графа $Q(2, k)$ с минимальной мощностью сигнатуры, а мощность сигнатуры замыкающего слоя, согласно (12), равна мощности носителя основания $3 \cdot 2^{k-1} - 1$.

Отсюда минимальная мощность сигнатуры $|U_{\min}(Q(2, k))|$ графа $Q(2, k)$ равна

$$\begin{aligned}
 & 1 \text{ при } k = 0, \\
 & 3 \cdot 1 + 3 \cdot 2^0 - 1 \text{ при } k = 1, \\
 & 3 \cdot (3 \cdot 1 + 3 \cdot 2^0 - 1) + 3 \cdot 2^1 - 1 \text{ при } k = 2, \\
 & 3 \cdot (3 \cdot (3 \cdot 1 + 3 \cdot 2^0 - 1) + 3 \cdot 2^1 - 1) + 3 \cdot 2^2 - 1 \text{ при } k = 3, \\
 & \dots \\
 & 3 \cdot \underbrace{(3 \cdot (3 \cdot (\dots \cdot 3 \cdot (3 \cdot 1 + 3 \cdot 2^0 - 1) \dots))}_{(k-1) \text{ скобка}} + 3 \cdot 2^1 - 1) + 3 \cdot 2^2 - \\
 & - 1) + \dots + 3 \cdot 2^{k-2} - 1) + 3 \cdot 2^{k-1} - 1 \text{ при } k = k.
 \end{aligned}$$

Последовательно упрощая выражение для $|U_{\min}(Q(2, k))|$, получаем

$$\begin{aligned}
 |U_{\min}(Q(2, k))| &= 3^k + 3^k \cdot 2^0 - 3^{k-1} + 3^{k-1} \cdot 2^1 - 3^{k-2} + 3^{k-2} \cdot 2^2 - \\
 &- 3^{k-3} + \dots + 3^1 \cdot 2^{k-1} - 3^0 = 3^k + \sum_{i=1}^k 3^i \cdot 2^{k-i} - \sum_{i=0}^{k-1} 3^i = 3^k + 3 \cdot (3^k - \\
 &- 2^k) - 0,5 \cdot (3^k - 1), \\
 2|U_{\min}(Q(2, k))| &= 7 \cdot 3^k - 6 \cdot 2^k + 1.
 \end{aligned} \tag{13}$$

Квазиполный граф $Q(p, k)$, обладающий рассматриваемыми экстремальными свойствами, получается суммированием квазиполного графа $Q(2, k)$ и полного графа плотности $p - 2$:

$$Q(2, k) + Q(p - 2, 0) = Q(p, k). \tag{14}$$

Обобщением (14) является следующая теорема.

Теорема 13. Сумма квазиполных графов является квазиполным графом:

$$\sum_i Q_i(p_i, k_i) = Q\left(\sum_i p_i, \sum_i k_i\right). \tag{15}$$

Согласно (14), минимальная мощность носителя квазиполного графа $Q(p, k)$ равна

$$|V_{\min}(Q(p, k))| = 3 \cdot (2^k - 1) + p. \tag{16}$$

Минимальная мощность сигнатуры графа $Q(p, k)$ представляет собой сумму минимальной мощности сигнатуры графа $Q(2, k)$ и минимальной мощности сигнатуры полного графа плотности $(p - 2)$, каждая вершина которого конусирует граф $Q(2, k)$. Согласно (12) и (13), имеем

$$\begin{aligned}
 |U_{\min}(Q(p, k))| &= 0,5 \cdot (7 \cdot 3^k - 6 \cdot 2^k + 1) + \\
 &+ 0,5 \cdot (p - 2) \cdot (p - 3) + (p - 2) \cdot (3 \cdot 2^k - 1).
 \end{aligned}$$

Окончательно получаем

$$2|U_{\min}(Q(p, k))| = 7 \cdot 3^k + 6 \cdot 2^k \cdot (p - 3) + (p - 3)^2 - p + 2. \tag{17}$$

Согласно (16), любой квазиполный граф первого порядка, имеющий плотность p , содержит не менее $p + 3$ вершин. Граф $Q(p, 1)$, имеющий ровно $p + 3$ вершин, является суммой цикла длины 5 и полного графа с плотностью $p - 2$:

$$Q(p, 1) = Q(2, 1) + Q(p - 2, 0).$$

Соотношения (16) и (17) справедливы для любых квазиполных графов, имеющих плотность 2 или 3, и для неразложимых по аддитивной операции «сумма» квазиполных графов. Структура этих графов была рассмотрена выше. Для разложимых квазиполных графов, у которых по теореме 13 плотность не меньше 4, и хотя бы у двух слагаемых $k_i > 0$ эти формулы неверны. Например, квазиполный граф $Q(4, 2)$, являющийся суммой двух графов $Q_a(2, 1)$ и $Q_b(2, 1)$ (рис. 10).

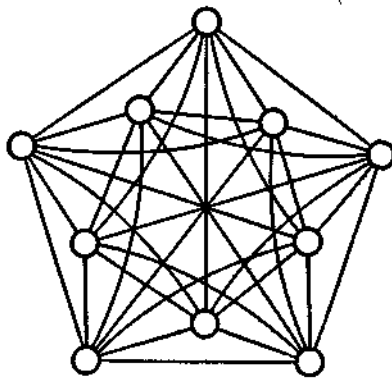


Рис.10

$Q(4, 2) = Q_a(2, 1) + Q_b(2, 1)$, имеет мощность носителя, равную 10, и мощность сигнатуры, равную 35, в то время как минимальные мощности носителя и сигнатуры неразложимого квазиполного графа $Q(4, 2)$ равны соответственно 13 и 43 [согласно (16) и (17)].

Теорема 14. *Хроматическое число $\gamma(G)$ графа G равно его квази-плотности $q(G)$:*

$$\gamma(G) = q(G). \quad (18)$$

Согласно определению квазиполного графа,

$$\gamma(G) \geq q(G).$$

Докажем, что в данном соотношении всегда имеет место равенство. Выделим все *невыключаемые квазиполные графы*, т. е. графы, для каждого Q_i из которых не найдется квазиполного графа Q_a такого, что $Q_i \subset \subset Q_a$. Согласно (9), среди выделенных подграфов имеется хотя бы один, для раскраски которого необходимо q красок. Покажем, что для минимальной раскраски остальной части $G \setminus Q$ достаточно тех же q красок. Действительно, если для минимальной раскраски подграфа $G \setminus Q$ потребуется хотя бы еще одна новая краска — $(q + 1)$ -я, то это означает, что в подграфе $G \setminus Q$ существуют замещающий и замыкающий слои, что приводит к образованию квазиполного графа квазиплотности $q + 1$. Последнее противоречит соотношению (9). \mathcal{V}

Используя формулы (16) - (18) для графов G_i с плотностью $\rho(G_i) < 4$ и для графов, которые не содержат квазиполные подграфы, разложимые по связке «сумма», можно предложить логарифмические оценки хроматического числа. Имеем:

$$\begin{aligned} \gamma(G) = q(G) &= \max_i (p(Q_i) + k(Q_i)) \leq \\ &\leq \max_i p(Q_i) + \max_i k(Q_i), \end{aligned} \quad (19, a)$$

$$\gamma(G) \leq p(G) + \max_i k(Q_i), \quad Q_i \subset G, \quad G = \langle V, U \rangle. \quad (19, б)$$

Согласно (16),

$$\max_i k(Q_i) \leq \left[\log_2 \frac{|V| - p(G)}{3} + 1 \right].$$

Отсюда

$$\gamma(G) \leq p(G) + \left[\log_2 \frac{|V| - p(G)}{3} + 1 \right]. \quad (20)$$

В выражении (17) $(p - 3)^2 - p + 2 > 0$, а при $p > 4, p = 4$ и любом k $6 \cdot 2^k - 1 > 0$, поэтому

$$\max_i k(Q_i) < \left[\log_3 \frac{2|U|}{7} \right], \quad p(G) \geq 4;$$

следовательно,

$$\gamma(G) \leq p(G) + \left[\log_3 \frac{2|U|}{7} \right], \quad p(G) \geq 4. \quad (21, a)$$

При $p(G) = 3$ имеем

$$\gamma(G) \leq 3 + \left[\log_3 \frac{2|U| + 1}{7} \right]. \quad (21, б)$$

При $p(G) = 2$ выражение (17) принимает вид

$$2|U_{\min} Q(2, k)| = 7 \cdot 3^k - 6 \cdot 2^k + 1 > 7 \cdot 3^k - 6 \cdot 3^k + 1 = 3^k + 1,$$

поэтому

$$\max_i k(Q_i) < \left[\log_3 (2|U| - 1) \right].$$

Отсюда при $p(G) = 2$ получаем

$$\gamma(G) \leq 2 + \left[\log_3 (2|U| - 1) \right]. \quad (21, в)$$

Согласно структуре квазиполного графа $Q(p, k)$, степень $s(v_i)$ каждой вершины v_i этого графа удовлетворяет неравенству

$$s(v_i) \geq q(Q) - 1 = p + k - 1. \quad (22)$$

Тогда по формуле (16) мощность сигнатуры $|U(Q(p, k))|$ неразложимого квазиполного графа удовлетворяет неравенству

$$|U(Q(p, k))| \geq 0,5 \cdot (3 \cdot (2^k - 1) + p) \cdot (p + k - 1). \quad (23)$$

Эта оценка достигается для полных графов, когда $k = 0$. Согласно (16), (19, а) и (23),

$$\gamma(G) \leq \left\lceil \frac{2 \cdot |U(\tilde{G})|}{|V(\tilde{G})|} \right\rceil + 1, \tilde{G} \subset G, \quad (24)$$

где подграф \tilde{G}^0 удовлетворяет всем неравенствам системы

$$\begin{aligned} |V(\tilde{G})| &\geq 3 \cdot (2^k - 1) + p(\tilde{G}), \\ 2|U(\tilde{G})| &\geq 7 \cdot 3^k + 6 \cdot 2^k \cdot (p(G) - 3) + (p(G) - 3)^2 - p(\tilde{G}) + 2, \\ s(v_i) &\geq p(\tilde{G}) + k - 1, v_i \in \tilde{G}, \end{aligned} \quad (25)$$

т. е. по своим ресурсам, как вершинному, так и реберному, и по локальной топологии (степеням вершин) подграф \tilde{G}^0 может включать квазиполный граф $Q(p, k)$, где

$$k = \left\lceil \log_2 \left(\frac{|V(\tilde{G}) - p(\tilde{G})|}{3} + 1 \right) \right\rceil. \quad (26)$$

Здесь $\lceil \cdot \rceil$ — целая часть числа. Очевидно, что

$$\left\lceil \frac{2|U(\tilde{G})|}{|V(\tilde{G})|} \right\rceil = \left\lceil \frac{1}{|V(\tilde{G})|} \sum_{v_i \in V(\tilde{G})} s(v_i) \right\rceil = \lceil s_{\text{средн}}(\tilde{G}) \rceil.$$

Отсюда имеем верхнюю оценку хроматического числа $\gamma(G)$ графа G :

$$\gamma(G) \leq \lceil s_{\text{средн}}(\tilde{G}) \rceil + 1, \quad (27)$$

где граф $\tilde{G}^0 \subset G$ удовлетворяет системе неравенств (25). Оценка (27) является более эффективной, чем ранее приведенные: $\gamma(G) \leq s_{\text{макс}}(G) + 1$ и для определенного класса графов оценка Брукса $\gamma(G) \leq s_{\text{макс}}(G)$.

Как композицию оценок (20), (21) и (27), получаем оценки хроматического числа $\gamma(G)$ графа $G = \langle V, U \rangle$:

при $p(G) > 4$

$$\begin{aligned} \gamma(G) &\leq \min \left(p(G) + \left\lceil \log_2 \left(\frac{|V(G)| - p(G)}{3} + 1 \right) \right\rceil, \right. \\ &p(G) + \left\lceil \log_3 \frac{2 \cdot |U(G)|}{7} \right\rceil, \\ &\left. \lceil s_{\text{средн}}(\tilde{G}) \rceil + 1 \right); \end{aligned} \quad (28, a)$$

при $p(G) = 3$

$$\begin{aligned} \gamma(G) &\leq \min \left(p(G) + \left\lceil \log_2 \left(\frac{|V(G)| - p(G)}{3} + 1 \right) \right\rceil, \right. \\ &p(G) + \left\lceil \log_3 \frac{2 \cdot |U(G)| + 1}{7} \right\rceil, \\ &\left. \lceil s_{\text{средн}}(\tilde{G}) \rceil + 1 \right); \end{aligned} \quad (28, b)$$

при $p(G) = 2$

$$\gamma(G) \leq \min \left(2 + \left[\log_2 \left(\frac{|V(G)| - 2}{3} + 1 \right) \right], \right. \\ \left. 2 + \lceil \log_3 (2 \cdot |U(G)| - 1) \rceil, \lceil s_{\text{средн}}(\tilde{G}) \rceil + 1 \right). \quad (28, \text{в})$$

Для вычисления оценки хроматического числа $\gamma(G)$ графа G на основании формулы (27) находим плотность $p(G)$ графа G и, уменьшая p от $p(G)$ до 2 с шагом 1, по формуле (26) находим k для каждого из этих значений. Максимальная сумма p и соответствующего значения k , для которых выполняются неравенства системы (25), согласно (19,а), определяют верхнюю оценку хроматического числа $\gamma(G)$. Если оцениваемый граф является квазиполным или вычисленная верхняя оценка совпадает с нижней, то она равна хроматическому числу.

Нижняя оценка хроматического числа

$$\gamma \geq(G) > p(G) \quad (29)$$

минорируется, как показано Майерсом и Лином, оценкой Геллера

$$\gamma(G) \geq \left[\frac{|V(G)|^2}{|V(G)|^2 - 2|U(G)|} \right], \quad (30)$$

или

$$\gamma(G) \geq \left[\frac{|V(G)|}{|V(G)| - s_{\text{средн}}(G)} \right]. \quad (31)$$

Оценки хроматического числа (20), (21), (27) и (28) минорируют известные верхние оценки $\gamma(G)$.

Пример 5. Рассмотрим граф $G = \langle V, U \rangle$ (рис. 11, а).

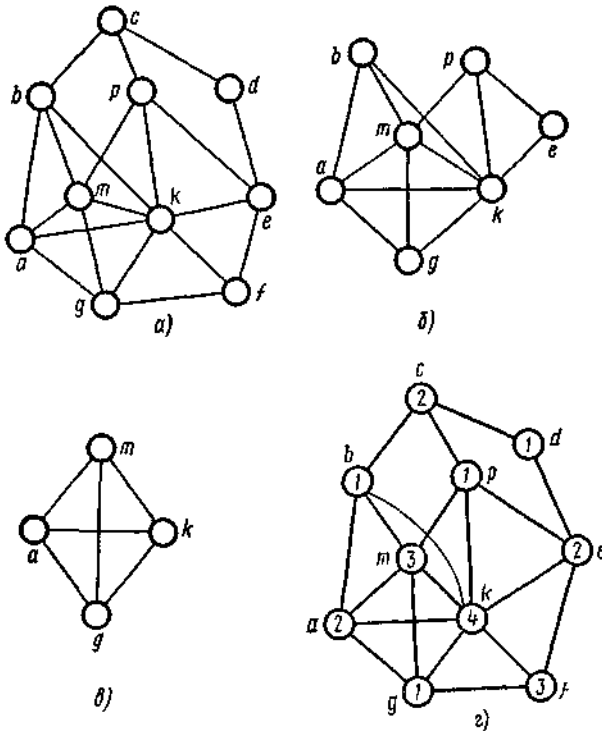


Рис. 11

По оценкам Брукса и Геллера, его хроматическое число $\gamma(G)$ заключено в отрезке $[1, 7]$:

$$\left[\frac{|V(G)|^2}{|V(G)|^2 - 2|U(G)|} \right] = \left[\frac{10^2}{10^2 - 2 \cdot 20} \right] = 1,$$

$$s_{\max}(G) \approx 7; 1 \leq \gamma(G) \leq 7.$$

Оценим хроматическое число $\gamma(G)$, используя полученные верхние оценки. Плотность графа G равна 4: $p(G) = 4$ Согласно (26),

$$k(G) = \left\lceil \log_2 \left(\frac{10 - 4}{3} + 1 \right) \right\rceil = 4.$$

Если включение $Q(4, 1)$ в заданный граф G возможно, то степень каждой вершины должна быть не меньше 4. Вершины c, d, f имеют степень, меньшую 4, и, следовательно, не входят в граф $Q(4, 1)$. При удалении этих вершин в полученном графе (рис. 11, б) k сохраняет свое значение:

$$k = \left\lceil \log_2 \left(\frac{7-4}{3} + 1 \right) \right\rceil = 1,$$

но при этом вершины b, p, e не могут входить в этот граф, так как их степень (рис. 11, б) меньше 4. Удаляем вершины b, p, e из графа G' (рис. 11, б), получаем граф, изображенный на рис. 11, в, который ни по вершинному ресурсу, ни по реберному, ни по локальной топологии не может содержать граф $Q(4, 1)$. Следовательно, $k_{\max} = 0$ при $p = 4$.

Уменьшим плотность на единицу и определим максимальное значение k :

$$k(G) = \left\lceil \log_2 \left(\frac{10-3}{3} + 1 \right) \right\rceil = 1.$$

Следовательно, заданный граф не может содержать квазиполный граф $Q(3, 2)$. Еще раз уменьшаем плотность на единицу и определяем максимальное значение k при $p = 2$. Имеем

$$k(G) = \left\lceil \log_2 \left(\frac{10-2}{3} + 1 \right) \right\rceil = 1.$$

Таким образом заданный граф не содержит даже квазиполного графа $Q(2, 2)$. Отсюда верхняя оценка хроматического числа $\gamma(G)$ графа G равна 4. Она совпала с нижней оценкой $p(G) \leq \gamma(G)$; следовательно, хроматическое число заданного графа также равно 4: $k_1 = \{b, d, p, g\}$,

$k_2 = \{a, c, e\}$, $k_3 = \{m, f\}$, $k_4 = \{k\}$. (рис. 11, в)

Таким образом, использование характеристики раскраски вершин графа позволило в данном случае определить его хроматическое число, не находя минимальное покрытие пустых подграфов вершинами графа. Для графа G , содержащего разложимый квазиполный граф, хотя бы с двумя слагаемыми, у которых $k_i > 0$, хроматическое число $\gamma(G)$, согласно теореме 13, равно сумме хроматических чисел квазиполных графов, которые являются слагаемыми графа $Q(q) \subset G$ максимальной квазиплотности. Поэтому

$$\gamma(G) = \max_i q_i(Q_i) = \max_i \sum_j q_{ij}(Q_{ij}),$$

$$Q_{ij} \subset Q_i, Q_i = \sum_j Q_{ij},$$

$$\gamma(G) = \max_i \sum_j (p_{ij}(Q_{ij}) + k_{ij}(Q_{ij})) \leq \max_i \sum_j p_{ij}(Q_{ij}) + \max_i \sum_j k_{ij}(Q_{ij}),$$

$$\gamma(G) \leq p(G) + \max_i \sum_j k_{ij}(Q_{ij}),$$

где $\max_i \sum_j k_{ij}(Q_{ij})$ определяется на основании (16), (17) и (22).

Пример 6. Определим хроматическое число $\gamma(G)$ графа $G = \langle V, U \rangle$ со следующими параметрами: $|V| = 90$, $|U| = 1000$, $p(G) = G$; распределение степеней вершин таково: 20 вершин имеет степень 5; 61 вершина — степень 21, остальные 9 вершин имеют степень, равную 86.

Рассмотрим случаи разложения 6 на слагаемые: 1) $6 = 6 + 0$; 2) $6 = 2 + 4$; 3) $6 = 2 + 2 + 2$; 4) $6 = 3 + 3$.

Случай 1. Определим максимальный порядок k_{\max} неразложимого квазиполного графа $Q(6, k_{\max})$ плотности 6. Согласно (16),

$$k_{\max} = \left\lfloor \log_2 \left(\frac{90 - 6}{3} + 1 \right) \right\rfloor = 4.$$

Установим, может ли заданный граф G содержать квазиполный граф $Q(6, 4)$ по реберным ресурсам и локальной топологии (по значениям степеней вершин). Имеем:

$$\begin{aligned} |U| &> 0,5 \cdot (7 \cdot 3^4 + 6 \cdot 2^4 \cdot (6 - 3) + (6 - 3)^2 - \\ &- 6 + 2) = 0,5 \cdot (567 + 288 + 5) = 430, \\ |V_{\min}(Q(6, 4))| &= 3(2^4 - 1) + 6 = 51, \\ (\forall v_i (v_i \in Q(6, 4))) (s(v_i) &\geq 6 + 4 - 1 = 9). \end{aligned}$$

Таким образом получен положительный ответ. Следовательно, определена верхняя оценка: $\gamma(G) \leq 10$.

Случай 2. Квазиполный граф $Q(q_{\max})$ максимальной квазиплотности можно разложить на два слагаемых:

$$Q_a(2, k_a) + Q_b(4, k_b),$$

при этом $\gamma(G) \leq 6 + \max_{a,b}(k_a + k_b)$. Определим $\max_{a,b}(k_a + k_b)$. Для этого определяем

$$\begin{aligned} k_a &= \left\lfloor \log_2 \left(\frac{(90 - 4) - 2}{3} + 1 \right) \right\rfloor = 4, \\ |V_{\min}(Q(2, 4))| &= 3 \cdot (2^4 - 1) + 2 = 47. \end{aligned}$$

Второе слагаемое содержит 43 вершины:

$$90 - 47 = 43.$$

Тогда

$$k_b = \left[\log_2 \left(\frac{43 - 4}{3} + 1 \right) \right] = 3,$$

$$|V_{\min}(Q(4, 3))| = 3 \cdot (2^3 - 1) + 4 = 25,$$

$$(\forall v_i (v_i \in Q_a(2, 4))) (s(v_i) \geq p_a + k_a - 1 + \\ + |V_{\min}(Q_b(4, 3))| = 2 + 4 - 1 + 25 = 30),$$

$$(\forall v_i (v_i \in Q_b(4, 3))) (s(v_i) \geq p_b + k_b - 1 + \\ + |V_{\min}(Q(2, 4))| = 4 + 3 - 1 + 47 = 53).$$

Вершин с такими степенями в графе G нет. Продолжая аналогичные вычисления, находим $\max_{a,b} (k_a + k_b) = 2 + 2 = 4$, т. е. в графе G может

содержаться квазиполный граф $Q(10)$, разложимый в виде суммы:

$$Q_a(2, 2) + Q_b(4, 2).$$

Действительно,

$$|V_{\min}(Q_a(2, 2))| = 3 \cdot (2^2 - 1) + 2 = 11,$$

$$|V_{\min}(Q_b(4, 2))| = 3 \cdot (2^2 - 1) + 4 = 13,$$

$$(\forall v_i \in Q_a(2, 2)) (s(v_i) \geq 2 + 2 - 1 + 13 = 16),$$

$$(\forall v_i \in Q_b(4, 2)) (s(v_i) \geq 4 + 2 - 1 + 11 = 16),$$

$$|U_{\min}(Q_a(2, 2))| + |U_{\min}(Q_b(4, 2))| +$$

$$+ |V_{\min}(Q_a(2, 2))| \cdot |V_{\min}(Q_b(4, 2))| =$$

$$= 0,5 \cdot (7 \cdot 3^2 + 6 \cdot 2^2 \cdot (2 - 3) + (2 - 3)^2 - 2 + 2) +$$

$$+ 0,5 \cdot (7 \cdot 3^2 + 6 \cdot 2^2 \cdot (4 - 3) + (4 - 3)^2 - 4 + 2) + 11 \cdot 13 = 206.$$

В этом случае хроматическая оценка $\gamma(G)$ не изменяется. Остальные случаи предлагаем рассмотреть читателю самостоятельно.

Окончательно получаем $\max_{a,b} (k_a, k_b) = 4$.

Следовательно, согласно (29) и (32), $6 \leq \gamma(G) \leq 10$.

Используя оценки Геллера — Брукса, мы получили бы отрезок: $1 \leq \gamma(G) \leq 86$.

Согласно (18), запрещенными фигурами раскраски вершин графа в α красок являются квазиполные графы квазиплотности $\alpha + 1$.

Приведем свойства квазиполных графов.

1°. Свойство включения

$$Q(p, k) \supseteq Q(i, j), \quad i = 1, \dots, p, \quad j = 0, \dots, k. \quad (33)$$

2°. Свойство идемпотентности замещающего слоя. При замещении замещающего слоя остается в классе квазиполных графов, причем такое замещение можно проводить многократно (рис. 12).

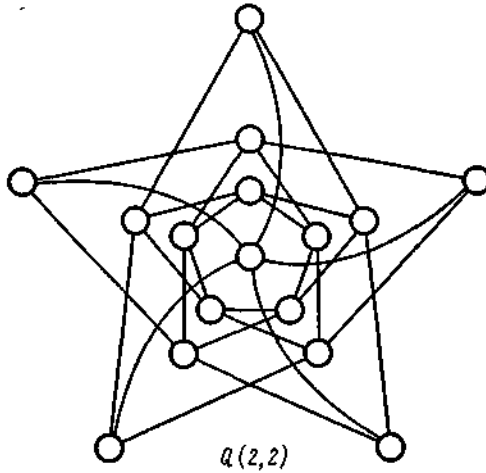


Рис.12

Теорема 15. При подстановке вместо каждой вершины v_i квазиполного графа $Q(q)$ одного из элементов множества $\{Q_i(q_i)/q_i (Q_i) = \alpha\}$ получаем квазиполный граф $Q(\alpha \cdot q)$.

Теорема 16 (теорема М. В. Горбатовой). Квазиполный граф $Q(p, k)$, $k > 1$, не обладает свойством реберности.

Если граф обладает свойством реберности, то его порядок не превышает 1. При построении реберно производного графа G по графу G ребра, инцидентные одной и той же вершине графа G , соответствуют вершинам полного подграфа графа G .

Следовательно,

$$s \leq H(G) \leq s + 1, \quad (34)$$

где $H(G)$ — хроматический класс графа G , s — степень графа G , $s = \max_i s(v_i)$, $v_i \in G$.

Для мультиграфа

$$s \leq H(G) \leq s + p, \quad (35)$$

где p — параллельность мультиграфа, т. е. максимальное число параллельных ребер.

В соотношении (35) степень графа G и его параллельность удовлетворяют следующим соотношениям (рис. 13):

$$\begin{cases} s \equiv 0 \pmod{p}, \\ s \neq p, \end{cases} \quad (36)$$

где запись $s \equiv 0 \pmod{p}$ означает, что число s сравнимо по модулю p с числом 0.

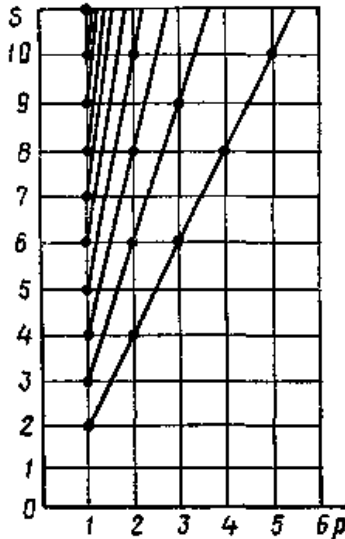


Рис.13

Согласно (35) и (36), получаем оценку Шеннона.

$$H(G) \leq \left\lceil \frac{3}{2} s \right\rceil. \quad (337)$$

Теорема 17. *Квазиполный граф квазиплотности q содержит подграф, гомеоморфный полному подграфу, плотность которого равна q .*

Эта теорема позволяет связать раскраску графов со свойством их вложения в поверхности.

Теорема 18. *Граф $G(p, k)$, $k > 1$, $p > 2$ невлжим в плоскость.*

Действительно, граф $G(p, k)$ содержит квазиполный подграф $Q(3, 2)$, который, в свою очередь, содержит подграф $G_\Gamma(F(5))$, гомеоморфный полному подграфу плотности, равной 5:

$$G(p, k) \supset Q(3, 2) \supset G_\Gamma(F(5)).$$

Следовательно, согласно теореме Понтрягина, граф G невлжим в плоскость.

Теорема 19. *Граф $G(p, k)$, $p > 3$, $k > 0$, невлжим в плоскость.*

Действительно, граф $G(p, k)$ содержит подграф $Q(4, 1)$, основание которого и вершина из замещающего слоя входят в подграф $G_\Gamma(F(5))$, гомеоморфный полному подграфу плотности, равной 5:

$$G(p, k) \supseteq Q(4, 1) \supseteq G_\Gamma(F(5)).$$

Следовательно, согласно теореме Понтрягина, граф G невлжим в плоскость.

На основании теорем 18 и 19 получаем доказательство проблемы четырех красок (теорема 20).

Теорема 20. *Хроматическое число планарного графа не превышает четырех.*

Согласно теореме Понтрягина, планарный граф G не содержит подграф $G_\Gamma(F(5))$, гомеоморфный полному подграфу плотности, равной 5: $G \not\supseteq G_\Gamma(F(5))$. Следовательно, этот граф не содержит подграфы $Q(3, 2)$ и $Q(4, 1)$. Отсюда хроматическое число $\gamma(G)$, согласно теореме 12, не превышает четырех.

2.4. Оценки хроматических чисел

Ранее было введено понятие кликового числа $\rho(G)$ графа G (как наибольшего числа вершин в полном порожденном подграфе графа G) и было отмечено, что поскольку между кликами графа G и максимальными независимыми множествами дополнительного графа G^c существует взаимно однозначное соответствие, то справедливы равенства $\rho(G) = \alpha(G^c)$ и $\rho(G^c) = \alpha(G)$

2.4.1. Нижние оценки для $\gamma(G)$

Очевидно, поскольку по крайней мере $\rho(G)$ цветов требуется для раскраски соответствующей клики графа G (той самой клики, которая «определяет» кликовое число графа G), что $\rho(G)$ является нижней оценкой хроматического числа, т. е.

$$\gamma(G) \geq \rho(G). \quad (38)$$

Более того, Зыков доказал, что эта оценка — точная и что разность $\gamma(G) - \rho(G)$ может быть сколько угодно большой. В действительности Татт показал, что можно построить граф G , который не содержит даже полного подграфа третьего порядка (т. е. $\rho(G) = 2$) и который будет иметь произвольно большое заданное значение хроматического числа. На рис. 14 изображен граф с $\rho(G) = 2$ и $\gamma(G) = 5$.

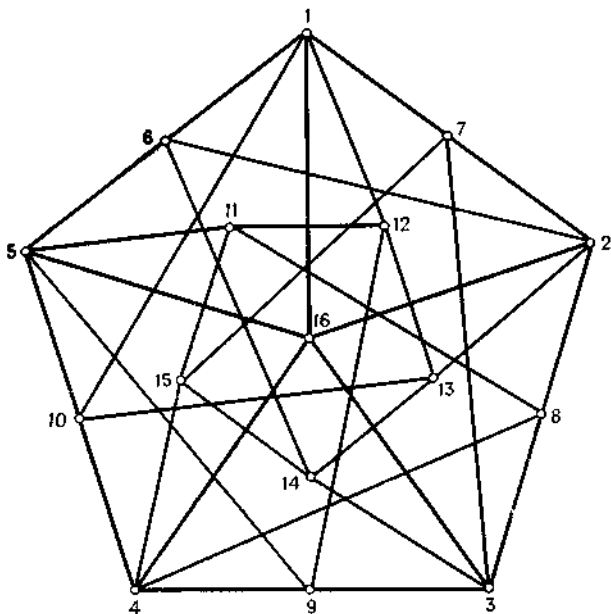


Рис. 14. Граф с $\rho(G) = 2$ и $\gamma(G) = 5$.

Поскольку число $\alpha(G)$ равно мощности наибольшего множества попарно несмежных вершин графа G , то оно совпадает также с мощностью наибольшего множества вершин в G , которые могут быть окрашены в один цвет, и, следовательно,

$$\gamma(G) \geq \left\lceil \frac{n}{\alpha(G)} \right\rceil, \quad (39)$$

где n — число вершин графа G , а $\lceil x \rceil$ обозначает наибольшее целое число, не превосходящее числа x . (Число $\lceil x \rceil$ называется «целой частью числа x » и часто обозначается через $[x]$).

Если через $\gamma(\bar{G})$ обозначить хроматическое число дополнения графа G , то можно записать следующие два неравенства, полученные Нордхаузом и Гаддумом:

$$\gamma(G) \geq \lfloor 2\sqrt{n} \rfloor - \gamma(\bar{G}) \quad (40)$$

и

$$\gamma(G) \geq \frac{n}{\gamma(\tilde{G})}, \quad (41)$$

где $\lfloor x \rfloor$ означает наименьшее целое число, которое не меньше x . Это число часто обозначается символом $\lfloor x \rfloor$.

Еще одна нижняя оценка для $\gamma(G)$ предложена Геллером:

$$\gamma(G) \geq \frac{n^2}{n^2 - 2m}. \quad (42)$$

Майерс и Лин показали, что оценка из (38) равномерно мажорирует приведенную выше, и, следовательно, единственное преимущество оценки (42) состоит в том, что ее проще вычислять, чем оценку (38).

2.4.2. Верхние оценки для $\gamma(G)$

Нижние оценки хроматического числа более интересны, чем верхние, поскольку (если они достаточно близки к истинному значению) они могут быть использованы в процедуре вычисления $\gamma(G)$, включающей дерево поиска. В то же время верхние оценки хроматического числа подобного применения не находят. Тем не менее в литературе приводятся формулы для вычисления верхних оценок хроматического числа; так Бруксом предложена следующая легко вычисляемая оценка:

$$\gamma(G) \leq 1 + \max_{x_i \in X} [d(x_i) + 1]. \quad (43)$$

Другие достаточно точные верхние оценки, также использующие степени вершин графа, можно найти у Уэлша и Вилфа и Секереша.

Верхние оценки, связывающие значения $\gamma(G)$ и $\gamma(\tilde{G})$, приводятся у Нордхауза и Гаддума:

$$\gamma(G) \leq n + 1 - \gamma(\tilde{G}) \quad (44)$$

и

$$\gamma(G) \leq \left\lceil \left(\frac{n+1}{2} \right)^2 \right\rceil / \gamma(\tilde{G}). \quad (45)$$

Оценки, даваемые формулами (40), (41), (44) и (45), являются наилучшими в том смысле, что можно построить графы, на которых эти оценки достигаются. В большинстве случаев, однако, они столь неточны, что не имеют никакой практической значимости.

2.5. Точные алгоритмы раскраски

В этом разделе рассматриваются некоторые алгоритмы, которые являются точными в том смысле, что они гарантируют нахождение

оптимальной раскраски и истинного значения хроматического числа для любого графа.

2.5.1. Метод динамического программирования

1. Максимальные r -подграфы. Порожденный подграф $\langle S_r [G] \rangle$ графа $G = (X, \Gamma)$, где $S_r [G] \subseteq X$, называется r -подграфом, если он r -хроматический. Если не существует такого множества H , что $H \supset S_r (G)$ и подграф $\langle H \rangle$ является r -хроматическим, то подграф $\langle S_r [G] \rangle$ называется *максимальным r -подграфом* графа G . Очевидно, что для фиксированного значения r , вообще говоря, существует много различных максимальных r -подграфов данного графа G . Подграф, у которого множество вершин совпадает с некоторым максимальным независимым множеством в G и который не имеет ребер (т. е. является вполне несвязным), есть максимальный 1-подграф графа G , поскольку в G нет 1-хроматического подграфа, имеющего большее множество вершин, чем у рассматриваемого подграфа.

Хроматическое число графа G можно определить как такое наименьшее число r , что $S_r [G] = X$ по крайней мере для одного из максимальных r -подграфов графа G .

Теорема 1. *Если граф G является r -хроматическим, то он может быть раскрашен с использованием r (или меньшего числа) красок с помощью следующей процедуры: сначала в один цвет окрашивается некоторое максимальное независимое множество $S_1[G]$, затем окрашивается в следующий цвет множество $S_1 [X - S_1[G]]$ и т. д. до тех пор, пока не будут раскрашены все вершины.*

Доказательство. Тот факт, что такая раскраска, использующая только r цветов, всегда существует, может быть установлен следующим образом. Пусть существует раскраска в r цветов, такая, что одно или больше множеств, окрашенных в один и тот же цвет, не являются максимальными независимыми множествами в смысле, упомянутом выше. Перенумеруем цвета произвольным способом. Очевидно, что мы можем всегда покрасить в цвет 1 те вершины (пусть это множество \bar{V}_1), которые не были окрашены в этот цвет и которые образуют максимальное независимое множество вместе с множеством V_1 всех вершин графа, уже окрашенных в цвет 1. Эта новая раскраска возможна потому, что никакая вершина из множества \bar{V}_1 не является смежной ни с какой вершиной из \bar{V}_1 и, следовательно, всякая вершина, которая смежна хотя бы с одной вершиной из \bar{V}_1 , окрашена

в цвет, отличный от цвета 1, и поэтому не затрагивается процедурой перемены цвета вершин из \bar{V}_1 . Рассматривая теперь подграф

$\langle X - V_1 \cup \bar{V}_1 \rangle$ и проводя с ним аналогичные манипуляции, мы окрасим в цвет 2 какое-то (новое) максимальное независимое множество и т. д.

Раскраску указанного в теореме вида будем называть *оптимальной независимой раскраской*.

2. Рекуррентные соотношения. Вышеприведенную теорему можно использовать для получения рекуррентного соотношения, связывающего максимальные r -подграфы графа с его максимальными $(r - 1)$ -подграфами.

Пусть $Q_{r-1}[G]$ — семейство максимальных $(r - 1)$ -подграфов графа G , а $S_{r-1}^j[G]$ — множество вершин j -го подграфа из семейства $Q_{r-1}[G]$. Множество $S_1^k[G^j]$ является тогда множеством вершин k -го максимального 1-подграфа (независимого множества) графа $G^j \equiv \langle X - S_{r-1}^j[G] \rangle$, образованного вершинами графа G , не попавшими в $(r - 1)$ -подграф $\langle S_{r-1}^j[G] \rangle$.

Тогда следующим образом можно описать семейство всех максимальных r -подграфов графа G .

Строим множества H^i :

$$H^i = S_{r-1}^j[G] \cup S_1^k[G^j], \quad (46)$$

$j = 1, 2, \dots, q_{r-1}$ и $k = 1, 2, \dots, q_1^j$, где q_{r-1} и q_1^j — соответственно число $(r - 1)$ -подграфов и число 1-подграфов.

Семейство максимальных r -подграфов содержится в семействе Θ множеств H^i ($i = 1, 2, \dots, q_1 \cdot q_{r-1}$) и оно может быть получено из него исключением таких множеств семейства, которые содержатся в других множествах.

Итак, $\Theta_r[G]$ можно описать следующим образом:

$$\Theta_r[G] = \{H^i \mid H^i \in \Theta \text{ и } H^i \not\subseteq H^j \text{ для любого } H^j \in \Theta, j \neq i\}. \quad (47)$$

3. Алгоритм, основанный на рассмотрении максимальных r -подграфов. В предыдущем разделе было указано, что хроматическое число графа G является таким наименьшим значением r , при котором $S_r[G]$ — множество вершин некоторого максимального r -подграфа — совпадает с X (множеством вершин графа G). Поэтому рекуррентные соотношения (46) и (47) могут быть использованы для последовательного построения максимальных 1-подграфов, 2-подграфов и т. д. и нахождения хроматического числа графа G , нужно просто на каждом шаге указанной процедуры проверять, не

содержится ли множество вершин графа G в каком-нибудь из построенных подграфов. Алгоритм, описанный в этом разделе, можно толковать как алгоритм динамического программирования или как древовидный поиск (с приоритетом) по ширине.

Ниже приводится описание алгоритма, позволяющего находить хроматическое число графа G и раскраску, соответствующую этому числу.

Шаг 1. Положить $r = 1$. Найти множества вершин $S_r^j [G]$, $j = 1, \dots, q_r$, максимальных r -подграфов графа G . (Считаем, что имеется q_r таких множеств.) Пусть $Q = \{ S_r^j [G] \mid j = 1, \dots, q_r \}$. Положить $j = 1$.

Шаг 2. Найти максимальное независимое множество $S_1 [G^j]$ графа $G^j = \langle X - S_2^j [G] \rangle$. Если такое множество существует, то перейти к шагу 3. Если все такие множества уже найдены, то перейти к шагу 6.

Шаг 3. Вычислить $S = S_r^j [G] \cup S_1 [G^j]$.

Шаг 4. Если $S = X$, то остановиться. Число $r+1$ есть хроматическое число графа G . Подмножества, включенные в множество S , дают требуемую раскраску. (Эти подмножества накапливаются по мере их введения и хранятся отдельно с соответствующими отметками (маркерами)). Если $S \neq X$, то перейти к шагу 5.

Шаг 5. (i) Если $S \subseteq S'$ для некоторого $S' \in Q$, то перейти к шагу 2.

(ii) Если $S \supset S'$ для некоторых $iS'' \in Q$, то положить $Q = Q - \{S'\}$ по всем таким S' из Q . Положить $Q = Q \cup \{S\}$ и перейти к шагу 2.

(iii) Если не выполняется ни одно из условий (i) и (ii), указанных выше, то положить $Q = Q \cup \{S\}$ и перейти к шагу 2.

Шаг 6. Если $j < q_r$, то положить $j = j + 1$ и перейти к шагу 2.

Если $j = q_r$, то положить $j = 1$, $r = r + 1$, $q_r =$ числу множеств в Q и перейти к шагу 2.

Если работу алгоритма не завершать при первом выполнении условия $S = X$ на шаге 4, то алгоритмический процесс будет продолжаться до получения раскраски в $r + 1$ цветов, если такая раскраска существует. Следует отметить, однако, что описанный алгоритм не дает полного перечисления всех возможных раскрасок в $r + 1$ цветов, а только порождает оптимальные независимые раскраски. Такие раскраски могут оказаться только небольшой частью всех возможных раскрасок в $r + 1$ цветов.

4. Пример. Рассмотрим семивершинный граф G , изображенный на рис. 15.

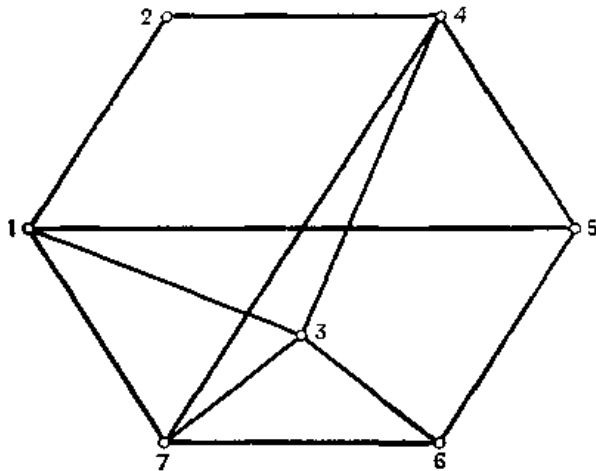


Рис. 15. Граф из примера 4.

Шаг 1. Множествами вершин максимальных 1-подграфов являются: $S^1_1[G] = \{1, 4, 6\}$; $S^2_1[G] = \{2, 3, 5\}$; $S^3_1[G] = \{2, 5, 7\}$; $S^4_1[G] = \{2, 6\}$. Следовательно. $Q_1 = 4$ и $Q = \{S^1_1[G], S^2_1[G], S^3_1[G], S^4_1[G]\}$.

Применяем (необходимое число раз) шаги 2—5 алгоритма:

Для $S^1_1[G]$

Шаг 2. $G^1 = \langle X - S^1_1[G] \rangle = \langle \{2, 3, 5, 7\} \rangle$; $S_1[G^1] = \{2, 3, 5\}$.

Шаг 3. $S = \{1, 4, 6 \uparrow 2, 3, 5\}$.

Шаг 5. $Q = \{\{1, 4, 6 \uparrow 2, 3, 5\}\}$.

Шаг 2. $S_1 [G^1] = \{2, 5, 7\}$.

Шаг 3. $S = \{1, 4, 6 \uparrow 2, 5, 7\}$.

Шаг 5. $Q = \{\{1, 4, 6 \uparrow 2, 3, 5\}, \{1, 4, 6 \uparrow 2, 5, 7\}\}$.

Для $S_1^2 [G]$

Шаг 2. $G^2 = \langle \{1, 4, 6, 7\} \rangle$; $S_1 [G^2] = \{1, 4, 6\}$.

Шаг 3. $S = \{2, 3, 5 \uparrow 1, 4, 6\}$, исключено в соответствии с шагом 5 (i).

Шаг 2. $S_1 [G^2] = \{7\}$.

Шаг 3. $S = \{2, 3, 5 \uparrow 7\}$.

*Шаг 5. $Q = \{\{1, 4, 6 \uparrow 2, 3, 5\}, \{1, 4, 6 \uparrow 2, 5, 7\}, \{2, 3, 5 \uparrow 7\}\}$.

Для $S_1^3 [G]$

Шаг 2. $G^3 = \langle \{1, 3, 4, 6\} \rangle$; $S_1 [G^3] = \{1, 4, 6\}$.

Шаг 3. $S = \{2, 5, 7 \uparrow 1, 4, 6\}$, исключено в соответствии с шагом 5(i).

Шаг 2. $S_1 [G^3] = \{3\}$.

Шаг 3. $S = \{2, 5, 7 \uparrow 3\}$, исключено в соответствии с шагом 5(i).

Для $S_1^4 [G]$

Шаг 2. $G^4 = \{1, 3, 4, 5, 7\}$, $S_1 [G^4] = \{1, 4\}$.

Шаг 3. $S = \{2, 6 \uparrow 1, 4\}$, исключено в соответствии с шагом 5(i).

Шаг 2. $S_1 [G^4] = \{3, 5\}$.

Шаг 3. $S = \{2, 6 \uparrow 3, 5\}$, исключено в соответствии с шагом 5(i).

Шаг 2. $S_1 [G^4] = \{5, 7\}$.

Шаг 3. $S = \{2, 6 \uparrow 5, 7\}$, исключено в соответствии с шагом 5 (i).

Таким образом, в конце первой итерации, использующей шаги 2—5 алгоритма, получается семейство Q множеств $S_2^j [G]$, соответствующее максимальным 2-подграфам и показанное на шаге, отмеченном звездочкой. Следует обратить внимание на то, что 5 из восьми полученных множеств исключены в результате применения первого правила из шага 5 алгоритма.

Действуя аналогичным образом дальше, можно построить максимальные 3-подграфы и т. д. В действительности уже следующее порожденное множество $S_3^1 [G] = \{1, 4, 6 \uparrow 2, 5, 3 \uparrow 7\}$ будет удовлетворять условию $S_3^1 [G] = X$ на шаге 4. Следовательно, хроматическое число графа равно 3 и оптимальная раскраска задается

следующим разбиением (множества X) : $\{1, 4, 6\}$, $\{2, 5, 3\}$, $\{7\}$. Если применять алгоритм дальше, то будет получена еще одна возможная раскраска:

$$S_3^2[G] = \{1, 4, 6 \uparrow 2, 5, 7 \uparrow 3\} = X.$$

Все другие множества $S_r^j[G]$ либо содержатся в двух предшествующих множествах, либо не содержат все вершины из X .

Заметим, что такие раскраски, как $\{5, 3\}$, $\{1, 4, 6\}$, $\{2, 7\}$, хотя и возможны, но не являются оптимальными независимыми и не порождаются описанным выше алгоритмом.

2.5.2. Формулировка задачи о раскраске на языке 0-1-программирования

Пусть q — какая-нибудь верхняя оценка хроматического числа графа G . Эта верхняя оценка может быть одной из оценок, данных в разд. 2 этой главы, или может быть равна числу цветов, получающемуся в одном из тех эвристических методов решения задачи о раскраске, которые подробно будут описаны позже.

Пусть $\Xi = [\xi_{ij}]$ — матрица раскраски графа (задающая некоторую конкретную раскраску вершин графа), так что

$$\xi_{ij} = \begin{cases} 1, & \text{если вершина } x_i \text{ окрашена в цвет } j, \\ 0 & \text{в противном случае.} \end{cases}$$

(Здесь не предполагается, что смежные вершины должны быть окрашены в разные цвета. Если же это условие выполняется, то раскраска называется *допустимой*).

Если $A = [a_{ik}]$ — матрица смежности графа G с диагональными элементами, равными 0, то следующие два условия гарантируют допустимость раскраски вершин графа G (т. е. допустимость матрицы раскраски $[\xi_{ij}]$):

$$\sum_{j=1}^q \xi_{ij} = 1 \quad (\text{для всех } i = 1, \dots, n), \quad (48)$$

$$L \cdot (1 - \xi_{ij}) - \sum_{k=1}^n a_{ik} \xi_{kj} \geq 0 \quad \begin{cases} \text{для всех } i = 1, \dots, n \\ \text{и} \quad j = 1, 2, \dots, q. \end{cases} \quad (49)$$

Условие (48) обеспечивает раскраску вершины в один и только один цвет.

В условии (49) L — очень большое положительное число (любое целое, большее чем n). Если вершина x_i окрашена в цвет j (т. е. $\xi_{ij} = 1$), то первый член в (49) равен 0. Тогда и второй член должен быть равен 0, чтобы выполнялось неравенство, поскольку числа a_{ik} и ξ_{kj}

неотрицательны. Таким образом, условие (49) обеспечивает допустимость раскраски, т. е. если вершина x_i окрашена в цвет j , то нет смежной с x_i вершины того же цвета. Если вершина x_i окрашена в цвет, отличный от j ($\xi_{ij} = 0$), то первый член в (49) равен L . Поскольку второй член в (49) не может, очевидно, достигнуть значения L (его наибольшее значение равно в действительности $n - 1$), то какое бы число вершин x_k , смежных с вершиной x_i , ни было окрашено в цвет j , неравенство (49) по-прежнему будет выполняться. Заметим, что если вершины x_k' и x_k'' смежны с x_i , а также смежны между собой, то условие (49), записанное для вершины x_i , не будет препятствовать раскраске x_k' и x_k'' в один и тот же цвет j . Однако, записав условие (49) для x_k' (или x_k''), мы обеспечим тем самым раскраску этих двух вершин (x_k' и x_k'') в разные цвета.

Пусть теперь каждому цвету j сопоставлен штраф p_j , выбранный так, что

$$P_{i+j} > h \cdot p_j \text{ (штраф } p_j \text{ принят равным единице)} \quad (50)$$

и где h есть верхняя оценка для наибольшего числа вершин в графе, которые могут быть окрашены в один цвет, т. е. h — произвольное число, большее чем $\alpha(G)$ — число независимости графа. При отсутствии лучшей оценки можно положить $h = n$, не проводя лишних вычислений.

Задача раскраски вершин графа с использованием наименьшего числа цветов может быть сформулирована следующим образом:
минимизировать

$$z = \sum_{j=1}^q \sum_{i=1}^n p_j \xi_{ij} \quad (51)$$

при ограничениях (48) и (48). Минимизация выражения (51) обеспечивает выполнимость следующего условия: цвет $j + 1$ не будет использован в раскраске вершин, если цвета от 1 до j достаточны для допустимой раскраски. Матрица (раскраски графа) $[\xi_{ij}^*]$, которая дает решение приведенной выше задачи линейного 0-1-программирования, определяет оптимальную раскраску, а используемое при этом число цветов равно хроматическому числу графа.

Берг вместо условия (49) предложил следующее:

$$\sum_{i=1}^n b_{ik} \xi_{ij} \leq 1 \begin{cases} \text{для всех } k = 1, 2, \dots, m \\ \text{и} & j = 1, 2, \dots, q, \end{cases} \quad (52)$$

где $[b_{ik}]$ — матрица инцидентностей, т. е. $b_{ik} = 1$, если вершина x_i инцидентна ребру a_k , и $b_{ik} = 0$ в противном случае. Условие (52) отражает тот факт, что не более чем одна из двух концевых вершин любого ребра может быть окрашена в цвет j .

Хотя это условие более естественное, чем (49), но для его описания требуются mq ограничений, тогда как для условия (49) нужно только nq ограничений. Поскольку число ребер (m) связного графа обычно значительно больше числа его вершин (n), то условие (49) с вычислительной точки зрения предпочтительнее. Насколько велик при этом выигрыш, можно увидеть на следующем примере: если в некотором 100-вершинном графе число ребер будет составлять только 20% от числа ребер в полном 100-вершинном графе, то для задания условия (52) потребуется 1000 ограничений на каждый цвет, а для описания условия (49) — только 100 ограничений на каждый цвет.

2.5.3. Сведение задачи о раскраске к ЗНП

Поскольку при любой допустимой раскраске графа G множество вершин, окрашиваемых в один и тот же цвет, должно быть независимым множеством, то всякую допустимую раскраску можно интерпретировать как *разбиение* всех вершин графа G на такие независимые множества. Далее, если каждое независимое множество расширить до максимального (путем добавления к нему других вершин), то раскраска графа G может быть тогда истолкована как *покрытие* вершин графа G максимальными независимыми множествами. Очевидно, что в последнем случае некоторые вершины графа G могут принадлежать более чем одному максимальному независимому множеству. Это говорит о возможности существования различных допустимых раскрасок (использующих одно и то же число цветов), так как вершину, принадлежащую разным максимальным множествам, можно окрасить в любой из цветов, соответствующих тем максимальным независимым множествам, которым она принадлежит.

Итак, пусть построены все максимальные независимые множества графа G , (пусть таких множеств t), и пусть задана $(n \times t)$ -матрица

$M = [m_{ij}]$, у которой $m_{ij} = 1$, если вершина x_i принадлежит j -му максимальному независимому множеству, и $m_{ij} = 0$ в противном случае. Если теперь каждому максимальному независимому множеству сопоставить единичную стоимость, то задача раскраски сведется просто к задаче нахождения наименьшего числа столбцов в матрице M , покрывающих все ее строки (Напоминаем, что i -й столбец в 0-1-матрице покрывает те и только те строки, в которых на пересечении с i -м столбцом стоят единицы). Каждый столбец из решения этой ЗНП соответствует определенному цвету, который может быть использован для окраски всех вершин максимального независимого множества, представленного этим столбцом.

Хотя t — число столбцов матрицы \mathbf{M} — может быть весьма велико даже для графов средних размеров, формулировка задачи раскраски как ЗНП предпочтительнее, чем непосредственная ее постановка как задачи общего 0-1-программирования, потому что ЗНП могут быть решены для таких размеров (т. е. для числа переменных), которые по крайней мере на порядок больше, чем «подходящие» размеры в случае задач общего 0-1-программирования.

1. Пример. Для графа, изображенного на рис. 16, число всех максимальных независимых множеств равно 15, и в матрице, приведенной ниже, они представлены столбцами; на пустых местах в матрице должны стоять нули.

Максимальные независимые множества

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
x_1	1	1	1	1											
x_2					1	1	1	1							
x_3	1	1							1	1	1	1			
x_4			1	1	1								1		
x_5	1			1		1	1	1							
x_6				1		1	1	1	1	1	1	1	1	1	1
x_7							1	1	1	1	1	1	1	1	1
x_8			1											1	
x_9	1						1	1	1	1	1	1	1	1	1
x_{10}					1										
		↑	↑	↑				↑							

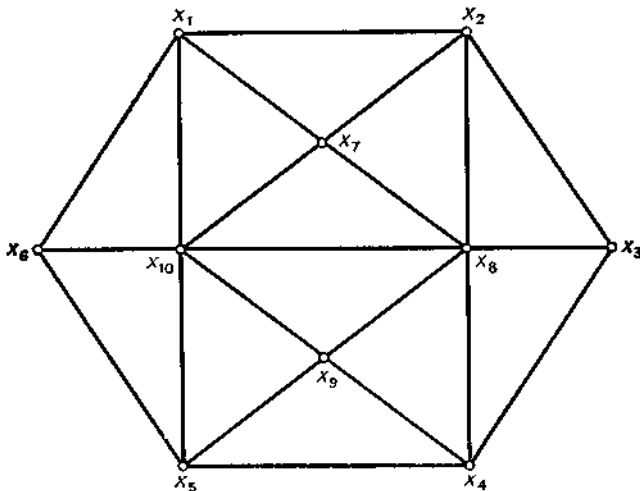


Рис. 16. Граф из примера 1.

Решение задачи о покрытии для этой матрицы состоит из 4 столбцов и не является единственным. Например, решениями будут множества столбцов $\{4, 6, 10, 14\}$, $\{4, 6, 10, 12\}$, $\{4, 6, 10, 11\}$, $\{4, 6, 10, 8\}$ и $\{4, 6, 10, 2\}$. Если мы выберем последнее из этих решений (отмеченное стрелками) и сопоставим цвета a, b, c, d столбцам 4, 6, 10 и 2 соответственно, то вершину x_1 (принадлежащую одновременно столбцам 2 и 4) можно окрасить в цвет a или в цвет d , а вершину x_3 (принадлежащую обоим столбцам 2 и 10) можно окрасить в цвет c или в цвет d и таким образом получить различные оптимальные раскраски. Интересно отметить, что в данном примере нижняя оценка для $\gamma[G]$ равна 4 (это можно получить из соотношения (4.2), так как $\alpha[G] = 3$ и $n = 10$) и, значит, совпадает с истинным значением $\gamma[G]$.

2.5.4. Алгоритм прямого неявного перебора, использующий дерево поиска

Для определения хроматического числа графа может быть использован с большой эффективностью простой метод неявного перебора. Метод состоит в следующем.

Предположим, что множество вершин как-то упорядочено и x_i — i -я вершина этого множества. Тогда первоначальная допустимая раскраска может быть получена так:

(i) окрасить x_i в цвет 1.

(ii) каждую из оставшихся вершин окрашивать последовательно (в соответствии с заданным упорядочением): вершина x_i окрашивается в цвет с наименьшим возможным «номером» (т. е. выбираемый цвет должен быть первым в данном упорядочении цветом, не использованным при окраске какой-либо вершины, смежной с x_i).

Пусть q — число цветов, требуемое для вышеупомянутой раскраски. Если существует раскраска, использующая только $q - 1$ цветов, то все вершины, окрашенные в цвет q , должны быть перекрашены в цвет $j < q$. Пусть x_{i^*} — первая вершина при заданном упорядочении, которая была окрашена в цвет q . Поскольку (согласно (ii)) она была так окрашена потому, что не могла быть окрашена в цвет $j < q$, то ее можно перекрасить в цвет $j < q$, лишь перекрасив предварительно хотя бы одну из смежных с ней вершин. Итак, шаг возвращения из вершины x_{i^*} можно осуществить следующим образом.

Из смежных с x_{i^*} вершин в множестве $\{x_1, \dots, x_{i^*-1}\}$ найти последнюю (при заданном упорядочении), т. е. вершину с наибольшим индексом. Пусть это будет вершина x_k . Если x_k окрашена в цвет j_k , то x_k перекрашивается в другой допустимый цвет с наименьшим возможным «номером» j'_k , таким, что $j'_k \geq j_k + 1$.

Если $j'_k < q$, то надо продолжать последовательно перекрашивать все вершины с x_{k+1} до x_{i^*} , применяя указанное выше правило (ii) и помня о том, что цвет q использовать нельзя. Если такая процедура осуществима, то будет найдена новая лучшая раскраска, использующая меньше, чем q цветов. В противном случае, т. е. если встретится вершина, «требующая» цвет q , то можно снова сделать шаг возвращения — из такой вершины.

Если $j'_k = q$ или нет другого допустимого цвета j'_k (см. ниже замечание (а)), то можно сразу же делать шаг возвращения из вершины x_k . Алгоритм заканчивает работу, когда на шаге возвращения достигается вершина x_1 .

Следующие замечания могут помочь ускорить вышеприведенную процедуру прямого неявного перебора.

(а) При любом упорядочении вершин допустимые цвета j для вершины x_i удовлетворяют условию $j \leq i$ (если $i < q$). Это очевидно, так как вершине x_i предшествуют (при данном упорядочении) только $i - 1$ вершин и, следовательно, никакой цвет $j > i$ не использовался. Таким образом, для вершины x_1 допустимым является только цвет 1, для x_2 — цвет 1 и цвет 2 (если x_2 смежна с x_1 то для x_2 допустим только цвет 2) и т. д.

(б) Из замечания (а) следует, что с вычислительной точки зрения выгодно располагать переменные в таком порядке, чтобы, например,

первые ρ вершин образовывали наибольшую клику графа G . Это приведет к тому, что каждая вершина x_i ($1 \leq i \leq \rho$) будет иметь только один допустимый цвет, т. е. цвет i , и алгоритм может закончить работу раньше, когда на шаге возвращения будет достигнута вершина x_0 .

Хотя процедура неявного перебора, описанная в этом разделе, является примитивным древовидным поиском (в котором не нужно вычислять никакие оценки для ограничения ветвлений), тем не менее она несколько не хуже других известных методов раскраски графов.

2.6. Приближенные алгоритмы раскрашивания

Существует много эвристических процедур раскрашивания графов, позволяющих находить хорошие приближения для хроматического числа графа в тех случаях, когда размеры графа слишком велики и получение оптимальной раскраски точными методами, упоминавшимися ранее, затруднительно. В настоящем разделе дается краткое описание одной из таких процедур и ряда ее разновидностей.

2.6.1. Последовательные методы, основанные на упорядочивании множества вершин

В этом простейшем из методов вершины вначале располагаются в порядке невозрастания их степеней.

Первая вершина окрашивается в цвет 1; затем список вершин просматривается сверху вниз (по невозрастанию степеней) и в цвет 1 окрашивается всякая вершина, которая не смежна с другой, уже окрашенной в этот цвет. Потом возвращаемся к первой в списке неокрашенной вершине, окрашиваем ее в цвет 2 и снова просматриваем список вершин сверху вниз, окрашивая в цвет 2 любую неокрашенную вершину, которая не соединена ребром с другой, уже окрашенной в цвет 2 вершиной. Аналогично действуем с цветами 3, 4 и т. д., пока не будут окрашены все вершины. Число использованных цветов будет тогда приближенным значением хроматического числа графа.

Простая модификация описанной выше эвристической процедуры состоит в переупорядочивании неокрашенных вершин после окраски каждой очередной вершины: оставшиеся неокрашенные вершины записываются в порядке невозрастания их «относительных» степеней, т. е. степеней в таком графе, который получается из данного после удаления окрашенных вершин (вместе с ребрами, инцидентными удаленным вершинам).

В этой процедуре молчаливо предполагалось, что если две вершины имеют одинаковые степени, то их взаимное положение в списке случайно. В таких ситуациях уточнение в размещении вершин можно осуществлять с помощью двухшаговых степеней $d_i^{(2)}$ вершин x_i , имеющих одинаковые степени (одинаковые 1-шаговые степени), где $d_i^{(2)}$ определяется как число маршрутов длины 2, исходящих из x_i . Эти вершины могут быть размещены тогда в соответствии с величинами степеней $d_i^{(2)}$. Если все-таки найдутся вершины, у которых совпадают и степени d_i и степени $d_i^{(2)}$, то можно вычислить трехшаговые степени $d_i^{(3)}$ (определяемые аналогичным образом) и разместить вершины с учетом степеней $d_i^{(3)}$ и т. д.

Если A — матрица смежности графа G с диагональными элементами, равными 0, то $d_i = \sum_{j=1}^n a_{ij}$. Двухшаговая степень определяется тогда по

формуле

$$d_i^{(2)} = \sum_{j=1}^n a_{ij}d_j,$$

и вообще k -шаговая степень определяется по рекуррентной формуле:

$$d_i^{(k)} = \sum_{j=1}^n a_{ij}d_j^{(k-1)}$$

Можно действовать иначе: размещать вершины сразу в соответствии с их степенями $d_i^{(2)}$ или степенями $d_i^{(3)}$ и применять тот же самый последовательный метод раскраски. Таким образом, описанный выше метод раскрашивания очерчивает целый класс последовательных методов, каждый из которых связан с определенным способом упорядочивания вершин, либо статическим, т. е. фиксированным сразу для всей процедуры, либо динамическим, т. е. изменяющимся в процессе раскраски. Способ упорядочивания может базироваться на многих возможных критериях, зависящих от степеней вершин или от каких-либо других родственных характеристик.

2.7. Обобщения и приложения

Задача раскраски в том «чистом» виде, в каком она рассматривалась выше в настоящей главе, редко встречается на практике. Однако ее обобщения и разновидности (незначительно отличающиеся от нее) находят широкое применение в большом числе различных прикладных задач. Целью данного раздела является ознакомление читателя с несколькими наиболее часто встречающимися обобщениями. Список приложений, естественно, этими примерами не ограничивается.

2.7.1. Простая задача размещения (загрузки)

Рассмотрим задачу размещения (загрузки) n каких-то предметов по ящикам. Пусть каждый предмет соответствует определенной вершине графа G . Всякий раз, когда два предмета x_i и x_j не могут быть размещены в одном ящике (например, когда предмет x_i может загрязнить предмет x_j), в граф G вводится ребро (x_i, x_j) . Если ящики имеют неограниченную вместимость, так что в каждый из них можно поместить сколько угодно предметов, то задача нахождения наименьшего числа ящиков для размещения предметов эквивалентна задаче нахождения хроматического числа графа G ; причем каждому ящику соответствует определенный «цвет», а предметы, окрашенные в один цвет, укладываются в один и тот же ящик.

(i) Ящики одинаковой вместимости. В действительности ящики обладают ограниченной вместимостью. Предположим, что вместимость одинакова у всех ящиков и равна Q . Это можно интерпретировать так: в один цвет окрашиваются не более чем Q вершин. В терминах 0-1-программирования данная задача загрузки может быть сформулирована следующим образом:

минимизировать выражение (51)

при ограничениях (48), (49) и

$$\sum_{i=1}^n \xi_{ij} \leq Q \text{ (для всех } j = 1, 2, \dots, q). \quad (53)$$

Решением этой задачи является матрица $[\xi_{ij}^*]$, описывающая оптимальное *размещение* (распределение) *предметов по ящикам* (иначе говоря, *группировка предметов по цвету*). Число q есть верхняя оценка для числа ящиков.

(ii) Ящики имеют, вообще говоря, различные вместимости, и ящикам приписаны стоимости. Пусть j -й ящик имеет вместимость Q_j и стоимость v_j . Предположим еще, что предметам (вершинам графа) сопоставлены веса, скажем, i -му предмету соответствует вес w_i .

Требуется так разместить предметы по ящикам, чтобы

- (а) выполнялись условия, «накладываемые» графом G ,
- (б) удовлетворялись ограничения на вместимость ящиков,
- (в) была наименьшей общая стоимость используемых ящиков.

Тогда задача примет следующий вид:

$$\text{минимизировать функцию } z = \sum_{j=1}^q \psi_j v_j \quad (54)$$

при ограничениях (48), (49) и

$$\left. \begin{aligned} \sum_{i=1}^n \xi_{ij} w_i &\leq Q_j, \\ \sum_{i=1}^n \xi_{ij} &\leq L \psi_j \end{aligned} \right\} \text{ для всех } j = 1, \dots, q, \quad (55, 56)$$

где переменная ψ_j принимает значение 1, если в ящик j помещен какой-либо предмет, и равна 0 в противном случае. L — произвольное положительное целое число, большее, чем n , а q — общее число имеющихся ящиков. Ограничение (55) означает, что ни один из ящиков не перегружается, а ограничение (56) гарантирует выполнимость условия: если $\psi_{j_0} = 0$ (т. е. ящик j_0 пуст), то все ξ_{ij_0} ($i = 1, \dots, n$) также равны 0.

В этом самом общем случае необходимо было ввести дополнительные переменные ψ_j , поскольку ящики (цвета) нельзя штрафовать тем способом, который характеризуется выражением (51). Следует отметить, что чем более общей является задача, тем менее важным становится ее «раскрасочный» аспект. Так, например, в только что рассмотренной задаче можно выделить две подзадачи:

(а) о «раскраске», соответствующую ограничению (49), и
 (б) о «ранце», определяемую ограничением (55). Два других ограничения, (48) и (56), являются ограничениями структурного характера.

Из-за этих двух взаимосвязанных аспектов общая задача раскраски значительно труднее для решения, чем «чистая» задача раскраски.

Две следующие прикладные задачи можно рассматривать как задачи о раскраске графа (или соответствующие обобщения задачи о раскраске).

2.7.2. Составление-графиков осмотра (проверки)

В задачах теории расписаний осмотры представляются в виде временных интервалов. Каждому осмотру можно сопоставить вершину некоторого графа, причем две любые вершины графа будут соединены ребром лишь тогда, когда соответствующие им осмотры нельзя осуществлять одновременно. Требуется составить такой график осмотра, который связан с наименьшими временными затратами (с учетом приведенных выше ограничений на «совместимость» осмотров). Эта задача эквивалентна задаче о раскраске вершин графа с использованием наименьшего числа цветов. Хроматическое число графа как раз и соответствует осмотру, требующему наименьших временных затрат.

Если число осмотров, которые можно осуществлять в одно и то же время, ограничено (например, из-за размеров помещения), то приходим к задаче типа (i) из разд. 2.7.1, и Q в этом случае является числом помещений, где происходит осмотр.

2.7.3. Распределение ресурсов

Пусть для выполнения каких-то n работ надо распределить m имеющихся в наличии ресурсов. Считаем, что каждая из работ выполняется за некоторый (одинаковый для всех работ) промежуток времени и что для выполнения i -й работы требуется подмножество ресурсов S_i . Построим граф G : каждой работе соответствует определенная вершина графа, а ребро (x_i, x_j) существует в графе тогда и только тогда, когда для выполнения i -й и j -й работ требуется хотя бы один общий ресурс, т. е. когда $S_i \cap S_j \neq \emptyset$. Это означает, что i -я и j -я работы не могут выполняться одновременно. Раскраска графа G определяет тогда некоторое распределение ресурсов (по выполняемым работам), причем такое, что работы, соответствующие вершинам одного цвета, выполняются одновременно. Наилучшее использование ресурсов (т. е. выполнение всех n работ за наименьшее время) достигается при оптимальной раскраске вершин графа G .

2.8. Перечисление раскрашенных и других графов

2.8.1. Число графов

Мы займемся перечислением графов, корневых графов, связных графов, 2-раскрашенных графов, локально ограниченных графов, раскрашенных графов, булевых функций и эйлеровых графов.

Эффективный метод Пойа для перечисления графов требует построения группы подстановок, орбиты которой соответствуют в точности изоморфным классам помеченных графов с p вершинами и q ребрами. После получения явной формулы для соответствующего циклового индекса, применив теорему Пойа с рядом $1 + x$ в качестве перечисляющего ряда для фигур, находим перечисляющий многочлен, в котором коэффициент при x^q равен числу (p, q) -графов.

Более точно, пусть

$$g_p(x) = \sum_{q=0}^m g_{p,q} x^q, \quad (57)$$

где $m = \binom{p}{2}$ и $g_{p,q}$ — число (p, q) -графов. Например, если взглянуть на

рис. 17, можно убедиться, что

$$g_4(x) = 1 + x + 2x^2 + 3x^3 + 2x^4 + x^5 + x^6. \quad (58)$$

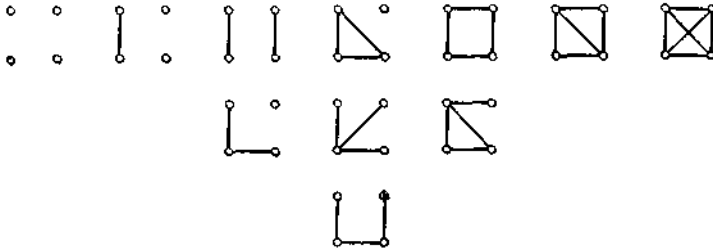


Рис. 17. 11 графов четвертого порядка.

Заметим, что коэффициенты в этих многочленах всегда «симметричны относительно концов», ибо число (p, q) -графов равно числу

$(p, m - q)$ -графов, дополнительных к первоначальным.

Так как теорема Пойа перечисляет орбиты функций, то мы сначала установим естественное соответствие между графами и функциями. Пусть $X = \{1, \dots, p\}$, в то время как $X^{(2)}$ обозначает все 2-подмножества множества X . Тогда, если $Y = \{0, 1\}$, то функции из $X^{(2)}$ в Y представляют помеченные графы порядка p . Каждая функция f соответствует тому графу $G(f)$, у которого множество вершин есть X и вершины i и j смежны тогда и только тогда, когда $f\{i, j\} = 1$. Таким образом, две функции f и h представляют один и тот же граф, если существует некоторая подстановка α на множестве X , такая, что всякий раз, когда i и j — смежные вершины в $G(j)$, вершины αi и αj смежны в графе $G(h)$. Следовательно, графы $G(f)$ и $G(h)$ изоморфны тогда и только тогда, когда для некоторой подстановки α , действующей на множестве X ,

$$f\{i, j\} = h\{\alpha i, \alpha j\} \quad (59)$$

для всех пар $\{i, j\}$ из множества $X^{(2)}$. Это уравнение подсказывает следующую унарную операцию на группах подстановок, которая приводит к группе, требуемой для перечисления графов. Пусть A — группа подстановок с множеством объектов $X = \{1, 2, \dots, p\}$. *Парная группа группы A* , обозначаемая $A^{(2)}$, есть группа подстановок, индуцируемая группой A и действующая на множестве $X^{(2)}$. Точнее, каждая подстановка α группы A индуцирует подстановку α' группы $A^{(2)}$, такую, что для каждого элемента $\{i, j\}$ множества $X^{(2)}$

$$\alpha^i \{i, j\} = \{\alpha i, \alpha j\}. \quad (60)$$

Таким образом, степень группы $A^{(2)}$ равна

$$\binom{p}{2} \text{ и } A \cong A^{(2)},$$

если A отлична от S_2 . Чтобы пояснить это определение, введем реберную группу графа — понятие, которое будет также достаточно полезным и в дальнейшем. Пусть G — граф с множеством вершин $V(G)$ и множеством ребер $X(G)$. Каждая подстановка α группы $\Gamma(G)$ индуцирует подстановку α' , действующую на множество $X(G)$ следующим образом. Если u и v — смежные в G вершины, так что $\{u, v\}$ — ребро графа G , то

$$\alpha' \{u, v\} = \{\alpha u, \alpha v\}. \quad (61)$$

Эта совокупность подстановок на множестве $X(G)$ образует группу, обозначаемую $\Gamma_1(G)$ и называемую *реберной группой графа G* . Например, реберная группа полного графа порядка p и парная группа симметрической группы степени p идентичны, т. е.

$$\Gamma_1(K_p) = S_p^{(2)}. \quad (62)$$

Теперь, учитывая формулы (59) и (60), заключаем, что две функции f и h представляют один и тот же граф тогда и только тогда, когда существует подстановка α на множестве X , такая, что для всех z из $X^{(2)}$

$$f(z) = h(\alpha' z). \quad (63)$$

Но это условие совпадает в точности с требованием, чтобы f и h содержались в одной и той же орбите степенной группы $E_2^{S_p^{(2)}}$. Следовательно, орбиты этой степенной группы соответствуют различным непомеченным графам порядка p , т. е. изоморфным классам помеченных графов. Применение теоремы Пойа к перечислению этих орбит в соответствии с их весами приводит к следующей разновидности формулы Пойа для перечисления графов.

Теорема. Многочлен $g_p(x)$, перечисляющий графы порядка p в соответствии с числом ребер в них, дается соотношением

$$g_p(x) = Z(S_p^{(2)}, 1 + x), \quad (64)$$

где

$$Z(S_p^{(2)}) = \frac{1}{p!} \sum_{\{j\}} \frac{p!}{k^j j! k!} \prod_k S_{2k+1}^{k^j / 2k+1} \prod_k (S_k S_{2k}^{k-1})^{j/2k} S_k^k \binom{ik}{2} \prod_{r < t} S_{[r, t]}^{(r, t) / r! t!}. \quad (65)$$

Доказательство. Определим весовую функцию w на множестве $Y = \{0, 1\}$, положив $w(0) = 0$ и $w(1) = 1$. Тогда вес функции, действующей из $X^{(2)}$ в Y , и вес орбиты, которой она принадлежит,



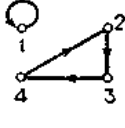
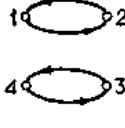
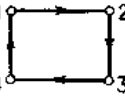
определяются обычным образом. Значит, вес орбиты, задаваемой группой $E_2^{S_p^{(2)}}$, есть просто число ребер в соответствующем графе. Тогда формула (64) следует из теоремы Пойа.

Теперь получим формулу (65) для $Z(S_p^{(2)})$, проиллюстрировав ее вначале для случая $p = 4$. В цикловом индексе

$$Z(S_4) = \frac{1}{24} (s_1^4 + 6s_1^2s_2 + 8s_1s_3 + 3s_2^2 + 6s_4) \quad (66)$$

каждое слагаемое может быть представлено одной из отвечающих ему подстановок, как это показано в табл. 1.

Таблица 1

Слагаемое из $Z(S_4)$	Подстановка из S_4	Диаграмма этой подстановки
s_1^4	(1) (2) (3) (4)	
$s_1^2 s_2$	(1) (2) (34)	
$s_1 s_3$	(1) (234)	
s_2^2	(12) (34)	
s_4	(1234)	

Каждый из пяти типов подстановок, принадлежащих группе S_4 , индуцирует некоторую подстановку из парной группы $S_4^{(2)}$. Таблица 2 содержит все слагаемые циклового индекса $Z(S_4^{(2)})$, индуцируемые соответствующими слагаемыми из $Z(S_4)$.

Слагаемое из $Z(S_4)$	Индукцируемое слагаемое из $Z(S_4^{(2)})$
s_1^4	s_1^6
$s_1^2 s_2$	$s_1^2 s_2^2$
$s_1 s_3$	s_3^2
s_2^2	$s_1^2 s_2^2$
s_4	$s_2 s_4$

Одно из этих слагаемых иллюстрируется на рис. 18, где 2-подмножества обозначены ради краткости следующим образом: $\{1, 2\} = 12$ и т. д.



Рис. 18. Подстановка (1) (2) (34) и индуцируемая ею подстановка в $S_4^{(2)}$

Приводя подобные члены, находим, что

$$Z(S_4^{(2)}) = \frac{1}{24} (s_1^6 + 9s_1^2 s_2^2 + 8s_3^2 + 6s_2 s_4). \quad (67)$$

Подставляя $1 + x$ в этот цикловой индекс, имеем, как и в формуле (58),

$$Z(S_4^{(2)}, 1 + x) = g_4(x) = 1 + x + 2x^2 + 3x^3 + 2x^4 + x^5 + x^6. \quad (68)$$

Мы указываем соответствие между слагаемыми, входящими в цикловые индексы групп S_p и $S_p^{(2)}$, записывая

$$s_1^4 \rightarrow s_1^4, \quad s_1^2 s_2 \rightarrow s_1^2 s_2^2$$

и т. д. Чтобы получить выражение для $Z(S_p^{(2)})$, мы должны найти отсутствующий член в соотношении

$$s_1^i s_2^j \dots s_p^p \rightarrow ? \quad (69)$$

Пусть α — подстановка из S_p , которой в цикловом индексе $Z(S_p)$ соответствует слагаемое $\prod S_k^{j_k}$. Существуют два различных вклада, которые делаются подстановкой α' в соответствующее слагаемое циклового индекса $Z(S_p^{(2)})$. Первый вклад определяется парами таких элементов множества $X = \{1, 2, \dots, p\}$, которые принадлежат общему циклу подстановки α ; второй — парами тех элементов множества X , которые содержатся в разных циклах подстановки α . Сейчас мы определим первый из этих вкладов. Пусть $z_k = (12 \dots k)$ — цикл длины k подстановки α . На рис. 19 показана подстановка из парной группы, индуцируемая циклом z_k при $k = 2, 3, \dots, 6$.

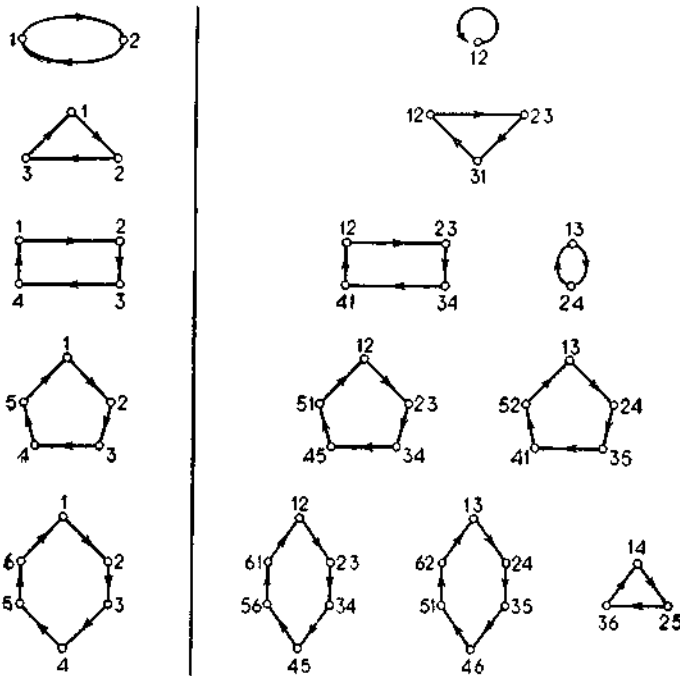


Рис. 19. Циклы в S_p и индуцируемые ими подстановки в $S_p^{(2)}$

Очевидно, что если k — нечетно, то цикл z_k индуцирует $(k - 1)/2$ циклов одной и той же длины, т. е.

$$s_k \rightarrow s_k^{(k-1)/2}.$$

С другой стороны, когда k — четное, то находим

$$s_k \rightarrow s_{k/2} s_k^{(k-2)/2}.$$

Таким образом, так как существуют j_k циклов длины k в подстановке α , то пары элементов, лежащих в общих циклах, порождают соответствие

$$s_k^{j_k} \rightarrow s_k^{j_k \cdot (k-1)/2} \tag{70}$$

для нечетных k и

$$s_k^{j_k} \rightarrow (s_{k/2} s_k^{(k-2)/2})^{j_k} \tag{71}$$

для четных k .

Чтобы вычислить второй вклад, рассмотрим два цикла z_r и z_t в подстановке α . Пусть, как обычно, $[r, t]$ и (r, t) обозначают соответственно наименьшее общее кратное и наибольший общий делитель чисел r и t . Тогда циклы z_r и z_t индуцируют на парах таких элементов, один из которых принадлежит циклу z_r , а другой — циклу z_t , ровно (r, t) циклов длины $[r, t]$. В частности, когда $r = t = k$, то соответствующий «взнос» состоит из k циклов длины k . Итак, когда $r \neq t$, то имеем

$$s_r^{j_r} s_t^{j_t} \rightarrow s_{[r, t]}^{(r, t) j_r j_t}, \tag{72}$$

а если $r=t = k$, то

$$s_k^{j_k} \rightarrow s_k^{\binom{j_k}{2}}. \tag{73}$$

Теперь, перемножая правые части соотношений (70) — (73) по всем подходящим случаям, получаем отсутствующий член в правой части формулы (69). Таким образом, справедливость формулы (65) установлена.

В *мультиграфе* две вершины могут быть соединены более чем одним ребром (см. рис. 20).

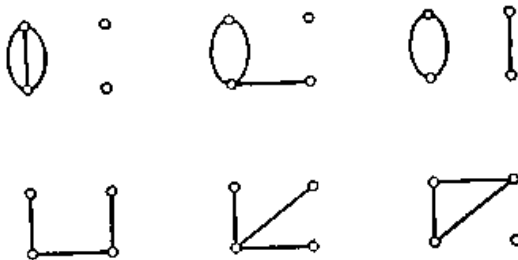


Рис. 20. Шесть мультиграфов четвертого порядка с тремя ребрами.

Пусть $m_p(x)$ — производящая функция для мультиграфов, так что коэффициент при x^q есть число мультиграфов с p вершинами и q ребрами. Следующая теорема Харари получена видоизменением

перечисляющего ряда для фигур, который использовался при перечислении графов.

Теорема. Производящая функция $t_p(x)$ для мультиграфов порядка p дается формулой

$$t_p(x) = Z(S_p^{(2)}, 1/(1-x)). \quad (74)$$

Для $p = 4$ находим

$$t_4(x) = 1 + x + 3x^2 + 6x^3 + 11x^4 + 18x^5 + 32x^6 + 48x^7 + 75x^8 + 111x^9 + 160x^{10} + \dots \quad (75)$$

Правильность коэффициента при x^3 подтверждается рис. 20. Следуя Партасарати, Харари и Палмеру, мы можем описать перечисление локально ограниченных графов. Это по существу сводится к дальнейшему уточнению циклового индекса парной группы $S_p^{(2)}$.

Разбиение графа представляет собой последовательность степеней его вершин, обычно записываемую в порядке невозрастания. *Локально ограниченный* граф — это граф с данным разбиением. Производящая функция, перечисляющая локально ограниченные графы с p вершинами, является многочленом $N(x_1, x_2, \dots, x_p)$, таким, что каждое слагаемое $x_1^{e_1} x_2^{e_2} \dots x_p^{e_p}$ удовлетворяет условию

$$e_1 \geq e_2 \geq \dots \geq e_p.$$

Коэффициент при таком слагаемом есть число графов с разбиением e_1, e_2, \dots, e_p . К примеру, слагаемое в $N(x_1, \dots, x_5)$, отвечающее графу, изображенному на рис. 21, имеет вид;

$x_1^4 x_2^3 x_3^3 x_4^2 x_5^2$, ибо последовательность степеней, соответствующая этому графу, такова: 4, 3, 3, 2, 2.

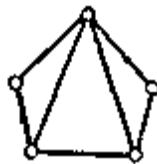


Рис. 21. Граф со степенной последовательностью (4, 3, 3, 2, 2).

Чтобы получить формулу для производящей функции, воспользуемся естественным представлением, даваемым подходящей степенной группой. Полагая, как раньше, $Y = \{0, 1\}$, рассмотрим функцию W , которая отображает множество $Y^{X^{(2)}}$ в кольцо многочленов, зависящих от переменных x_i , и определяется формулой

$$W(f) = \prod_{(i, j) \in X^{(2)}} (x_i x_j)^{f(i, j)}, \quad (76)$$

Тогда в графе, соответствующем функции f , степень вершины i дается показателем степени у переменной x_i в выражении для $W(f)$. Введем для удобства линейный оператор θ , действующий на $W(f)$ в соответствии с правилом: θ переупорядочивает показатели степеней так, чтобы они шли в порядке невозрастания, в то время как сами переменные располагаются в порядке возрастания индексов (слева направо). Например,

$$\theta(x_3^4 x_1^2 x_2^3 x_4) = x_1^4 x_2^3 x_3^2 x_4.$$

Применяя «взвешенный» вариант леммы Бернсайда, приходим к следующему результату, который был получен Партасарати в несколько иной форме.

Теорема. *Производящая функция, перечисляющая локально ограниченные графы, имеет вид*

$$N(x_1, x_2, \dots, x_p) = \frac{1}{p!} \sum_{\gamma \in \mathbb{E}_2^{S_p}} \theta \left(\sum_{f=\gamma f} W(f) \right), \quad (77)$$

где

$$\begin{aligned} \sum_{f=\gamma f} W(f) &= \prod_{z_r z_s} \left(1 + \prod_{i \in z_r} x_i^{s/(r, s)} \prod_{i \in z_s} x_i^{r/(r, s)} \right)^{(r, s)} \times \\ &\times \prod_{\substack{z_r \\ r\text{-четное}}} \left(1 + \prod_{i \in z_r} x_i \right) \left(1 + \prod_{i \in z_r} x_i^2 \right)^{(r-2)/2} \times \\ &\times \prod_{\substack{z_r \\ r\text{-нечетное}}} \left(1 + \prod_{i \in z_r} x_i^2 \right)^{(r-1)/2}; \end{aligned} \quad (78)$$

здесь первое произведение берется по всем различным парам циклов z_r и z_s в подстановке α , а два других произведения — по всем циклам такой подстановки из группы S_p , которая соответствует подстановке γ . Далее, применяя эту теорему, имеем

$$\begin{aligned} N(x_1, x_2, x_3) &= (1/3!) \{ \theta((1 + x_1 x_2)(1 + x_2 x_3)(1 + x_1 x_3)) + \\ &+ 3\theta((1 + x_1 x_2 x_3^2)(1 + x_1 x_2)) + 2\theta(1 + x_1^2 x_2^2 x_3^2) \} = \\ &= (1/3!) \{ (1 + 3x_1 x_2 + 3x_1^2 x_2 x_3 + x_1^2 x_2^2 x_3^2) + \\ &+ 3(1 + x_1 x_2 + x_1^2 x_2 x_3 + x_1^2 x_2^2 x_3^2) + 2(1 + x_1^2 x_2^2 x_3^2) \} = \\ &= 1 + x_1 x_2 + x_1^2 x_2 x_3 + x_1^2 x_2^2 x_3^2. \end{aligned}$$

Указанный метод дает только формальное решение и не подходит для получения точных значений или порядков искомым величин.

2.8.2. Число связных графов

Как и в случае помеченных графов, мы убедимся в том, что производящие функции для графов и связных графов тесно связаны некоторым соотношением, содержащим экспоненциальную функцию. Кроме того, связные графы с указанными свойствами часто могут быть перечислены аналогичным способом в терминах общего числа графов с теми же свойствами.

Пусть $g(x)$ — производящая функция для графов, так что

$$g(x) = \sum_{p=1}^{\infty} g_p x^p, \quad (79)$$

где g_p — число графов порядка p , и пусть $c(x)$ — соответствующая производящая функция для связных графов, т. е.

$$c(x) = \sum_{p=1}^{\infty} c_p x^p. \quad (80)$$

Теперь можно сформулировать теорему Риддела, связывающую эти два степенных ряда.

Теорема. *Производящие функции $g(x)$ и $c(x)$ для графов и связных графов удовлетворяют соотношению*

$$1 + g(x) = \exp \left\{ \sum_{k=1}^{\infty} c(x^k)/k \right\}. \quad (81)$$

Доказательство. Из теоремы Пойа вытекает, что ряд $Z(S_n, c(x))$ перечисляет графы, состоящие ровно из n компонент. Следовательно, суммируя этот ряд по всем n , получаем

$$1 + g(x) = Z(S_{\infty}, c(x)). \quad (82)$$

Формула (81) выводится из формулы (82).

Кэдоган разработал эффективный метод подсчета числа связных графов порядка p , неявно содержащийся в формуле (81). Сначала полагаем

$$\sum_{p=1}^{\infty} a_p x^p = \ln(1 + g(x)). \quad (83)$$

Тогда из формулы

$$a_m = A_m - \frac{1}{m} \sum_{k=1}^{m-1} k a_k A_{m-k}.$$

следует, что

$$pa_p = pg_p - \sum_{k=1}^{p-1} ka_k g_{p-k}. \quad (84)$$

В табл. 1 представлены значения для pa_p ($p \leq 9$), вычисленные с использованием уравнения (84) и значений величин g_p .

Таблица 1

Число связных графов

p	g_p	pa_p	c_p
1	1	1	1
2	2	3	1
3	4	7	2
4	11	27	6
5	34	106	21
6	156	681	112
7	1 044	5 972	853
8	12 346	88 963	11 117
9	274 668	2 349 727	261 080

Мы далее увидим, как выражаются значения коэффициентов c_p через величины a_p . Так как

$$\sum_{p=1}^{\infty} a_p x^p = \sum_{k=1}^{\infty} c(x^k)/k, \quad (85)$$

то, сравнивая коэффициенты при соответствующих степенях, имеем

$$pa_p = \sum_{d|p} dc_d. \quad (86)$$

Обращая это соотношение с использованием функции Мёбиуса $\mu(d)$, выражаем числа c_p через a_p :

$$c_p = \sum_{d|p} \frac{\mu(d)}{d} a_{p/d}. \quad (87)$$

Эта формула Кэдогана была использована для вычисления значений коэффициентов c_p ($p \leq 9$), помещенных в табл. 1. Всякий раз, когда две произвольные производящие функции удовлетворяют соотношению (81) из теоремы, их коэффициенты связаны уравнениями (84) и (87). Это часто случается, когда производящие функции для графов со специальным свойством и для связных графов с тем же свойством удовлетворяют соотношению (81). Следовательно, рассматриваемые связные графы можно перечислить, используя формулы (84) и (87). Например, если $w(x)$ — производящая функция для четных графов и функция $u(x)$ перечисляет связные четные графы, то

$$1 + w(x) = \exp \left\{ \sum_{k=1}^{\infty} u(x^k)/k \right\}. \quad (88)$$

Следовательно, коэффициенты функции $u(x)$ можно вычислить, используя коэффициенты функции $w(x)$ и соотношения, подобные тем, которые даются формулами (84) и (87). Связные четные графы — это как раз эйлеровы графы.

Кэдоган развил свой метод так, чтобы можно было в качестве параметра при перечислении включить не только число вершин, но и число ребер. Набросаем детали этого процесса. Пусть $g_{p,q}$ и $c_{p,q}$ — числа (p, q) -графов и связных (p, q) графов соответственно. Положим

$$g(x, y) = \sum g_p x^p y^q, \quad (89)$$

$$c(x, y) = \sum c_p x^p y^q. \quad (90)$$

Тогда из теоремы перечисления Пойа для случая двух переменных следует, что

$$1 + g(x, y) = \exp \left\{ \sum_{k=1}^{\infty} c(x^k, y^k)/k \right\}. \quad (91)$$

Пусть для всякого $p \geq 1$ многочлен $b_p(y)$ от переменной y определяется формулой

$$\sum_{p=1}^{\infty} b_p(y) x^p = \ln(1 + g(x, y)). \quad (92)$$

Можно показать (так же, как для случая соотношения (84)), что

$$p b_p(y) = p g_p(y) - \sum_{k=1}^{p-1} k b_k(y) g_{p-k}(y). \quad (93)$$

Коэффициенты многочлена $b_p(y)$ можно вычислить, используя коэффициенты многочлена $g_p(y)$ и выражение (93). Положим для удобства

$$b_p(y) = \sum b_{p,q} y^q. \quad (94)$$

Имеем также соотношение

$$\sum b_p(y) x^p = \sum_{k=1}^{\infty} c(x^k, y^k)/k. \quad (95)$$

Приравнивая коэффициенты при $x^p y^q$ и используя обращение Мёбиуса, получаем формулу

$$c_{p,q} = \sum_{r \mid (p,q)} \bar{b}_{p/r, q/r} \mu(r)/r. \quad (96)$$

Значения коэффициентов c_p приводимые в табл 2, вычислены Кэдоганом с помощью формулы (96)

Таблица 2

Число связных (p, q) -графов

$p \backslash q$	0	1	2	3	4	5	6	7	8	9	10	11	12	13
1	1													
2		1												
3			1	1										
4				2	2	1	1							
5					3	5	5	4	2	1	1			
6						6	13	19	22	20	14	9	5	2
7							11	33	67	107	132	138	126	95
8								23	89	236	486	814	1169	1454

Этот подход можно также использовать для нахождения числа связных (p, q) -графов с указанными свойствами..

2.8.3. Число k -раскрашенных графов

Раскрашенный граф состоит из графа G с множеством вершин V и такого отношения эквивалентности на множестве V , что любые смежные вершины не эквивалентны. k классов эквивалентности рассматриваются как различные цвета и граф G называется k -раскрашенным. Два k -раскрашенных графа изоморфны, если существует взаимно однозначное соответствие между их множествами вершин, которое сохраняет не только смежность, но и цвета. Заметим, что цвета не закреплены постоянно, а являются взаимно-заменяемыми. Данный граф может быть k -раскрашен многими способами. Например, все 3-раскраски некоторого помеченного графа порядка 6 показаны на рис. 22, где буквы a, b и c обозначают цвета, а натуральные числа обозначают пометки.

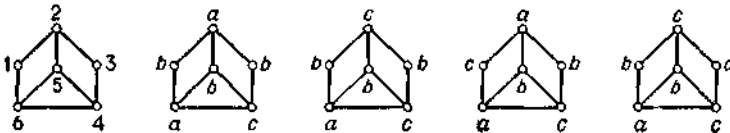


Рис. 22. Все 3-раскраски графа.

Следуя Риду, и обобщая результат Гилберта, найдем формулу для числа помеченных k -раскрашенных графов порядка p .

Пусть p_1, \dots, p_k — целые положительные числа, образующие упорядоченное разбиение числа p , так что

$$\sum_{i=1}^k p_i = p. \quad (97)$$

Если через $\{p\}$ обозначить произвольное решение уравнения (97), то теорему (и формулу) Рида можно записать в следующей форме.

Теорема. Число $C_p(k)$ k -раскрашенных помеченных графов порядка p равно

$$C_p(k) = \frac{1}{k!} \sum_{\{p\}} \binom{p}{p_1, \dots, p_k} 2^{(p^2 - \sum p_i^2)/2}. \quad (98)$$

Доказательство. Заметим, что число k -раскрашенных помеченных графов порядка p , в которых цвета фиксированы, равно $k!C_p(k)$. Поэтому рассмотрим k фиксированных цветов. Каждое решение $\{p\}$ уравнения (97) определяет упорядоченное разбиение числа p на k частей, и поэтому мы ищем число помеченных графов, у которых p_i вершин имеют i -й цвет. Число способов, которыми могут быть выбраны пометки для вершин, дается полиномиальным коэффициентом

$$\binom{p}{p_1, p_2, \dots, p_k}.$$

Очевидно, что существуют

$$\binom{p}{2} = \sum_{i=1}^k \binom{p_i}{2} \quad (99)$$

пар вершин, имеющих различные цвета. Так как каждая такая пара может быть либо смежной, либо нет, то, возводя 2 в степень, равную выражению (99), и используя равенство (97), мы получаем для общего числа графов с p_i вершинами цвета i в точности то выражение, которое стоит под знаком суммы в формуле (98).

Суммируя по всем решениям $\{p\}$ уравнения (87), получаем произведение $k!C_p(k)$. Тем самым формула (98) доказана.

Заметим, что коэффициент при x^q в многочлене

$$\frac{1}{k!} \sum_{\{p\}} \binom{p}{p_1, \dots, p_k} (1+x)^{(p^2 - \sum p_i^2)/2}$$

равен числу k -раскрашенных помеченных (p, q) -графов. Применяя, например, это утверждение в случае $p = 4, q = 5$ и $k = 3$, получим число шесть — число 3-раскрашенных помеченных $(4, 5)$ -графов. В

справедливости этого результата легко убедиться непосредственно, ибо существует лишь один немеченный (4, 5)-граф и он может быть помечен шестью способами, как это показано на рис. 23.

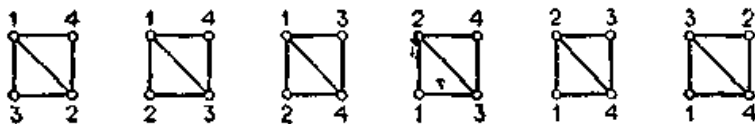


Рис. 23. Шесть различных распределений пометок в графе.

Рекурсивную формулу для $C_p(k)$ нетрудно получить как следствие:

$$C_p(k) = \frac{1}{k} \sum_{n=1}^{p-1} \binom{p}{n} 2^{n(p-n)} C_n(k-1). \quad (100)$$

Проверку формулы (100) можно осуществить, выразив производящую функцию для $C_p(k)$ через производящую функцию $C_p(k-1)$. Значения произведения $k!C_p(k)$ для $p \leq 7$ приводятся у Рида и были использованы нами при составлении табл. 1.

Таблица 1

$p \backslash k$	1	2	3	4	5	6	7
1	1	0	0	0	0	0	0
2	1	2	0	0	0	0	0
3	1	12	8	0	0	0	0
4	1	80	192	64	0	0	0
5	1	720	5 120	5 120	1 024	0	0
6	1	9152	192 000	450 560	24 576	32 768	0
7	1	165 312	10 938 368	56 197 120	64 225 280	22 020 096	2 097 152

2.8.4. Ациклические орграфы

Маршрут длины n в орграфе D определяется последовательностью своих вершин v_0, v_1, \dots, v_n в которой вершина v_i смежна к вершине v_{i+1} при $i < n$. В замкнутом маршруте начальная и конечная вершины совпадают. Контур представляет собой нетривиальный замкнутый маршрут, все вершины которого, исключая первую и последнюю, различны. Ациклический орграф не имеет контуров. Перечисление помеченных ациклических орграфов довольно легко осуществить, следуя работе Робинсона.

Орграф E называется *расширением* орграфа D , если D является ориентированным подграфом орграфа E , порожденным подмножеством тех вершин орграфа E , которые имеют положительную полустепень захода. Каждый ациклический орграф должен содержать по крайней мере одну вершину с нулевой полустепенью захода. Следовательно, всякий ациклический орграф, имеющий хотя бы одну дугу, является расширением единственного собственного ориентированного подграфа. Кроме того, каждый ациклический орграф имеет много расширений, но все они должны быть ациклическими орграфами. Предположим, что D — ациклический орграф, имеющий в точности $n \geq 1$ вершин u_i , полустепень захода которых равна нулю, и s остальных вершин v_i . Мы можем построить расширение E орграфа D , содержащее ровно k вершин с нулевой полустепенью захода: добавляем k новых вершин w_i и новые дуги, такие, что каждая из n вершин u_i является смежной *из* некоторой новой вершины w_i (однако каждая новая вершина w_i может быть смежной *к* любой вершине v орграфа D). На рис. 24 новые (добавленные) вершины обозначены через w_1, w_2 и w_3 ; каждая старая вершина u_1 и u_2 с нулевой полустепенью захода является смежной из некоторой вершины w_i .

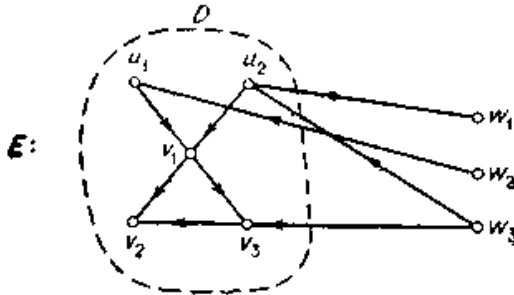


Рис. 24. Расширение ациклического орграфа,

Таким образом, все ациклические орграфы порядка p могут быть получены с помощью расширения ациклических орграфов, порядок которых меньше p . Детальнее, пусть a_p — число помеченных ациклических орграфов порядка p и $a_{p,k}$ — число таких же орграфов, у которых ровно k вершин имеют нулевую полустепень захода. Если $k=p$, то, конечно, $a_{p,p} = 1$, так как в этом случае подходит только вполне несвязный орграф. Очевидно, что для всех p

$$a_p = \sum_{k=1}^p a_{p,k} \cdot k. \tag{101}$$

Сейчас мы покажем, как может быть выражено $a_{p,k}$ через $a_{p-k,n}$ при

$n \leq p - k$. Сначала мы докажем, что всевозможные расширения всех $a_{p-k, n}$ орграфов с $p - k$ вершинами, из которых в точности n вершин имеют нулевую полустепень захода, вносят в значение $a_{p, k}$ число, равное

$$(2^k - 1)^n 2^{k(p-n-k)} \binom{p}{k} a_{p-k, n}. \quad (102)$$

Мы ищем число помеченных расширений E всех $a_{p-k, n}$ помеченных ациклических орграфов D . Для каждого из $\binom{p}{k}$ способов расставить пометки у k новых вершин w_i в расширении E существуют $a_{p-k, n}$ распределений пометок в орграфах D , подлежащих расширению. Это объясняет множитель $\binom{p}{k} a_{p-k, n}$ в выражении (102). Каждая из n вершин с нулевой полустепенью захода и орграфе D должна быть смежна по крайней мере из одной новой вершины w_i ($1 \leq i \leq k$). Следовательно, существуют $2^k - 1$ возможностей направить дуги в каждую из этих n вершин, что дает $(2^k - 1)^n$ для всех вершин. Каждая новая вершина может либо быть, либо не быть смежной ко всякой из $p - n - k$ вершин, имеющих в орграфе D положительную полустепень захода. Значит, существуют 2^{p-n-k} возможных комбинаций для каждой из новых вершин и, следовательно, $(2^{p-n-k})^k$ комбинаций для всех таких вершин. Перемножая все эти выражения, получаем (102). Суммируя (102) по всем n , имеем соотношение для $a_{p, k}$.

Теорема. Число $a_{p, k}$ помеченных ациклических орграфов порядка p , имеющих ровно k вершин с нулевой полустепенью захода, удовлетворяет соотношению

$$a_{p, k} = \sum_{n=1}^{p-k} (2^k - 1)^n 2^{k(p-n-k)} \binom{p}{k} a_{p-k, n}. \quad (103)$$

Таким образом, для нахождения a_p можно использовать формулы (101) и (103). Эти результаты можно также выразить в терминах производящих функций. Пусть в ряде $a(x, y)$ коэффициент при $x^k y^{p-k}$ равен числу помеченных ациклических орграфов с p вершинами, из которых в точности k вершин имеют нулевую полустепень захода. Тогда несколько первых членов ряда $a(x, y)$ даются выражением

$$\begin{aligned}
 a(x, y) = & x + x^2 + 2xy + x^3 + 9x^2y + 15xy^2 + x^4 + 28x^3y + \\
 & + 198x^2y^2 + 316xy^3 + x^5 + 75x^4y + 1610x^3y^2 + \\
 & + 10710x^2y^3 + 16885xy^4 + \dots
 \end{aligned}
 \tag{104}$$

Например, существуют шесть способов расставить пометки в ациклическом орграфе D_1 и три способа расставить пометки в ациклическом орграфе D_2 , изображенных на рис. 1.6.2.



Рис. 25. Два ациклических орграфа третьего порядка, у которых по две вершины имеют нулевую полу степень захода.

Всего — девять способов, что соответствует члену $9x^2y$ в ряде $a(x, y)$.

2.8.5. Деревья

Деревом называется связный граф, не имеющий циклов. Хорошо известно, что всякое нетривиальное дерево имеет не менее двух висячих вершин (вершин степени 1). Это следует из того, что если T — дерево с p вершинами и q ребрами, то

$$q = p - 1. \tag{105}$$

Все деревья с не более чем пятью вершинами показаны на рис. 26 (вместе с числом способов, которыми они могут быть помечены).

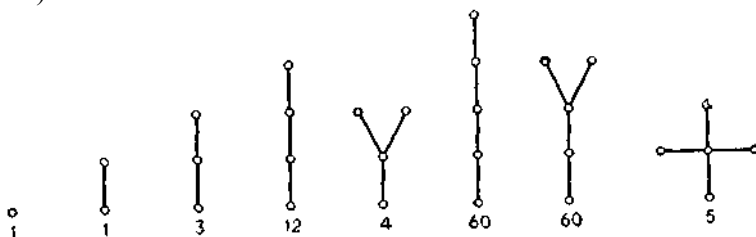


Рис. 26. Деревья, порядки которых не выше 5, и число способов распределения пометок у каждого из этих деревьев.

Отсюда вытекает, что число t_p помеченных деревьев с p вершинами имеет следующие наименьшие значения: 1, 1, 3, 16, 125. Многие авторы справедливо предполагали, исходя из этой последовательности значений, что формула для подсчета помеченных деревьев дается следующей теоремой.

Теорема (Кэли). Число t_p помеченных деревьев порядка p равно

$$t_p = p^{p-2}. \quad (106)$$

Мы дадим наброски только четырех доказательств, принадлежащих Кэли, Приюферу, Пойа и Кирхгофу, хотя существует много других доказательств.

Кэли высказал мысль, что существует соответствие между помеченными деревьями и функциями, отображающими множество из $p - 2$ объектов в множество из p объектов. Например, если $p = 5$, то существует 5^3 функций из $\{a, b, c\}$ в $\{v_1, v_2, v_3, v_4, v_5\}$. Эти функции перечисляются многочленом

$$(v_1 + v_2 + v_3 + v_4 + v_5)^3. \quad (107)$$

Слагаемые этого многочлена сопоставляются функциям естественным образом. Например, v_3^3 соответствует постоянной функции $f(x) = v_3$, слагаемое $3v_1v_3^2$ отвечает трем функциям, которые отображают только один элемент в v_1 , а два других — в v_3 , и $6v_2v_3v_5$ дает шесть функций, отображающих по одному элементу в v_2, v_3 и v_5 . Теперь, умножая многочлен (107) на $v_1v_2v_3v_4v_5$ и получая

$$(v_1 + v_2 + v_3 + v_4 + v_5)^3 v_1v_2v_3v_4v_5, \quad (108)$$

устанавливаем тем самым соответствие между слагаемыми из этого произведения и помеченными деревьями порядка 5. Это соответствие с использованием слагаемого $3v_1^2v_2v_3^3v_4v_5 = 3v_1v_3^2(v_1v_2v_3v_4v_5)$ демонстрируется на рис. 27.

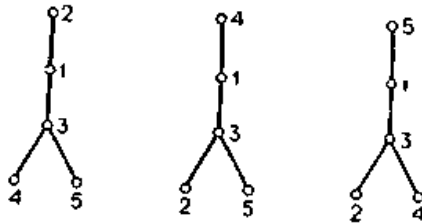


Рис. 27. Помеченные деревья, перечисляемые членом $v^2_1v_2v^3_3v_4v_5$.

Заметим, что в деревьях, соответствующих слагаемому $v^2_1v_2v^3_3v_4v_5$, степень вершины, помеченной числом k , равна показателю степени у v_k . Справедливость этого высказывания может быть установлена и в общем случае. Следовательно, число помеченных деревьев, у которых вершины, помеченные числом k , имеют степень d_k , равно полиномиальному коэффициенту

$$\binom{p-2}{d_1-1, d_2-1, \dots, d_p-1}. \quad (109)$$

Кэли проиллюстрировал это соответствие для $p = 6$ и не стал рассматривать другие случаи, заметив: «Сразу видно, что доказательство, данное для этого частного случая, применимо при любом значении p ».

Прюфер получил соответствие между помеченными деревьями порядка p и наборами из $p - 2$ элементов $(a_1, a_2, \dots, a_{p-2})$, где каждое a_k является целым числом от 1 до p , причем в наборах допускаются повторения. Таким образом, имеется p^{p-2} таких наборов. В заданном помеченном дереве T возьмем висячую вершину v , имеющую наименьшую пометку, и выберем пометку a_1 вершины, смежной с v . Далее, проделав такой же шаг для дерева $T - v$, находим пометку a_2 (дерево $T - v$ получено из дерева T удалением вершины v и ребра, инцидентного вершине v). Этот процесс оборвется, когда останутся только две смежные вершины. Заметим, что дерево, изображенное на рис. 28, соответствует последовательности $(2, 4, 1, 2, 4, 4)$.

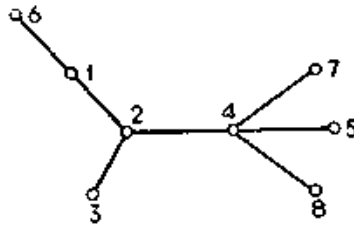


Рис. 28. Помеченное дерево восьмого порядка.

Так как каждое помеченное дерево порядка p дает единственный набор длины $p - 2$, то $t_p \leq p^{p-2}$.

Чтобы показать, что $t_p \leq p^{p-2}$, опишем некоторую процедуру, позволяющую однозначным образом строить по каждому набору $(a_1, a_2, \dots, a_{p-2})$ помеченное дерево. Следуя Муну, обозначим через b_1 наименьшее целое положительное число, не встречающееся в наборе $(a_1, a_2, \dots, a_{p-2})$, и пусть (c_2, \dots, c_{p-2}) обозначает набор длины

$p - 3$, получающийся из набора (a_2, \dots, a_{p-2}) уменьшением всех его координат, больших, чем b_1 , на 1. Тогда набор (c_2, \dots, c_{p-2}) состоит из чисел, принадлежащих множеству $\{1, \dots, p - 1\}$, и мы можем предположить, что существует соответствующее дерево T порядка $p - 1$. Изменим распределение пометок у вершин дерева T , прибавляя 1 к каждой пометке, превосходящей $b_1 - 1$. Затем введем p -ю вершину с пометкой b_1 и соединим ее с вершиной, помеченной числом a_1 в дереве T . Таким образом, получили единственное помеченное дерево, соответствующее данному набору длины $p - 2$. Далее рассмотрим метод Пойа для нахождения числа помеченных деревьев. Так как число корневых помеченных деревьев порядка p равно $p \cdot t_p$, то экспоненциальная производящая функция для этих деревьев дается выражением

$$y = \sum_{p=1}^{\infty} p t_p x^p / p! \tag{110}$$

Пойа нашел функциональное уравнение для y и затем для нахождения t_p применил формулу обращения Лагранжа.

Это функциональное уравнение для y будет сейчас выведено. Из леммы пересчета помеченных графов следует, что $y^n/n!$ является экспоненциальной производящей функцией для n -множеств корневых помеченных деревьев. Эти n -множества соответствуют в точности тем корневым помеченным деревьям, в которых корень имеет степень n и не помечен. Более точно, это соответствие получается так: сначала добавляем к каждому n -множеству новую вершину, не помечая ее, а затем соединяем эту новую вершину с каждым из старых корней. Описанная идея иллюстрируется рис. 29.

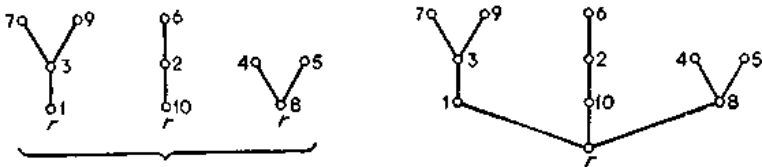


Рис. 29. 3-множество корневых деревьев и соответствующее ему дерево, корень которого имеет степень 3.

Умножение выражения $y^n/n!$ на x отвечает приписыванию пометки новому корню и включению его в число пересчитываемых вершин. Таким образом, $xy^n/n!$ перечисляет корневые помеченные деревья, в которых корень имеет степень n . Суммируя по n , получаем

$$y = \sum_{n=0}^{\infty} xy^n/n! \tag{111}$$

и, следовательно, мы приходим к функциональному уравнению

$$y = xe^y. \quad (112)$$

Чтобы найти решение уравнения (112), выразив y как функцию от x , мы применим весьма полезный частный случай формулы Лагранжа, приведенный в работе Муна.

Формула обращения Лагранжа. Если функция $\varphi(y)$ аналитична в некоторой окрестности точки $y = 0$ и $\varphi(0) \neq 0$, то уравнение

$$x = y/\varphi(y) \quad (113)$$

имеет единственное решение, задаваемое производящей функцией

$$y = \sum_{k=1}^{\infty} c_k x^k, \quad (114)$$

коэффициенты которой определяются по формулу

$$c_k = (1/k!) \left\{ (d/dy)^{k-1} (\varphi(y))^k \right\}_{y=0}. \quad (114)$$

Применяя эту формулу обращения к уравнению (112), где $\varphi(y) = e^y$, находим

$$y = \sum_{k=1}^{\infty} k^{k-1} x^k / k! \quad (116)$$

и, сравнивая это выражение с (110), снова получаем формулу (106) для t_p .

При решении некоторых задач перечисления помеченных объектов удобно использовать обобщение формулы (115), принадлежащее Лагранжу. В дополнение к условиям, наложенным на функцию φ , предположим, что дана еще одна функция $f(y)$, аналитическая в некоторой окрестности точки $y = 0$. Обобщенная формула Лагранжа утверждает, что функция $f(y)$ может быть выражена степенным рядом по x следующим образом:

$$f(y) = f(0) + \sum_{k=1}^{\infty} \frac{x^k}{k!} \left\{ \left(\frac{d}{dy} \right)^{k-1} [f'(y) \varphi^k(y)] \right\}_{y=0}. \quad (117)$$

При $f(y) = y$ эта формула приводит к выражениям (114) и (115). Доказательство формулы (117) можно найти у Гурса и Хедрика.

Более полезный результат, обычно называемый «Матричная теорема о деревьях», содержится в работе Кирхгофа. Число помеченных деревьев может быть быстро получено как простое следствие этого результата. Матрицей смежности $A = A(G) = [a_{ij}]$ помеченного графа G порядка p называется $(p \times p)$ -матрица, в которой $a_{ij} = 1$, если вершины i и j смежны, и $a_{ij} = 0$ в противном случае. Следовательно, существует взаимно однозначное соответствие между помеченными графами порядка p и симметрическими двоичными $((p \times p)$ -матрицами с

нулевыми диагональными элементами. Пусть $M(G)$ обозначает матрицу, которая получается из матрицы $-A$ с помощью подстановки на место i -го диагонального элемента в ней числа $\text{deg } v_i$. Подграф H графа G *покрывает* граф G , если каждая вершина графа G является вершиной подграфа H (рис. 30). Такой подграф называют *остовным*.

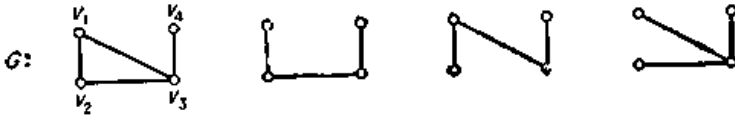


Рис. 30. Граф и три его остова.

Матричная теорема о деревьях для графов. Для всякого связного помеченного графа G все кофакторы (в другой терминологии — алгебраическое дополнение) матрицы $M(G)$ равны друг другу и их общее значение представляет собой число остовных деревьев графа G . (Остовное дерево называют также *остовом* или *каркасом*.)

Доказательство этой теоремы можно найти у Харари. Для иллюстрации сформулированного утверждения рассмотрим граф G , изображенный на рис. 30. Он имеет три остова, так как, например, кофактор, соответствующий элементу a_{14} матрицы

$$M(G) = \begin{bmatrix} 2 & -1 & -1 & 0 \\ -1 & 2 & -1 & 0 \\ -1 & -1 & 3 & -1 \\ 0 & 0 & -1 & 1 \end{bmatrix}, \text{ равен } \begin{vmatrix} -1 & 2 & -1 \\ -1 & -1 & 3 \\ 0 & 0 & -1 \end{vmatrix} = 3.$$

Полный граф K_p , в котором любые две вершины смежны, может быть помечен только одним способом, и каждый его остов соответствует своему помеченному дереву. Следовательно, число помеченных деревьев порядка p получается в результате применения матричной теоремы о деревьях к графу K_p . Каждый главный кофактор (то есть кофактор, соответствующий диагональному элементу матрицы) матрицы $M(K_p)$ есть определитель порядка $p - 1$:

$$\begin{vmatrix} p-1 & -1 & \dots & -1 \\ -1 & p-1 & \dots & -1 \\ \vdots & \vdots & \ddots & \vdots \\ -1 & -1 & \dots & p-1 \end{vmatrix}.$$

Вычитая первую строку из каждой другой и прибавляя последние $p - 2$ столбцов к первому, получаем верхнюю треугольную матрицу, определитель которой равен p^{p-2} .

2.8.6. Эйлеровы контуры в орграфах

В предыдущем разделе мы показали, что матричная теорема о деревьях для графов дает один из нескольких методов подсчета помеченных деревьев, основанный на определении числа остовов помеченного графа K_p . Сейчас мы приведем только формулировку утверждения, распространяющего эту теорему на орграфы и дающего число остовов данного орграфа D , каждый из которых ориентирован по направлению к некоторой своей вершине.

Предметом данного раздела является применение этой матричной теоремы о деревьях для орграфов к помеченному эйлерову орграфу D с целью получения точной формулы для числа эйлеровых контуров в орграфе D .

Входящее дерево получается из корневого дерева T с корнем v ориентированием всех его ребер по направлению к вершине v . *Выходящее дерево* является ориентированно-двойственным к входящему. Ясно, что оба эти множества ориентированных деревьев находятся во взаимном однозначном соответствии с множеством корневых деревьев.

Рассмотрим орграф D , который изображен на рис. 31 и вершины которого помечены числами 1, 2, 3, 4, 5.

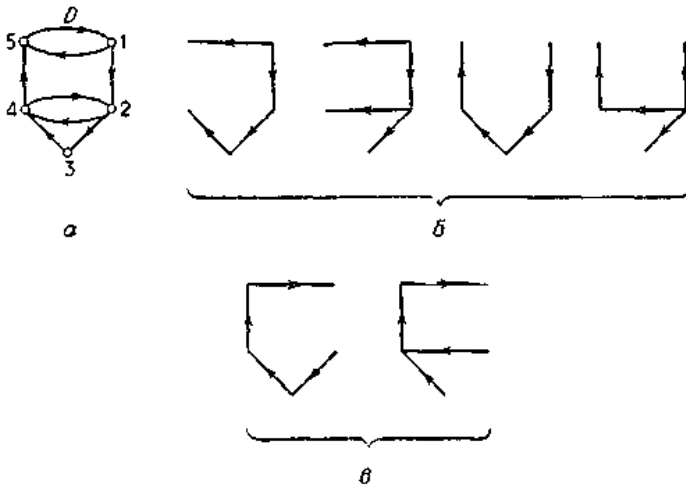


Рис. 31. Остовы орграфа D , выходящие и входящие в вершину, помеченную цифрой 1.

Существуют ровно четыре остовных дерева, выходящих из вершины 1, и два остовных дерева, входящих в вершину 1 (как показано на рис. 31, б и в).

Пусть D — орграф с матрицей смежности A . Определим диагональную матрицу M_{out} , у которой (i,i) -элемент равен $od\ v_i$ — полустепенени исхода вершины v_i . Затем положим $C_{out} = M_{out} - A$. Таким образом, сумма всех элементов матрицы C_{out} , стоящих в одной и той же строке, равна нулю, но необязательно это будет выполняться для столбцов. В действительности, как мы увидим вскоре, сумма элементов матрицы C_{out} , стоящих в одном и том же столбце, равна также нулю тогда и только тогда, когда D — эйлеров орграф. Аналогично определяется матрица $C_{in} = M_{in} - A$. Следующий важный результат был найден Воттом и Мейберри и доказан Таттом.

(1). Матричная теорема о деревьях для орграфов. *Все кофакторы i -й строки матрицы C_{out} равны друг другу, и их общее значение есть число остовов орграфа D , входящих в вершину v_i . Двойственным образом, общее значение кофакторов i -го столбца матрицы C_{in} равно числу остовов, выходящих из вершины v_i .*

Мы опускаем доказательство, но эту теорему легко проиллюстрировать на примере орграфа D , изображенного на рис. 31, для которого две матрицы, упомянутые в теореме, имеют вид

$$C_{out} = \begin{bmatrix} 2 & -1 & 0 & 0 & -1 \\ 0 & 2 & -1 & -1 & 0 \\ 0 & 0 & 1 & -1 & 0 \\ 0 & -1 & 0 & 2 & -1 \\ -1 & 0 & 0 & 0 & 1 \end{bmatrix},$$

$$C_{in} = \begin{bmatrix} 1 & -1 & 0 & 0 & -1 \\ 0 & 2 & -1 & -1 & 0 \\ 0 & 0 & 1 & -1 & 0 \\ 0 & -1 & 0 & 2 & -1 \\ -1 & 0 & 0 & 0 & 2 \end{bmatrix}.$$

Используя их, убеждаемся сразу, исходя из первой строки матрицы C_{out} и из первого столбца матрицы C_{in} , что орграф D имеет в точности четыре остова, выходящих из вершины 1, и два остова, входящих в эту вершину (как показано на рис. 31).

Орграф называется *эйлеровым*, если существует замкнутый остовный ориентированный маршрут, проходящий через каждую дугу точно один раз. Такой маршрут называется *эйлеровым контуром*. Один

критерий эйлеровости орграфа состоит в следующем: орграф должен быть связным, и для всякой его вершины полустепень исхода равна полустепени захода. Например, орграф, показанный на рис. 31, не является эйлеровым, а орграф D , изображенный на рис. 32, эйлеров.

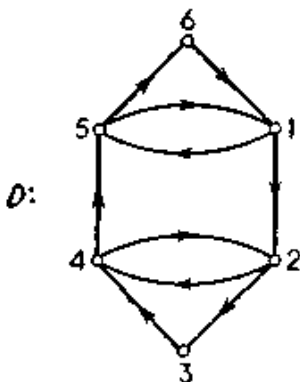


Рис. 32. Эйлеров орграф.

Из определения эйлерова орграфа следует, что матрицы C_{out} и C_{in} имеют одинаковые диагонали и поэтому равны. Для орграфа D , данного на рис. 32, эта матрица имеет вид

$$C = C_{out} = C_{in} = \begin{bmatrix} 2 & -1 & 0 & 0 & -1 & 0 \\ 0 & 2 & -1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & -1 & 0 & 2 & -1 & 0 \\ -1 & 0 & 0 & 0 & 2 & -1 \\ -1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Первый шаг в доказательстве матричной теоремы о деревьях для графов состоит в замечании, что если в матрице сумма элементов любой строки, так же как и сумма элементов любого столбца, равна нулю, то всякий кофактор матрицы имеет одно в то же значение. Следовательно, в силу теоремы 1, каждый эйлеров орграф имеет одно и то же число остовов, входящих в каждую вершину и выходящих из каждой вершины. Например, в приведенной выше матрице все кофакторы равны 4, так что существует по четыре остовных входящих дерева для каждой вершины орграфа (как показано на рис. 33).

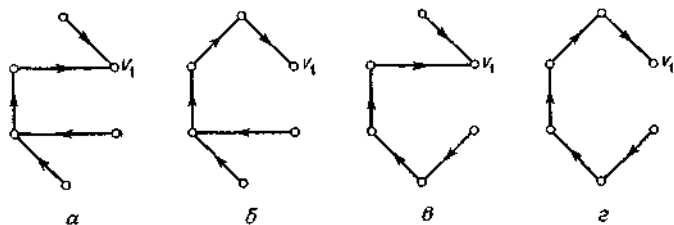


Рис. 33. Остовы, входящие в вершину v_i у орграфа, изображенного на рис. 32.

Теперь мы подготовлены к тому, чтобы применить матричную теорему о деревьях для орграфов к выводу формулы для числа эйлеровых контуров в данном орграфе. При доказательстве мы будем придерживаться изложения Кастелейна. Этот результат впервые был получен де Брёйном и ван Ардене-Эренфест, хотя частный случай был рассмотрен еще раньше в работе Смита и Татта. Так как для каждой вершины v_i эйлерова орграфа D выполняется равенство $od\ v_i = od\ v_i$, то мы можем обозначить это число через d_i .

Теорема. Число $e(D)$ эйлеровых контуров в помеченном эйлеровом орграфе D , у которого общее значение кофакторов матрицы $C = C_{out} = C_{in}$ равно c , задается формулой

$$e(D) = c \prod_i (d_i - 1)!. \tag{118}$$

Доказательство. Пусть v_j — произвольная вершина эйлерова орграфа D . Покажем, что каждый эйлеров контур E орграфа D определяет единственный остов T , входящий в вершину v_j , и что каждый такой остов T определяет в точности $\prod (d_i - 1)!$ эйлеровых контуров. А так как мы уже видели, что число остовов орграфа D , входящих в каждую вершину, равно c , то тем самым формула (118) будет доказана.

Чтобы построить остов, входящий в вершину v_j и определяемый данным эйлеровым контуром E в орграфе D , назовем *последней дугой* любой вершины $v_i \neq v_j$ такую дугу, выходящую из v_i , которая проходится последней при движении по контуру E , если началом (и концом) контура является вершина v_j . Таким образом, только вершина v_j не имеет последней дуги. Остов T определяется тогда как такой ориентированный остовный подграф орграфа D , у которого все дуги являются последними дугами. Так как в подграфе T полустепень исхода вершины v_j равна 0, а полустепени исхода всех других вершин равны 1, то он должен быть деревом, входящим в вершину v_j .

Пусть теперь T — некоторый остов, входящий в вершину v_j (один из c таких остовов). Чтобы построить все эйлеровы контуры E , связанные с

остовом T , мы действуем так же, как и в предыдущем абзаце, т. е. последними дугами контура E относительно вершины v_1 будут дуги остова T . Так как орграф D эйлеров, то, как мы уже отмечали, $\text{od } v_i = \text{id } v_i = d_i$. При построении контура E из остова T одна из дуг, выходящих из произвольной вершины $v_i \neq v_1$, сначала не учитывается, так как позднее используется как последняя дуга этой вершины; кроме того, одна дуга, выходящая из вершины v_1 , резервируется для использования в качестве первой дуги контура E . Итак, для каждой вершины v_i (включая вершину v_1) существует ровно $(d_i - 1)!$ различных упорядочений дуг, выходящих из v_i , по их появлению в контуре E . Так как эти выборы независимы, то, перемножая факториалы, получаем число эйлеровых контуров, определяемых остовом T . Но существует s таких остовов, что и доказывает формулу (118).

Следствие. *Если в эйлеровом орграфе каждое d_i равно либо 1, либо 2, то число эйлеровых контуров равно числу s остовов, входящих в произвольную вершину.*

Утверждение мгновенно следует из того, что каждый $(d_i - 1)! = 1$. Проиллюстрируем это следствие на примере орграфа D , который изображен на рис. 32 и в котором каждое d_i равно либо 1, либо 2. Из вычисления кофакторов, которое предшествовало рис. 33, мы знаем, что орграф D имеет в точности четыре эйлеровых контура. Они представлены на рис. 34 в соответствии с остовными входящими деревьями, показанными на рис. 33.

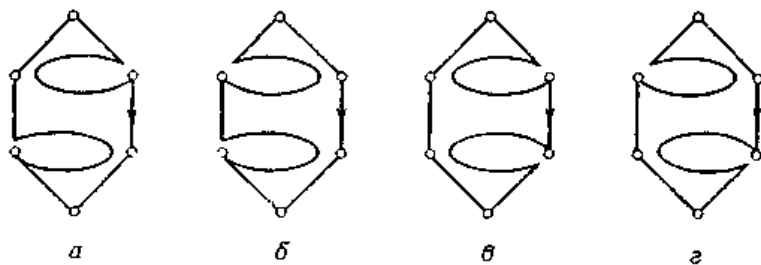


Рис. 34. Эйлеровы контуры в орграфе, изображенном на рис. 32

2.8.7. Раскрашенные графы

В п.2.8.3 мы видели, что вершины k -раскрашенного графа разбиваются на k множеств так, что смежные вершины всегда находятся в разных множествах. Вершины каждого множества рассматриваются

тогда как имеющие одинаковый цвет. В этом параграфе мы сосредоточим свое внимание на *2-раскрашенных графах*, для которых $k=2$.

Подставляя (62) в (64), получаем

$$g_p(x) = Z(\Gamma_1(K_p), 1+x). \tag{119}$$

Теперь весьма плодотворным оказывается обобщение этого уравнения путем замены K_p произвольным графом G порядка p . Из следствия теоремы Пойа, которое интерпретирует ряд $Z(A, 1+x)$, вытекает, что ряд $Z(\Gamma_1(G), 1+x)$ перечисляет $\Gamma_1(G)$ -эквивалентные классы множеств, состоящих из ребер графа G . Эти классы эквивалентности в точности соответствуют таким остовным подграфам графа G , каждые два из которых содержатся в одном и том же классе лишь тогда, когда существует автоморфизм графа G , переводящий один из этих подграфов в другой. Если два подграфа не принадлежат одному и тому же классу, то они называются *неподобными*.

Теорема. *Число неподобных остовных подграфов графа G , содержащих q ребер, равно коэффициенту при x^q в*

$$Z(\Gamma_1(G), 1+x). \tag{120}$$

Для иллюстрации этой теоремы рассмотрим (4, 5)-граф G , показанный на рис. 35.

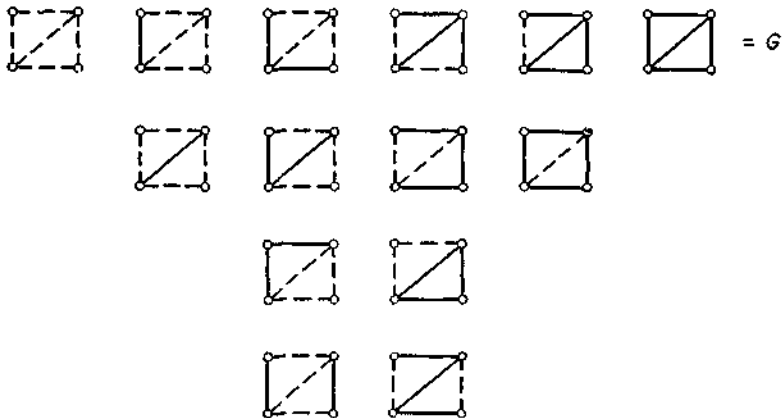


Рис. 35. Остовные подграфы некоторого графа.

Графаретные вычисления дают формулу

$$Z(\Gamma_1(G)) = \frac{1}{4} (s_1^5 + 3s_1s_2^2). \tag{121}$$

Следовательно,

$$Z(\Gamma_1(G), 1+x) = 1 + 2x + 4x^2 + 4x^3 + 2x^4 + x^5. \quad (122)$$

Правильность коэффициентов в этом многочлене подтверждается рис. 35. На этом рисунке штриховые ребра используются для указания на отсутствие ребра в подграфе, чтобы подчеркнуть эквивалентность относительно группы графа G . Заметим, что многочлен (122) дает также числа 2-раскрасок ребер графа G («сплошным» и «штриховым» цветами). Мы обнаружим, что эта интерпретация многочлена $Z(\Gamma_1(G), 1+x)$ весьма полезна, в частности при перечислении 2-раскрашенных графов, которые рассматриваются ниже.

Полный двудольный граф $K_{m,n}$ имеет $m+n$ вершин, из которых m окрашены в один цвет, а n — в другой, и две вершины смежны тогда и только тогда, когда они окрашены в разные цвета. Так как 2-раскрашенные графы с m вершинами одного цвета и n вершинами другого цвета соответствуют в точности остовным подграфам графа $K_{m,n}$ то из теоремы немедленно следует, что многочлен $b_{m,n}(x)$, перечисляющий эти 2-раскрашенные графы, удовлетворяет соотношению

$$b_{m,n}(x) = Z(\Gamma_1(K_{m,n}), 1+x). \quad (123)$$

Например,

$$b_{3,2}(x) = 1 + x + 3x^2 + 3x^3 + 3x^4 + x^5 + x^6 \quad (124)$$

и рис. 36 подтверждает правильность этих коэффициентов.

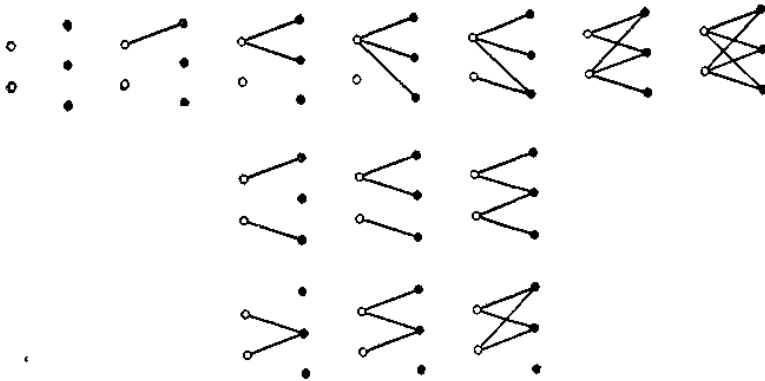


Рис. 36. 2-раскрашенные графы с тремя вершинами одного цвета и двумя вершинами другого цвета.

Заметим, что коэффициенты этих многочленов всегда симметричны «относительно концов».

Теперь, следуя Харари, найдем цикловой индекс реберной группы графа $K_{m,n}$. Проще всего начать со случая, когда $m \neq n$. Предположим, что X — множество из m вершин одного цвета в графе $K_{m,n}$, а Y — множество из n вершин другого цвета. Тогда упорядоченные пары (x, y) из декартова произведения $X \times Y$ в точности соответствуют ребрам графа $K_{m,n}$. Таким образом, подстановки из группы $\Gamma_1(K_{m,n})$ состоят из таких подстановок на множестве пар (x, y) , которые индуцируются подстановками множеств X и Y . Это наводит на мысль ввести следующую бинарную операцию на группах подстановок. Пусть A и B — группы подстановок с множествами объектов X и Y соответственно. *Декартовым произведением групп A и B* , обозначаемым $A \times B$, называется группа подстановок с множеством объектов $X \times Y$, состоящая из всех упорядоченных пар (α, β) подстановок α из A и β из B . Образ каждого элемента (x, y) из $X \times Y$, определяемый подстановкой (α, β) , имеет вид

$$(\alpha, \beta)(x, y) = (\alpha x, \beta y). \quad (125)$$

Следовательно, для $m \neq n$

$$\Gamma_1(K_{m,n}) = S_m \times S_n \quad (126)$$

и, подставляя это выражение в формулу (123), получаем теорему о перечислении таких 2-раскрашенных графов.

Теорема. *Многочлен $b_{m,n}(x)$, перечисляющий 2-раскрашенные графы, при $m \neq n$ дается формулой*

$$b_{m,n}(x) = Z(S_m \times S_n, 1 + x), \quad (127)$$

где

$$Z(S_m \times S_n) = \frac{1}{m! n!} \sum_{(\alpha, \beta)} \prod_{r, t=1}^{m, n} s_{\{r, t\}}^{(r, t)} I_r(\alpha) I_t(\beta). \quad (128)$$

На самом деле мы уже вычисляли цикловой индекс группы $S_m \times S_n$, когда обосновывали формулу (65) для $Z(S_p^{(2)})$. В частности, соотношение (72) дает в точности то выражение, которое стоит под знаками суммы и произведения в формуле (128).

Для иллюстрации теоремы воспользуемся формулами (127) и (128), чтобы установить вид многочлена $b_{3,2}(x)$, который уже приводился раньше в соотношении (124). Сначала с помощью формулы (128) найдем цикловой индекс декартова произведения $S_3 \times S_2$. В этой группе содержится двенадцать подстановок, но цикловая структура любой из них, скажем (α, β) , зависит только от цикловых структур подстановок α и β . Цикловые структуры этих конкретных подстановок получаются из формулы

$$Z(S_n) = (1/n!) \sum_{(j)} h(f) \prod_{k=1}^n s_k^{j_k},$$

которая выражает $Z(S_n)$ через разбиения числа n . Существуют только три разбиения числа 3 и только два разбиения числа 2; поэтому нам нужно рассмотреть ровно шесть различных видов пар подстановок. Например, если подстановка α из S_3 соответствует произведению $s_1 s_2$ в $Z(S_3)$, а β из S_2 является транспозицией, то, как следует из формулы (128), подстановка (α, β) дает вклад $s_1^2 s_2$ в цикловой индекс $Z(S_3 \times S_2)$. Так как существуют три подстановки с такой же цикловой структурой, как у α , то их общий вклад в цикловой индекс $Z(S_3 \times S_2)$, когда они находятся в паре с подстановкой β , составляет $3s_1^2 s_2$. Таблица 1 отражает все вклады, которые вносят в цикловой индекс $Z(S_3 \times S_2)$ пары из декартова произведения $S_3 \times S_2$.

Таблица 1

Члены циклового индекса $Z(S_3 \times S_2)$

Подстановка	α	β	(α, β)
Член циклового индекса	s_1^3	s_2^2	s_1^6
	$s_1 s_2$	$s_1^2 s_2$	$s_1^2 s_2^2$
	s_3	s_1^2	s_3^2
	s_1^3	s_2	$s_1^3 s_2$
	$s_1 s_2$	s_2	$s_1 s_2^2$
	s_3	s_2	$s_3 s_2$

Следовательно, формула для этого циклового индекса имеет вид

$$Z(S_3 \times S_2) = \frac{1}{12} (s_1^6 + 3s_1^3 s_2^2 + 2s_3^2 + 4s_2^3 + 2s_6). \quad (129)$$

Подставляя $1 + x^k$ вместо s_k в правую часть выражения (129), получаем многочлен $b_{3,2}(x)$, указанный в формуле (124).

Займемся теперь перечислением 2-раскрашенных графов для того более тонкого случая, когда $m = n$. Соответствующий перечисляющий многочлен обозначим через $b_n(x)$, а не через $b_{n,n}(x)$. Нетрудно, например, убедиться в том, что

$$b_3(x) = 1 + x + 2x^2 + 4x^3 + 5x^4 + 5x^5 + 4x^6 + 2x^7 + x^8 + x^9. \quad (130)$$

Рис. 36 подтверждает, что коэффициент при x^4 и, следовательно, при x^5 равен 5.

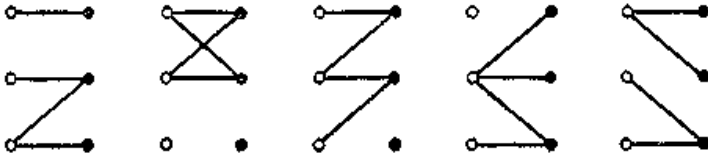


Рис. 36. Пять 2-раскрашенных (6,4)-графов с тремя вершинами каждого цвета.

Стоит заметить, что мы не получаем дополнительных изоморфных классов этих графов, если считаем, что цвета взаимозаменяемы.

Как и прежде, мы должны найти формулу для циклового индекса реберной группы графа $K_{n,n}$. Для этой цели удобно определить дополнительные операции на группах подстановок.

Пусть A и B — группы подстановок с множествами объектов $X = \{x_1, \dots, x_d\}$ и $Y = \{y_1, \dots, y_e\}$ соответственно. Композиция группы A относительно группы B обозначается через $A [B]$ и имеет в качестве множества объектов декартово произведение $X \times Y$. Это понятие было определено Пойа и другими математиками и иногда называлось «веночным произведением» («wreath product»). Для каждой подстановки α из A и каждой последовательности β_1, \dots, β_d из d подстановок, принадлежащих группе B , существует подстановка из $A [B]$, обозначаемая через $[\alpha; \beta_1, \dots, \beta_d]$ и такая, что для всякой упорядоченной пары $(x_i, y_j) \in X \times Y$ выполняется равенство

$$[\alpha; \beta_1, \dots, \beta_d](x_i, y_j) = (\alpha x_i, \beta_j y_j). \tag{131}$$

Основанием для того, чтобы назвать эту группу композицией групп A и B , является то, что ее цикловой индекс есть композиция цикловых индексов $Z(A)$ и $Z(B)$, как это проиллюстрировано в следующем примере. Возьмем $A = C_3$ и $B = S_2$. Используя различные буквы для переменных в $Z(A)$ и $Z(B)$, имеем

$$Z(A) = \frac{1}{3}(s_1^3 + 2s_3), \quad Z(B) = \frac{1}{2}(t_1^2 + t_2),$$

$$Z(A [B]) = \frac{1}{3} \left(\left[\frac{1}{2}(t_1^2 + t_2) \right]^3 + 2 \left[\frac{1}{2}(t_3^2 + t_6) \right] \right).$$

Теперь ясно, в каком смысле следует понимать, что цикловой индекс композиции двух групп представляет собой композицию их цикловых индексов. Пойа доказал, что $Z(A [B])$ есть многочлен, получающийся из $Z(A)$ заменой каждой переменной s_k цикловым индексом

$Z(B; s_k, s_{2k}, s_{3k}, \dots)$. Таким образом, $Z(B)$ нужно подставить в цикловой индекс группы A , чтобы получить цикловой индекс их композиции.

Эту процедуру мы часто обозначаем через

$$Z(A [B]) = Z(A) [Z(B)]. \quad (132)$$

Кроме того, для любых двух групп подстановок A и B и произвольного степенного ряда $f(x)$ мы имеем

$$Z(A [B], f(x)) = Z(A, Z(B, f(x))). \quad (133)$$

Наиболее типичный пример композиции двух групп подстановок получается при рассмотрении графа, состоящего из m непересекающихся экземпляров графа K_n . Группа такого графа идентична группе $S_m[S_n]$. Как указывалось ранее, граф G и его дополнение \bar{G} имеют одну и ту же группу автоморфизмов. Поэтому для описания группы графа $K_{n,n}$ мы используем тот факт, что дополнение графа $K_{n,n}$ состоит из двух экземпляров графа K_n , и получаем формулу

$$\Gamma(K_{n,n}) = S_2 [S_n]. \quad (134)$$

Так как группа $\Gamma_1(K_{n,n})$ индуцируется группой $\Gamma(K_{n,n})$, то для нахождения циклового индекса $Z(\Gamma_1(K_{n,n}))$ можно теперь воспользоваться соотношением (134). Подстановки из группы $\Gamma(K_{n,n})$, относительно которых два n -множества разного цвета инвариантны, соответствуют тем слагаемым в цикловом индексе $Z(S_2 [S_n])$, которые получаются с помощью замены переменной s_l в $Z(S_2)$ цикловым индексом $Z(S_n)$. Их вклад в $Z(\Gamma_1(K_{n,n}))$ составляет в точности $Z(S_n \times S_n)$. Те подстановки из $\Gamma(K_{n,n})$, которые взаимозаменяют два n -множества, вносят $Z(S_n; s_2, s_4, s_6, \dots)$ в цикловый индекс $Z(\Gamma(K_{n,n}))$. Таким образом, они состоят из циклов четной длины, которые всегда заменяют вершины одного цвета вершинами другого цвета. Их вклад в $Z(\Gamma_1(K_{n,n}))$, как показал Харари, равен

$$Z'_n = \frac{1}{n!} \sum_{(j)} \frac{n!}{\prod k^{j_k} j_k!} \prod_{k-\text{нечетное}} s_k^{j_k} \prod_k s_{2k}^{k \binom{j_k}{2} + [k/2] j_k} \prod_{r < t} s_{[r,t]}^{(r,t) j_r j_t}. \quad (135)$$

Эту формулу можно проверить так: рассматриваем циклы, составленные из вершин чередующихся цветов; затем выясняем, как эти циклы перемещают все ребра, соединяющие вершины разного цвета.

Итак, мы нашли, что

$$Z(\Gamma_1(K_{n,n})) = \frac{1}{2} (Z(S_n \times S_n) + Z'_n), \quad (136)$$

и можем написать формулу для этого циклового индекса, используя выражения (128) и (13). Однако существует еще одна операция на группах подстановок, которая в частном случае дает группу, идентичную группе $Z(\Gamma_1(K_{n,n}))$. Пусть A и B — группы подстановок с множествами объектов $X = \{x_1, \dots, x_d\}$ и $Y = \{y_1, \dots, y_e\}$, как и выше, но только теперь мы считаем, что $e > 1$. Экспоненциацией $[B]^A$ группы A

относительно группы B называется такая группа подстановок с множеством объектов Y^X , у которой подстановки строятся следующим образом. Каждая подстановка α из A и всякая последовательность β_1, \dots, β_d подстановок из B определяют ровно одну подстановку $(\alpha; \beta_1, \dots, \beta_d)$, принадлежащую группе $[B]^d$, и эта подстановка преобразует функцию f в функцию f^* , определяемую для всех $x_i \in X$ правилом

$$f^*(x_i) = \beta_i f(\alpha x_i). \quad (137)$$

Можно легко показать, что различные наборы подстановок $\alpha; \beta_1, \dots, \beta_d$ приводят к разным подстановкам на множестве Y^X и что эти подстановки образуют группу. Кроме того, если взять $A = S_2$ и $B = S_n$, то функции из множества Y^X соответствуют в точности ребрам графа $K_{n,n}$ и

$$\Gamma_1(K_{n,n}) = [S_n]^{S_2}. \quad (138)$$

Поэтому теорема перечисления может быть дана в следующей форме.

Теорема. *Многочлен $b_n(x)$, перечисляющий 2-раскрашенные графы с n вершинами каждого цвета, дается формулой*

$$b_n(x) = Z([S_n]^{S_2}, 1 + x), \quad (139)$$

где

$$Z([S_n]^{S_2}) = \frac{1}{2} (Z(S_n \times S_n) + Z'_n). \quad (140)$$

Как отмечалось выше, эта теорема перечисляет 2-раскрашенные графы с n вершинами каждого цвета, в которых цвета взаимозаменяемы. Простейший случай, когда цвета вершин неизменны, уже представлен формулой (127) — нужно положить $m = n$, даже хотя в той теореме предполагалось, что $m \neq n$.

Теперь проиллюстрируем теорему о многочлене $b_n(x)$, построив формулы, необходимые для нахождения $b_3(x)$. Цикловой индекс группы $S_3 \times S_3$, получаемый с помощью соотношения (128), имеет вид:

$$Z(S_3 \times S_3) = \frac{1}{36} (s_1^9 + 6s_1^3 s_2^3 + 8s_3^3 + 9s_1 s_2^4 + 12s_3 s_6). \quad (141)$$

Из формулы (135) следует, что

$$Z'_3 = \frac{1}{6} (s_1^3 s_2^3 + 3s_1 s_2^4 + 2s_3 s_6). \quad (142)$$

Комбинируя эти два соотношения, имеем:

$$Z([S_3]^{S_2}) = \frac{1}{72} (s_1^9 + 12s_1^3 s_2^3 + 8s_3^3 + 9s_1 s_2^4 + 18s_1 s_2^4 + 24s_3 s_6). \quad (143)$$

Теперь, подставляя в это выражение $1 + x^k$ вместо s_k , получаем формулу (130), дающую многочлен $b_3(x)$.

Граф называется *2-раскрашиваемым*, если он может быть 2-раскрашен. Мы только укажем, как могут быть перечислены такие графы. Харари и Принса установлено, что число связных 2-раскрашенных графов порядка p равно числу связных 2-раскрашиваемых графов порядка p . Имея перечисление 2-раскрашенных графов, мы можем перечислить связные 2-раскрашенные графы, используя метод, отражаемый формулой (81), и, таким образом, определить, кроме того, число связных 2-раскрашиваемых графов.

2.8.8. Корневые графы

Корневые помеченные графы были пересчитаны в процессе доказательства формулы определения числа связных помеченных графов. Перечисление корневых (непомеченных) графов является менее очевидным. Оно связано с построением парной группы путем незначительной переделки соответствующей симметрической группы. Эта модификация легко обобщается на корневые графы, у которых корень представляет собой произвольный порожденный подграф, могущий быть и графом, и орграфом, и мультиграфом.

В настоящем параграфе доказывается только одна теорема, а именно, теорема, дающая формулу для перечисления корневых графов G порядка p , у которых корнем является специальный порожденный подграф H , и при этом в графе G имеются q ребер, кроме тех, которые принадлежат подграфу H . Как проиллюстрировано в следствиях из этой теоремы, мы получим формулы для корневых графов, реберно-корневых графов, корневых графов, у которых в качестве корней выделены треугольники, корневых графов, у которых корнем является ориентированное ребро, и корневых графов с корнем, представляющим собой циклическую тройку. Например, все корневые (6,5)-графы, у которых в качестве корня взят треугольник $H=K_3$, представлены на рис. 37; здесь штриховыми ребрами выделен треугольник, являющийся корнем.

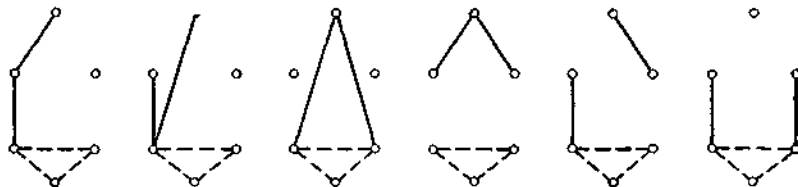


Рис. 37. (6,5)-графы, корнями которых являются треугольники.

Таким образом, в силу принятого выше соглашения, $p = 6$ и $q = 2$, ибо существуют два дополнительных ребра.

Пусть S — произвольное множество вершин графа G . *Порожденный подграф* $\langle S \rangle$ — это максимальный подграф графа G , имеющий S своим множеством вершин. Таким образом, подграф $\langle S \rangle$ содержит все ребра графа G , соединяющие две любые вершины множества S . Мы пытаемся найти число корневых графов G порядка p , у которых в качестве корня взят порожденный подграф H порядка $n < p$. Это означает, что в графе G нет ребер, которые соединяли бы две несмежные вершины подграфа H .

Группу подстановок, которая необходима нам для применения теоремы перечисления Пойа, будем обозначать через $\Gamma(H) \circ S_{p-n}$. Эта группа определяется следующим образом: положим $G = H \cup K_{p-n}$ и заметим, что произведение $\Gamma(H) S_{p-n}$ является подгруппой группы $\Gamma(G)$, оставляющей подграф H неподвижным. Нам требуется ограничение парной группы $(\Gamma(H) S_{p-n})^{(2)}$ на такие неупорядоченные пары $\{u, v\}$ вершин графа G , в которых обе вершины не принадлежат подграфу H . Результирующая группа подстановок как раз и обозначается через $\Gamma(H) \circ S_{p-n}$.

Если вклад некоторой подстановки в цикловой индекс $Z(S_p)$ составляет $\prod s_k^{j_k}$, то соответствующий вклад в цикловой индекс $Z(S_p^{(2)})$ обозначается через $(\prod s_k^{j_k})^{(2)}$. Теперь, когда все необходимые обозначения введены, можно сформулировать общий результат Харари и Палмер.

Теорема. Многочлен $h_p(x)$, перечисляющий корневые графы порядка p , у которых корнем является порожденный подграф H порядка $n < p$, дается формулой

$$h_p(x) = Z(\Gamma(H) \circ S_{p-n}, 1 + x), \tag{144}$$

где

$$Z(\Gamma(H) \circ S_{p-n}) = \frac{1}{|\Gamma(H)| (p-n)!} \sum_{(\alpha, \beta)} \prod_{r, i=1}^{n, p-n} s_{[r, i]}^{(r, i) I_r(\alpha) I_i(\beta)} (\prod s_k^{j_k(\beta)})^{(2)} \tag{145}$$

и суммирование ведется по всем парам подстановок $\alpha \in \Gamma(H)$ и $\beta \in S_{p-n}$.

Доказательство. Подстановки группы $\Gamma(H) \circ S_{p-n}$ индуцируются парами (α, β) подстановок α из $\Gamma(H)$ и β из S_{p-n} . Вклад в цикловой индекс $Z(\Gamma(H) \circ S_{p-n})$ вносимый произвольной парой, можно рассматривать как произведение двух сомножителей. В первом сомножителе мы просто отражаем тот факт, что подстановка (α, β) индуцирует подстановку на множестве ребер, соединяющих подграф H

с графом K_{p-n} абсолютно так, как это имело место в декартовом произведении. Второй сомножитель есть $(\prod s_k^{i_k})^{(2)}$, он описывает структуру подстановки β , индуцированной на парах вершин в графе K_{p-n} .

Теперь мы можем извлечь несколько следствий из этой теоремы, выбирая в качестве подграфа H подграфы частных типов. В нашем первом приложении мы рассмотрим корневые графы порядка p . Таким образом, $H = K_1$, так что $\Gamma(H) = S_1$ и, очевидно, $\Gamma(H) \circ S_{p-1} = (S_1 S_{p-1})^{(2)}$. Подставляя это выражение в формулу (144), получаем производящую функцию $G_p(x)$ для корневых графов:

$$G_p(x) = Z((S_1 S_{p-1})^{(2)}, 1+x), \quad (146)$$

которая совпадает с соответствующим результатом, полученным Харари.

Рассматривая только три вершины, находим

$$\begin{aligned} G_3(x) &= Z((S_1 S_2)^{(2)}, 1+x) = \frac{1}{2} [(1+x)^3 + (1+x)(1+x^2)] = \\ &= 1 + 2x + 2x^2 + x^3. \end{aligned}$$

Чтобы проиллюстрировать полученный результат для случая $p = 4$, замечаем, что

$$Z(S_1 S_3) = (1/3!) (s_1^4 + 3s_1^2 s_2 + 2s_1 s_3). \quad (147)$$

Использование данных из табл. 2 раз. 2.8.6 дает

$$Z((S_1 S_3)^{(2)}) = (1/3!) (s_1^6 + 3s_1^2 s_2^2 + 2s_3^2). \quad (148)$$

Подставляя $1+x$ в этот цикловой индекс, получаем многочлен

$$G_4(x) = 1 + 2x + 4x^2 + 6x^3 + 4x^4 + 2x^5 + x^6, \quad (149)$$

в правильности которого легко убедиться, просматривая графы четвертого порядка, изображенные на рис. 17. В общем случае для получения циклового индекса $Z((S_1 S_{p-1})^{(2)})$ можно, насколько это приемлемо, использовать формулы для цикловых индексов.

В следующей иллюстрации возьмем $H = K_2$, так что $h_p(x)$ перечисляет реберно-корневые графы:

$$h_p(x) = Z(S_2 \circ S_{p-2}, 1+x). \quad (150)$$

Тождество

$$Z(S_2 \circ S_{p-2}) = (\partial/\partial s_1) Z(S_p^{(2)}) \quad (151)$$

можно проверить. Поэтому мы можем воспользоваться формулами для цикловых индексов, чтобы при $p = 5$ получить соотношение

$$Z(S_2 \circ S_3) = \frac{1}{12} (s_1^5 + 4s_1^3 s_2 + 2s_3^2 + 3s_1 s_2^2 + 2s_3 s_4). \quad (152)$$

Подставляя $1 + x$, находим

$$h_5(x) = 1 + 2x + 6x^2 + 12x^3 + 16x^4 + 16x^5 + 12x^6 + 6x^7 + 2x^8 + x^9. \quad (153)$$

Так же легко получается и многочлен, перечисляющий корневые графы, у которых корнями являются треугольники. Приведем детали, относящиеся к перечислению таких графов с шестью вершинами. Так как $\Gamma(H) = S_3$, то, используя формулу (145), получаем

$$Z(S_3 \circ S_3) = \frac{1}{36} (s_1^{12} + 3s_1^4 s_2^4 + 6s_3^4 + 3s_1^6 s_2^3 + 9s_1^2 s_2^5 + 6s_3^2 s_6 + 2s_3^3 s_1^3 + 6s_1 s_2 s_3 s_6). \quad (154)$$

Из формулы (144) следует, что

$$h_6(x) = 1 + 2x + 6x^2 + 15x^3 + 21x^4 + 38x^5 + 44x^6 + 38x^7 + 21x^8 + 15x^9 + 6x^{10} + 2x^{11} + x^{12}.$$

Графы, перечисляемые коэффициентом при x^2 в этом многочлене, показаны на рис. 37; здесь для ребер треугольника, являющегося корнем, использованы штриховые линии.

Наша теорема справедлива также в том случае, когда H представляет собой орграф. Например, если H — ориентированное ребро, то $\Gamma(H) = E_2$, и поэтому

$$h_p(x) = Z(E_2 \circ S_{p-2}, 1 + x). \quad (155)$$

Выражение для этого циклового индекса получается из формулы (145).

Для каждой подстановки β из S_{p-2} обозначим через $\Pi S_k^{j_k}$ вклад, вносимый ею в цикловой индекс $Z(S_{p-2})$. Тогда

$$Z(E_2 \circ S_{p-2}) = \frac{1}{(p-2)!} \sum_{\beta} (\Pi S_k^{j_k})^2 (\Pi S_k^{j_k})^{(2)}, \quad (156)$$

где сумма берется по всем подстановкам β из S_{p-2} . Чтобы проиллюстрировать использование формулы (156) в случае $p = 5$, напомним, что $Z(S_3^{(2)}) = Z(S_3)$. Имеем

$$Z(E_2 \circ S_3) = (1/3!) ((s_1^2)^2 s_1^2 + 3(s_1 s_2)^2 s_1 s_2 + 2(s_3)^2 s_3). \quad (157)$$

Тогда из формулы (155) следует, что

$$h_5(x) = 1 + 3x + 9x^2 + 20x^3 + 27x^4 + 27x^5 + 20x^6 + 9x^7 + 3x^8 + x^9. \quad (158)$$

Графы, перечисляемые коэффициентами этого многочлена, можно построить без особого труда.

Наконец, мы найдем перечисляющий многочлен для корневых графов с шестью вершинами, у которых в качестве корня берется циклическая тройка. В этом случае

$$\Gamma(H) = \frac{1}{3} (s_1^3 + 2s_3). \quad (159)$$

Из формулы (145) получаем

$$Z(\Gamma(H) \circ S_3) = \frac{1}{18} (s_1^{12} + 3s_1^4 s_2^4 + 2s_1^3 s_3^3 + 6s_1 s_2 s_3 s_6 + 6s_3^4), \quad (160)$$

а из формулы (144) следует, что

$$h_6(x) = 1 + 2x + 6x^2 + 18x^3 + 34x^4 + 52x^5 + 62x^6 + 52x^7 + 34x^8 + 18x^9 + 6x^{10} + 2x^{11} + x^{12}. \quad (161)$$

Соответствующие графы с $q = 2$ показаны на рис. 38.

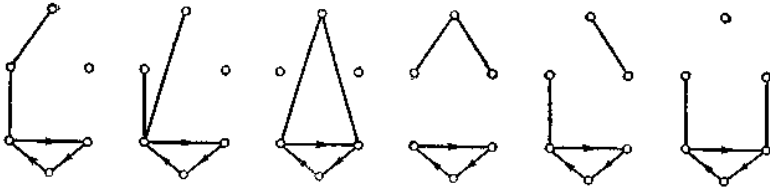


Рис. 38. (6,5)-графы, корнями которых являются циклические тройки.

Конечно, в качестве корня-подграфа H может быть выбрано много других графов. В частности, всякий раз когда граф или орграф H имеет единичную группу E_n , формула (145) принимает вид

$$Z(E_n \circ S_{p-n}) = \frac{1}{(p-n)!} \sum_{\beta} (\prod s_k^{j_k})^n (\prod s_k^{j_k})^{(2)}, \quad (162)$$

где, подобно тому, как это делалось в формуле (156), суммирование ведется по всем подстановкам β из группы S_{p-n} .

2.8.9. Надграфы и раскрашенные графы

Перечислению 2-раскрашенных графов был посвящен раз.2.8.7, а корневые графы, у которых в качестве корня выделялся некоторый порожденный подграф, изучались в предыдущем параграфе. Предметом данного параграфа является распространение этих исследований на случай перечисления таких корневых графов, у которых в качестве корня берется не обязательно порожденный

подграф; затем предполагается применить полученные выводы к перечислению m -раскрашенных графов. Приводимые ниже результаты представляют собой искусное сочетание довольно прозаического подхода из работы Харари с изящной методологией Робинсона.

Если H — подграф графа G , то G называется *надграфом* графа H . Вместо фразы «корневые графы G , у которых в качестве корня берется подграф H , не обязательно являющийся порожденным подграфом», мы предпочитаем говорить «надграфы графа H », что обеспечивает более эффективную мнемоническую схему.

Мы начнем с вывода формулы для числа неподобных p -вершинных надграфов графа H , также имеющего порядок p . Мы увидим в нижеследующем уравнении, что эта формула включает в себя цикловой индекс реберной группы дополнения графа H . Из метода, рассмотренного в предыдущем параграфе при перечислении корневых графов с корнями, представляющими собой порожденные подграфы H порядка n , следует, что группа подстановок, необходимая для исследования надграфов графа H , может быть получена расширением множества объектов группы $\Gamma(H) \circ S_{p-n}$ путем включения ребер из дополнения \bar{H} . В частности, если H — граф порядка p , то многочлен $r_p(x)$, перечисляющий корневиз надграфы графа H , дается формулой

$$r_p(x) = Z(\Gamma_1(\bar{H}), \mathbf{1} + x). \quad (163)$$

Интересен частный случай формулы (163), когда граф H представляет собой цикл C_p порядка p . В этом случае получаем формулу для подсчета общего числа неподобных гамильтоновых циклов, встречающихся во всех (гамильтоновых) графах порядка p . Реберную группу $\Gamma_1(\bar{C}_p)$ можно выразить на языке веночных произведений (композиций). Например, если $p=2n+1$, то имеем соотношение

$$\Gamma_1(\bar{C}_{2n+1}) = D_{2n+1}[E_{n-1}], \quad (164)$$

исходя из которого можно вычислить, используя формулы (132), цикловой индекс $Z(\Gamma_1(\bar{C}_{2n+1}))$. Если p — четное, то $Z(\Gamma_1(\bar{C}_{2n}))$ можно вычислить, умножая каждое слагаемое из $Z(D_{2n}[E_{n-1}])$ на соответствующее слагаемое из $Z(D_n)$.

В случае $p = 5$ формула (164) дает

$$\Gamma_1(\bar{C}_5) = D_5[E_1] = D_5$$

и имеем

$$Z(D_5) = \frac{1}{10} (s_1^5 + 4s_5 + 5s_1s_2^2). \quad (165)$$

Затем, применяя соотношение (163), получаем

$$r_5(x) = 1 + x + 2x^2 + 2x^3 + x^4 + x^5. \quad (166)$$

Правильность этого многочлена подтверждается графами, изображенными на рис. 39; здесь корень C_5 представляет собой цикл со штриховыми ребрами.

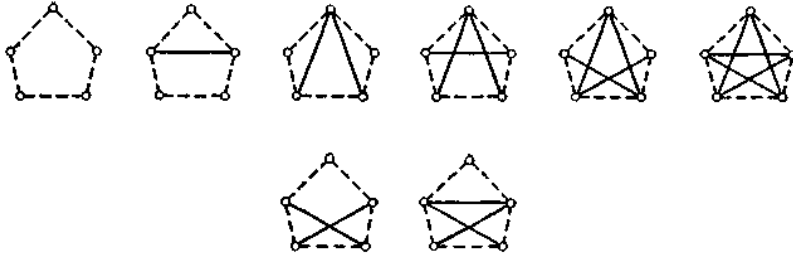


Рис. 39. Восемь графов пятого порядка, корнями которых являются остовные циклы.

Вернемся теперь к общему методу перечисления надграфов. Пусть H — произвольный подграф порядка n полного графа K_p . Тогда произведение $\Gamma(H)S_{p-n}$ есть группа, состоящая из всех таких подстановок группы $\Gamma(K_p) = S_p$, которые оставляют подграф H неподвижным, т. е. они могут переставлять вершины подграфа H только между собой. Группа, необходимая для перечисления надграфов графа H , является ограничением парной группы $(\Gamma(H) S_{p-n})^{(2)}$ на такие пары вершин, которые не смежны в H .

Обозначим это ограничение через $\Gamma_1(\bar{H}, K_p)$. Заметим, что если H имеет p вершин, то $\Gamma_1(\bar{H}, K_p) = \Gamma_1(\bar{H})$. Кроме того, если H — связный граф, то $\Gamma_1(\bar{H}, K_p)$ является в точности реберной группой дополнения графа H относительно графа K_p .

Мы пока только имеем теорему, дающую принципиальную возможность найти число надграфов данного графа, но у нас нет эффективного метода вычисления необходимого циклового индекса. Такой алгоритм был предложен Робинсоном. Как только эта процедура будет развита, мы тем самым докажем соответствующую теорему. Следовательно, при подобном положении дел чрезвычайно удобно в противоположность предостережению Уленбека сначала представить доказательство, а затем уже сформулировать теорему. Чтобы вычислить цикловой индекс группы $\Gamma_1(\bar{H}, K_p)$, мы определяем *вершинно-реберную группу* графа G , которая обозначается через $\Gamma_{0,1}(G)$, как группу подстановок, индуцированную группой $\Gamma(G)$ и переставляющую и вершины, и ребра графа G . Записывая цикловой

индекс этой группы, мы будем различать циклы, составленные из разных переставляемых объектов, используя для этого два множества переменных: s_k — для вершин и t_k — для ребер. Например, если G — (4,5)-граф $K_4 - x$, то

$$Z(\Gamma_{0,1}(K_4 - x)) = \frac{1}{4} (s_1^4 t_1^5 + 2s_1^3 s_2 t_1 t_2^2 + s_2^2 t_1 t_2^2). \quad (167)$$

Известно, что цикловой индекс группы $\Gamma(H) S_{p-n}$ дается соотношением

$$Z(\Gamma(H) S_{p-n}) = Z(\Gamma(H)) Z(S_{p-n}). \quad (168)$$

Аналогично, $Z(\Gamma_{0,1}(H)) Z(S_{p-n})$ есть цикловой индекс группы, которая индуцируется группой $Y(H)S_{p-n}$ действующей на множестве вершин графа K_p и множестве ребер графа H . Каждая подстановка $\alpha \in \Gamma(H)S_{p-n}$ вносит в $Z(\Gamma_{0,1}(H)) Z(S_{p-n})$ произведение $(\prod s_k^{j_k}) (\prod t_k^{i_k})$. Соответствующий вклад подстановки α в $Z(S_p^{(2)})$ обозначим, как обычно, через $(\prod s_k^{j_k})^{(2)}$. Далее, — и выделим особо этот прием, осуществленный Робинсоном, — подстановка α вносит в цикловой индекс группы $\Gamma_1(\bar{H}, K_p)$ произведение $(\prod s_k^{j_k})^{(2)} \prod s_k^{-i_k}$, ибо умножение на $\prod s_k^{-i_k}$ соответствует исключению тех членов в выражении $(\prod s_k^{j_k})^{(2)}$, которые возникают при рассмотрении пар вершин, являющихся смежными в графе H . Поэтому мы определяем линейный оператор ρ для вершинно-реберных цикловых индексов, задавая его для произведений таким образом:

$$\rho((\prod s_k^{j_k}) (\prod t_k^{i_k})) = (\prod s_k^{j_k})^{(2)} \prod s_k^{-i_k}. \quad (169)$$

Теперь утверждение о перечислении надграфов формулируется в следующем виде.

Теорема. Многочлен $r_p(x)$, перечисляющий надграфы порядка p произвольного графа H порядка $n \leq p$, дается формулой

$$r_p(x) = Z(\Gamma_1(\bar{H}, K_p), 1+x), \quad (170)$$

где

$$Z(\Gamma_1(\bar{H}, K_p)) = \rho(Z(\Gamma_{0,1}(H)) Z(S_{p-n})). \quad (171)$$

Заметим, что если H является графом порядка p , как было в (163), то формула (171) приобретает такой вид:

$$Z(\Gamma_1(\bar{H})) = \rho Z(\Gamma_{0,1}(H)). \quad (172)$$

Таким образом, оператор ρ дает нам возможность получить реберный цикловой индекс произвольного графа, исходя из вершинно-реберного циклового индекса его дополнения.

Мы проиллюстрируем эту теорему для случая $H = C_6$, т. е. взяв простой цикл порядка 6, и положим $p = 6$. Вершинно-реберный цикловой индекс произвольного циклического графа C_p легко вычисляется с использованием известной формулы, дающей выражение для $Z(C_p)$. Если $p = 6$, то имеем

$$Z(\Gamma_{0,1}(C_6)) = \frac{1}{12} (s_1^6 t_1^6 + s_2^3 t_2^3 + 2s_3^2 t_3^2 + 2s_6 t_6 + 3s_1^3 s_2^2 t_3^3 + 3s_3^2 t_1^2 t_2^2). \quad (173)$$

Применяя оператор ρ к соотношению (173), находим

$$\begin{aligned} Z(\Gamma_1(\bar{C}_6)) &= \frac{1}{12} (s_1^{15} s_1^{-6} + s_1^3 s_2^6 s_3^{-3} + 2s_3^5 s_3^{-2} + 2s_3 s_6^2 s_6^{-1} + \\ &+ 3s_1^3 s_2^3 s_2^{-3} + 3s_1^3 s_2^6 s_1^{-2} s_2^{-2}) = \\ &= \frac{1}{12} (s_1^9 + 4s_1^3 s_2^2 + 2s_3^3 + 2s_2 s_6 + 3s_1 s_2^4). \end{aligned} \quad (174)$$

Из (170) и (174) с помощью простых преобразований получаем явное выражение для $r_6(x)$:

$$\begin{aligned} r_6(x) &= 1 + 2x + 6x^2 + 12x^3 + 16x^4 + 16x^5 + \\ &+ 12x^6 + 6x^7 + 2x^8 + x^9. \end{aligned} \quad (175)$$

Этот многочлен можно также получить, используя соотношения, подобные (164), (165) и (166). Правильность коэффициента при x^2 подтверждается рис. 40.

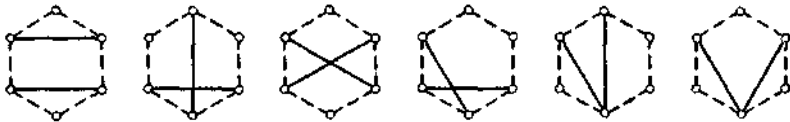


Рис. 40. Все шесть (6,8)-графов, корнями которых являются остовные циклы.

Сосредоточим теперь свое внимание на m -раскрашенных графах и на нижеследующем утверждении Робинсона; при этом мы будем использовать теорему, связанную с формулой (170). Как следует из определения, вершины m -раскрашенного графа разбиваются на m множеств так, что смежные вершины всегда принадлежат разным множествам. Вершины одного и того же множества имеют одинаковый цвет. Например, 3-раскрашенные графы, изображенные на рис. 41, содержат по две вершины каждого цвета и вершины, соединенные штриховыми ребрами, имеют одинаковый цвет.

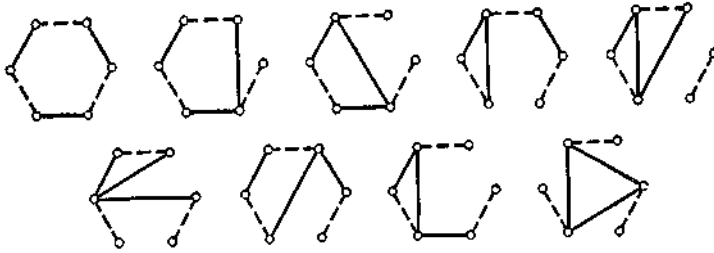


Рис. 41. Все девять 3-раскрашенных (6,3)-графов с двумя вершинами каждого цвета.

Как и ранее при перечислении 2-раскрашенных графов, мы ищем число m -раскрашенных графов для специального разбиения (j) числа p на m частей:

$$m = \sum_{k=1}^p j_k, \tag{176}$$

где, как обычно,

$$\sum_{k=1}^p k j_k = p.$$

Пусть $b_p(Q, x)$ — многочлен, переисляющий m -раскрашенные графы порядка p , для которых разбиение (j) является цветным. (То есть любое подмножество разбиения множества вершин состоит из вершин одинакового цвета и разные подмножества разбиения «имеют разный цвет»). Например, если $p = 6$ и разбиение имеет вид $2 + 2 + 2$, т. е. $j_2 = 3$, то многочлен $b_6((j), x)$ перечисляет 3-раскрашенные графы порядка 6, имеющие по две вершины каждого цвета. Мы покажем, что для этого разбиения числа 6 многочлен $b_6((j), x)$ имеет вид

$$b_6((j), x) = 1 + x + 4x^2 + 9x^3 + 18x^4 + 24x^5 + 30x^6 + 24x^7 + 18x^8 + 9x^9 + 4x^{10} + x^{11} + x^{12}. \tag{177}$$

Рис. 41 подтверждает тот факт, что коэффициент при x^3 в соотношении 177 равен 9.

Мы определяем *полный m -дольный граф* $K(j)$ для произвольного разбиения (j) числа p на m частей как дополнение объединения m полных графов:

$$K(j) = \overline{\bigcup_{k=1}^m j_k K_k}. \tag{178}$$

Как и прежде, m -раскрашенные графы, которые мы хотим подсчитать, являются основными подграфами графа $K(j)$. Из формулы (120) следует, что

$$b_p(j, x) = Z(\Gamma_1(K(j)), 1+x), \quad (179)$$

и поэтому мы должны найти метод для определения циклового индекса группы $\Gamma_1(K(j))$. Но остовные подграфы полного m -дольного графа

$K(j)$ являются как раз надграфами графа $\overline{K(j)}$. Этот факт

подсказывает идею: выразить цикловой индекс группы $\Gamma_1(K(j))$ на языке оператора ρ и циклового индекса $Z(\Gamma_{0,1}(\overline{K(j)}))$, взяв в формуле

(172) $\overline{H} = K(j)$. Таким образом, достаточно найти способ вычисления

циклового индекса группы $\Gamma_{0,1}(\overline{K(j)})$, и это мы сейчас сделаем.

Обозначим через nG граф, состоящий из n попарно непересекающихся экземпляров связного графа G . Как следует из определения веночного произведения,

$$\Gamma(nG) = S_n[\Gamma(G)]. \quad (180)$$

Поэтому, в силу соотношения (132), цикловой индекс для этой группы равен

$$Z(\Gamma(nG)) = Z(S_n)[Z(\Gamma(G))]. \quad (181)$$

Уравнение (181) легко обобщить и получить выражение

$$Z(\Gamma_{0,1}(nG)) = Z(S_n)[Z(\Gamma_{0,1}(G))]. \quad (182)$$

Рассмотрим тот же самый пример, который привел нас к многочлену (177). Полагая $G = K_2$ и $n = 3$ и используя формулу (182), имеем

$$Z(\Gamma_{0,1}(3K_2)) = Z(S_3)[Z(\Gamma_{0,1}(K_2))]. \quad (183)$$

Далее, $Z(\Gamma_{0,1}(K_2)) = \frac{1}{2}(s_1^2 t_1 + s_2 t_1)$, и если вместо каждого s_k

в цикловом индексе $Z(S_3)$ мы подставим выражение $(1/2)(s_k^2 t_k + s_{2k} t_k)$, получающееся из $Z(\Gamma_{0,1}(K_2))$ умножением каждого индекса у s_i и t_i на число k , то найдем

$$\begin{aligned} Z(\Gamma_{0,1}(3K_2)) &= \frac{1}{48} s_1^6 t_1^3 + \frac{1}{16} s_1^4 s_2 t_1^3 + \frac{1}{16} s_1^2 s_2^2 t_1^3 + \frac{1}{48} s_2^3 t_1^3 + \\ &+ \frac{1}{8} s_1^3 s_2^2 t_1 t_2 + \frac{1}{8} s_1^2 s_4 t_1 t_2 + \frac{1}{8} s_2^3 t_1 t_2 + \\ &+ \frac{1}{8} s_2 s_4 t_1 t_2 + \frac{1}{6} s_3^2 t_3 + \frac{1}{6} s_6 t_3. \end{aligned} \quad (184)$$

В общем случае, при рассмотрении графа $\overline{K(j)}$, компоненты которого являются полными графами, группа $\Gamma(\overline{K(j)})$ может быть представлена в виде произведения

$$\Gamma(\overline{K(j)}) = \prod_k S_{I_k}[\Gamma(K_k)]. \quad (185)$$

Из (182) следует, что

$$Z(\Gamma_{0,1}(\overline{K(j)})) = \prod_k Z(S_{I_k}) [Z(\Gamma_{0,1}(K_k))]. \quad (186)$$

Обозначим для удобства группу $\Gamma_{0,1}(K_n)$ через $S_n^{(1,2)}$ и заметим, что вершинно-реберный цикловой индекс легко получается из циклового индекса $Z(S_n^{(2)})$. Как отмечалось выше, $\overline{K(j)} = 3K_2$, если $p = 6$ и $j_2 = 3$. Таким образом, формула (184) дает цикловой индекс вершинно-реберной группы дополнения такого полного 3-дольного графа шестого порядка, который имеет по две вершины каждого цвета.

Теперь мы можем резюмировать метод Робинсона для перечисления t -раскрашенных графов.

Теорема. Многочлен $b_p(j, x)$, перечисляющий t -раскрашенные графы порядка p , отвечающие разбиению (j) , имеет вид

$$b_p(j, x) = Z(\Gamma_1(\overline{K(j)}), 1+x), \quad (187)$$

где

$$Z(\Gamma_1(K(j))) = \rho \left(\prod_k Z(S_{I_k}) [Z(S_k^{(1,2)})] \right). \quad (188).$$

В том примере, который нами рассматривался, мы уже нашли выражение для $Z(S_3)/[Z(S_2^{(1,2)})]$ (см. формулу (184)) Из определения оператора ρ в случае $j_2 = 3$ следует, что

$$\begin{aligned} Z(\Gamma_1(K(j))) = & \frac{1}{48} s_1^{15} s_1^{-3} + \frac{1}{16} s_1^7 s_2^4 s_1^{-3} + \frac{1}{16} s_1^3 s_2^6 s_1^{-3} + \\ & + \frac{1}{48} s_1^3 s_2^6 s_1^{-3} + \frac{1}{8} s_1^3 s_2^6 s_1^{-1} s_2^{-1} + \\ & + \frac{1}{8} s_1 s_2 s_4^3 s_1^{-1} s_2^{-1} + \frac{1}{8} s_1^3 s_2^6 s_1^{-1} s_2^{-1} + \\ & + \frac{1}{8} s_1 s_2 s_4^3 s_1^{-1} s_2^{-1} + \frac{1}{6} s_3^5 s_3^{-1} + \frac{1}{6} s_3 s_6^3 s_3^{-1}. \end{aligned} \quad (189),$$

Упрощая это выражение, имеем

$$\begin{aligned} Z(\Gamma_1(K(j))) = & \frac{1}{48} s_1^{12} + \frac{1}{16} s_1^4 s_2^4 + \frac{1}{12} s_2^6 + \frac{1}{4} s_1^2 s_2^6 + \\ & + \frac{1}{4} s_4^3 + \frac{1}{6} s_3^4 + \frac{1}{6} s_6^2. \end{aligned} \quad (190)$$

Наконец, делая подстановку $s_k = 1 + x^k$ в правую часть соотношения (190), получаем формулу (177). Используя эту теорему, Робинсон

нашел число 3-раскрашенных графов для значений $p \leq 9$ и для всевозможных разбиений.

2.8.10. Булевы функции

Булеву функцию n переменных можно рассматривать как отображение множества всех n -последовательностей из нулей и единиц в множество $\{0, 1\}$. Другими словами, булева функция представляет собой подмножество вершин n -куба Q_n . Нашей целью является подсчет числа типов булевых функций с отношением эквивалентности, определяемым группой n -куба Q_n .

Мы видели, что цикловой индекс реберной группы графа G позволяет подсчитывать остовные подграфы. В данном параграфе мы исследуем возможности циклового индекса вершинной группы графа G и увидим, что он полезен при подсчете неподобных множеств вершин. Наиболее интересное применение этой техники будет состоять в перечислении типов булевых функций.

Чтобы подготовиться к подсчету булевых функций, мы проинтерпретируем утверждение о цикловом индексе $Z(A, 1 + x)$ в случае, когда A есть группа $\Gamma(G)$ данного графа G .

Теорема 1. Коэффициент при x^r в $Z(\Gamma(G), 1 + x)$ равен числу неподобных множеств, состоящих в точности из r вершин, при условии, что эквивалентность определяется группой автоморфизмов графа G .

Для иллюстрации теоремы рассмотрим граф G , изображенный на рис. 35. Цикловой индекс его группы дается соотношением

$$Z(\Gamma(G)) = \left[\frac{1}{2} (s_1^2 + s_2) \right]^2 = \frac{1}{4} (s_1^4 + 2s_1^2s_2 + s_2^2), \quad (191)$$

так что

$$Z(\Gamma(G), 1 + x) = 1 + 2x + 3x^2 + 2x^3 + x^4. \quad (192)$$

Правильность среднего слагаемого этого многочлена подтверждается рис. 42, на котором пары вершин, образующие неподобные подмножества, обозначены темными точками.

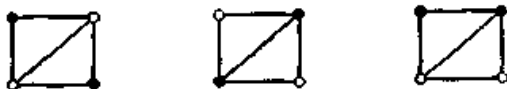


Рис. 42. Неподобные 2-множества вершин в некотором графе.

Пойа обнаружил, что ранние работы Джевонса и Клиффорда о перечислении типов булевых функций приводят к таким числовым

результатам, которые не являются полностью правильными. Задача состояла в нахождении числа «разных» высказываний, построенных из n предложений (элементарных высказываний), таких, что каждое из них имеет два значения истинности: 0 или 1. Используя символы \vee (или) и \wedge (и) для дизъюнкции и конъюнкции, можно из предложений P_1, P_2 и P_3 составить высказывание $P_1 \vee (P_2 \wedge P_3)$. Далее, если 1 и 0 — два значения истинности, то сложные высказывания соответствуют естественным образом булевым функциям. Например, высказывание $P_1 \vee (P_2 \wedge P_3)$ соответствует функции f , отображающей тройки (0, 0, 0), (0, 0, 1) и (0, 1, 0) в 0, а все остальные тройки — в 1. Однако мы хотим рассматривать два сложных высказывания как эквивалентные не только тогда, когда они соответствуют равным булевым функциям, но и в том случае, когда одно из этих высказываний может быть получено из другого перестановкой предложений или заменой нескольких предложений их отрицаниями. Так, например, высказывание $P_1 \vee (P_2 \wedge P_3)$ рассматривается как эквивалентное высказыванию $(P_3 \vee (\neg P_1)) \wedge (P_3 \vee P_2)$, что можно обосновать, используя дистрибутивный закон для дизъюнкции относительно конъюнкции.

Пойа обнаружил, что рассматриваемая задача состоит в определении числа способов, которыми можно раскрасить 2^n вершин n -куба Q_n двумя разными цветами. Это следует из того, что группа n -куба Q_n идентична группе, переставляющей произвольным образом n предложений и заменяющей предложения их отрицаниями. Мы можем выразить эту группу через симметрические группы, воспользовавшись операцией экспоненциации:

$$\Gamma(Q_n) = \{S_2\}^{S_n}. \quad (193)$$

В этом соотношении группа S_n указывает на то, что n предложений P_i можно переставлять как угодно, а группа S_2 учитывает возможность навешивания отрицания на предложения P_i после того, как они переставлены. Применяя теорему 1 для этой группы подстановок, можно получить основной результат данного параграфа. Цикловой индекс этой группы был впервые найден Слепяном, но представлен он был в довольно сложной форме, в которой невозможно узнать экспоненциацию как таковую.

Теорема 2. *Число $N(n, r)$ различных булевых функций, зависящих от n переменных и принимающих r ненулевых значений, равно коэффициенту при x^r в $Z([S_2]^{S_n}, 1+x)$.*

Для иллюстрации теоремы мы приводим на рис. 43 шесть способов раскраски вершин куба Q_3 , отвечающих тому случаю, когда в кубе имеется по четыре вершины каждого цвета.

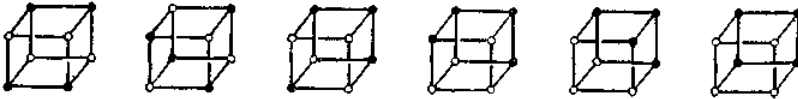


Рис. 43. Кубы с четырьмя вершинами каждого цвета.

Этим самым мы проверяем значение коэффициента при x^4 в $Z([S_2]^{S_3}, 1+x)$. Заметим, что ожерелья с четырех бусинок, соответствуют в точности булевым функциям от двух переменных, ибо всякое допустимое преобразование ожерелья с четырьмя бусинками можно заменить соответствующим преобразованием на 2-кубе. Используя тот факт, что группа $\Gamma(Q_n)$ абстрактно изоморфна вечноному произведению $S_n[S_2]$, Пойа вычислил цикловые индексы $Z(\Gamma(Q_n))$ для $n \leq 4$. Применяя сформулированную выше теорему, он нашел значения величины $N(n, r)$, представленные в табл. 1.

Таблица 1

Число булевых функций

$n \backslash r$	0	1	2	3	4	5	6	7	8	Общее число функций
1	1	1	1							3
2	1	1	2	1	1					6
3	1	1	3	3	6	3	3	1	1	22
4	1	1	4	6	19	27	50	56	74	402

Тесная связь, существующая между булевыми алгебрами и переключательными схемами, как раз явилась причиной, побудившей Слепьяна вычислить общее число булевых функций для $n = 5$ и 6.

Довольно сложный метод нахождения циклового индекса $Z([B]^{S_n})$ был предложен Харрисоном и Хаем, которые использовали этот метод для подсчета числа некоторых классов функций алгебры логики (функции Поста). Наиболее компактные формулы для $Z([B]^A)$ были найдены Робинсоном и Палмером. Мы дадим набросок некоторых деталей их подхода, чтобы снабдить читателя приемлемым средством вычисления циклового индекса, встречающегося в теореме 2. Сначала определяем произведение для одночленов, обозначая его через \times (сравните с соотношением (72)):

$$\prod s_k^{i,k} \times \prod s_k^{l,k} = \prod_{r,t} s_{[r,t]}^{(r,t)j_r t_j} \quad (194)$$

Как показал Харари, если это произведение распространить на многочлены по линейному закону, то цикловой индекс $Z(A \times B)$ декартова произведения можно представить в виде

$$Z(A \times B) = Z(A) \times Z(B). \quad (195)$$

Для каждого целого положительного числа r определяется новый оператор J_r . Сначала определяем его на одночленах:

$$J_r \left(\prod_{k=1}^n s_k^{j_k} \right) = \prod_{v=1}^{n^r} s_v^{i_v}, \quad (196)$$

где

$$\sum_{w|v} w i_w = \left(\sum_{k|v/(r, v)} k j_k \right)^{(r, v)}. \quad (197)$$

Ясно, что оператор J_1 является тождественным. Применяя к соотношению (197) формулу обращения Мёбиуса, можем написать

$$i_v = \frac{1}{v} \sum_{w|v} \mu(v/w) \left(\sum_{k|w/(r, w)} k j_k \right)^{(r, w)}. \quad (198)$$

Затем оператор J_r распространяется по линейному закону на многочлены. Приведенные только что пять формул можно использовать для выражения циклового индекса $Z(B^A)$ через цикловые индексы $Z(A)$ и $Z(B)$, где A и B — произвольные группы подстановок. Как это делается, станет ясно из рассмотрения примера, в котором $A = S_3$ и $B = S_2$, т. е. при вычислении циклового индекса группы куба Q_3 . Самой трудной частью является вычисление операторов J_r . Сначала вместо каждой переменной s_r в цикловом индексе $Z(S_3)$ подставим оператор J_r . Получим такое выражение:

$$Z(S_3; J_1, J_2, J_3) = \frac{1}{6} (J_1^3 + 3J_1J_2 + 2J_3). \quad (199)$$

Далее, слагаемые из (199) действуют на $Z(S_2)$ следующим образом:

$$\begin{aligned} J_1^3(Z(S_2)) &= J_1(Z(S_2)) \times J_1(Z(S_2)) \times J_1(Z(S_2)), \\ J_1J_2(Z(S_2)) &= J_1(Z(S_2)) \times J_2(Z(S_2)). \end{aligned} \quad (200)$$

В силу определения операторов J_r , имеем

$$J_1(Z(S_2)) = Z(S_2) = \frac{1}{2} (s_1^2 + s_2), \quad (201)$$

$$J_2(Z(S_2)) = \frac{1}{2} (J_2(s_1^2) + J_2(s_2)) = \frac{1}{2} (s_1^2s_2 + s_4), \quad (202)$$

$$J_3(Z(S_2)) = \frac{1}{2} (J_3(s_1^2) + J_3(s_2)) = \frac{1}{2} (s_1^2s_3^2 + s_2s_6). \quad (203)$$

Из соотношений (200) с учетом формул (195), (201) и (202) получаем равенства

$$J_1^3(Z(S_2)) = \frac{1}{2^3} (s_1^6 + 7s_4^2), \quad J_1J_2(Z(S_2)) = \frac{1}{2^2} (s_1^4s_2^2 + s_2^4 + 2s_4^2). \quad (204)$$

Наконец, подставляя многочлены из (203) и (204) в формулу (199) на место соответствующих слагаемых, находим выражение для циклового индекса $Z([S_2]^{S_3})$ в форме, напоминающей уравнения из работы Пойа:

$$Z([S_2]^{S_3}) = \frac{1}{6} \left(\frac{s_1^8 + 7s_2^4}{2^3} + 3 \frac{s_1^4 s_2^2 + s_2^4 + 2s_4^2}{2^2} + 2 \frac{s_1^2 s_3^2 + s_2 s_6}{2} \right). \quad (205)$$

Таким образом, как показано в работе Палмера, цикловой индекс $Z([B]^A)$ произвольной экспоненциации может быть выражен как образ циклового индекса $Z(B)$ при воздействии на него оператора $Z(A; J_1, J_2, \dots)$. На таком пути могут быть в принципе решены все задачи перечисления, в которых группа конфигураций может трактоваться как экспоненциация.

2.8.11. Хроматическое число

Предположим, что G есть неориентированный граф с однократными ребрами и без петель. Граф G называется k -раскрашиваемым, если существует такое разложение множества его вершин на k непересекающихся классов

$$C_1, C_2, \dots, C_k, \quad V = \bigcup_i C_i, \quad C_i \cap C_j = \emptyset, \quad (206)$$

что вершины в каждом классе независимы, т. е. что ребра в G соединяют вершины только из разных классов. Представление (206) называется k -раскраской графа G . Эта терминология подсказывается такой иллюстрацией, при которой каждый класс имеет свой цвет. Тогда каждая вершина окрашивается так, что концы любого ребра всегда имеют разный цвет. Можно также описывать цвета классов d в (206) целыми числами $1, 2, \dots, k$ и ввести функцию раскраски f , для которой

$$f(v_i) = i, \quad v_i \in C_i.$$

Наименьшее число $k = k(G)$ классов в возможной раскраске (206) называется *хроматическим числом* графа G . Тогда граф G — k -хроматический, и (206) есть *хроматическое разложение* V . Граф будет 2-хроматическим только тогда, когда он двудольный. Поэтому граф более чем 2-хроматический должен содержать нечетный простой цикл. Полный граф на n вершинах имеет хроматическое число n .

Имеется некоторый частный вид функции раскраски, иногда называемый *функцией Гранди*, которая характеризуется следующим свойством. Пусть для каждого v

$$\{v'_i\}, \quad i = 1, 2, \dots, \rho(v).$$

есть множество соседних вершин. Тогда $f(v)$ равно такому наименьшему числу из $1, 2, \dots$, что для всех i

$$f(v) \neq f(v'_i).$$

Очевидно,

$$f(v) \leq \rho(v) + 1.$$

Кроме того, для любого $f(v)$ значения

$$1, 2, \dots, f(v) - 1$$

должны достигаться на соседних вершинах.

Функцию Гранди можно построить следующим образом. Выберем максимальное независимое множество S_1 в V и положим

$$f(s_1) = 1, \quad s_1 \in S_1.$$

Все вершины в G соединены ребром с S_1 , так что получается $f(v) \geq 2$ для $v \notin S_1$. Затем выберем максимальное независимое множество S_2 в $G(V - S_1)$ и определим

$$f(s_2) = 2, \quad s_2 \in S_2.$$

Продолжая этот процесс, мы получим функцию Гранди.

Если функция Гранди удовлетворяет условию

$$\max_{v \in V} f(v) = k, \tag{207}$$

то она определяет раскраску в k цветов. Если граф G является k -хроматическим с разложением (206), то можно построить функцию Гранди, удовлетворяющую (207), следующим образом. Определим максимальное независимое множество S_1 как сумму

$$S_1 = C_1 \cup C'_2 \cup C'_3 \cup \dots \cup C'_k,$$

где C'_2 состоит из вершин C_2 , не соединенных с C_1 , C'_2 — из вершин C_3 , не соединенных с $C_1 \cup C'_2$, и т. д. Множество S_2 образуется из множеств

$$C_2 - S_1, C_3 - S_1, \dots$$

аналогично; так как граф G по условию k -хроматический, этот процесс закончится через k шагов.

Укажем несколько простых свойств хроматических чисел.

Теорема 1. Для числа независимости $\beta(G)$ и хроматического числа $k(G)$ графа G порядка n мы имеем

$$\rho(G)k(G) \geq n. \tag{208}$$

Доказательство. Так как все множества C_i в (206) являются независимыми, должно быть

$$v(C_i) \leq \beta(G),$$

и, следовательно,

$$n = \sum v(C_i) \leq k\beta(G) = k(G)\beta(G).$$

Столь же непосредственно получается

Теорема 2. *Если G содержит полный граф U_m на m вершинах, то $k(G) \geq m$.*

Рассмотрим теперь связь между раскрасками графа G и раскрасками его конечных частей. Возьмем такое конечное множество вершин F , что никакое пересечение $F \cap C_i$ в (206) не пусто. Ясно, что подграф $G(F)$ является $k(G)$ -раскрашиваемым. Если бы было

$$k_1 = k(G(F)) < k(G),$$

то нашлось бы k_1 -хроматическое разложение F в $G(F)$. Так как вершины в этих классах не остаются независимыми в G , можно, добавив конечное число ребер из G , повысим, хроматическое число для $G(F)$. Тогда мы получим, что G содержит конечные подграфы $G(F)$ с тем же хроматическим числом, что и G .

Обратный результат доказали де Брёйн и Эрдёш:

Теорема 3. *Если все конечные части графа G k -раскрашиваемые, то и сам граф G тоже k -раскрашиваемый.*

Доказательство следует из теоремы Радо. Строим двудольный граф $G(V, V')$, где V есть множество вершин для C , а V' — множество из k цветов. Для любого конечного множества $F \subset V$ определяется выбирающая функция при помощи такой раскраски вершин графа $G(F)$, что соединенные ребром вершины имеют различные цвета. Тогда существует выбирающая функция с этим же свойством для всего множества V .

Следующая теорема указывает верхнюю границу для хроматического числа.

Теорема 4. *Если ρ_1 есть точная верхняя граница для локальных степеней, то*

$$k(G) \leq \rho_1 + 1. \tag{209}$$

Доказательство. Согласно предыдущей теореме достаточно рассмотреть конечный граф G . Теорема очевидна для случаев $n = 1$ и $n = 2$; докажем ее в общем случае индукцией по n . Предположим, что граф G_{n-1} k -раскрашиваем, причем $k = \rho_1 + 1$. Добавление к G_{n-1} вершины v_n с $\rho(v_n) \leq \rho_1$ дает новый граф G_n , в котором v_n не соединена со всеми k классами C_i в k -раскраске (206) для G_{n-1} . Поэтому v_n можно добавить к одному из этих классов без увеличения k .

Границу (209) улучшил Брукс, доказавший, что

$$k(G) \leq \rho_1.$$

2.8.12. Суммы хроматических графов

Прежде всего отметим следующий факт.

Теорема 1. Для графов G и G' с одним и тем же свойством вершин мы имеем

$$k(G \cup G') \leq k(G) k(G'). \quad (210)$$

Доказательство. Пусть

$$V = \bigcup_{j=1}^k C'_j$$

есть k -раскраска G' . Из (206) мы получаем kk' -раскраску $G \cup G'$:

$$V = \bigcup C_i \cap C'_j.$$

Следующий результат получили Гаддам и Нордхауз:

Теорема 2. Для хроматических чисел графа G и его дополнения \bar{G} имеют место неравенства

$$2n^{1/2} \leq k(G) + k(\bar{G}) \leq n + 1, \quad (211)$$

$$n \leq k(G) k(\bar{G}) \leq \left(\frac{1}{2}(n + 1)\right)^2. \quad (212)$$

Доказательство. Нижняя граница в (212) получается из теоремы 1 для $G' = \bar{G}$. Из нее же получается и нижняя граница в (211) при помощи общего неравенства

$$(a + b)^2 \geq 4ab.$$

Это же неравенство дает верхнюю границу в (212) из верхней границы в (211). Таким образом, остается только доказать, что

$$k(G) + k(\bar{G}) \leq n + 1.$$

Это справедливо для $n = 1$ и $n = 2$ и может быть доказано в общем случае индукцией по n . Для графа G_n с n вершинами

$$G_n \cup \bar{G}_n = U_n.$$

По этому полному графу U_n построим U_{n+1} , добавляя вершину v_{n+1} . Присоединим q ребер из тех n ребер, которые идут от вершины v_{n+1} к V_n , к графу G_n ; остальные $n - q$ ребер присоединим к \bar{G}_n . В результате мы получим дополнительные друг для друга графы G_{n+1} и \bar{G}_{n+1} в U_{n+1} . Так как добавление к графу вершины и ее ребер может увеличить хроматическое число не более чем на единицу, мы получаем

$$k(G_{n+1}) \leq k(G_n) + 1, \quad k(\bar{G}_{n+1}) \leq k(\bar{G}_n) + 1. \quad (213)$$

Складывая эти два неравенства, получим требуемый результат, за исключением того случая, когда оба хроматических числа увеличились. Рассмотрим этот случай. Хроматическое число для G_n возрастает при присоединении v_{n+1} ; поэтому должно быть $q \geq k(G_n)$, так как иначе v_{n+1} можно было бы добавить к одному из классов C_1 в (206) без нарушения его независимости. Точно так же

$$n - q \geq k(\bar{G}_n),$$

так что в этом случае

$$k(G_n) + k(\bar{G}_n) \leq n,$$

и требуемое неравенство следует из индуктивного предположения и (213).

Пусть, например, граф G двудольный. Тогда

$$\frac{1}{2} n \leq k(\bar{G}) \leq n - 1.$$

Можно построить специальные примеры, показывающие, что указанные в теореме 2 границы не могут быть улучшены.

Каждая раскраска вида (206) определяет гомоморфное отображение графа G на граф G/C , при котором множествами прообразов элементов G/C являются независимые классы C_i . В G/C пара вершин (C_i, C_j) соединена ребром тогда и только тогда, когда существует ребро в G

$$(c_i, c_j), \quad c_i \in C_i, \quad c_j \in C_j.$$

Но если $k = k(G)$, то такое ребро всегда должно существовать, так как иначе $C_i \cup C_j$ можно было бы объединить в один класс. Отсюда следует

Теорема 3. Пусть G имеет k -хроматическое разложение (206). Тогда G имеет такой независимый гомоморфизм, что его образ, граф G/C , является полным графом на k вершинах.

Используем это представление для получения некоторых аддитивных разложений k -хроматического графа. Пусть $a < k$ — некоторое целое положительное число; разделим k на a с остатком:

$$k = ab + r, \quad 0 \leq r \leq a - 1. \quad (214)$$

Разложим теперь каждый из k классов C_i в (206) соответственно на классы:

$$\begin{array}{cccc}
 C_{11} & C_{12} & \dots & C_{1a}, \\
 C_{21} & C_{22} & \dots & C_{2a}, \\
 \dots & \dots & \dots & \dots \\
 C_{b1} & C_{b2} & \dots & C_{ba}, \\
 C_{b+1,1} & C_{b+1,2} & \dots & C_{b+1,r}.
 \end{array} \tag{215}$$

Построим две части A и B графа следующим образом. Граф A состоит из всех ребер в G , соединяющих любые два класса в различных строках в (215), а граф B состоит из всех ребер, соединяющих классы в двух различных столбцах. Из этого определения ясно, что

$$G = A \cup B, \tag{216}$$

причем граф A является $(b + 1)$ -раскрашиваемым, а граф B — a -раскрашиваемым при $r \neq 0$. Для $r = 0$ мы получаем теорему Зыкова:

Теорема 4. Если хроматическое число графа G равно произведению $k = ab$, то G есть сумма (216) двух таких графов, что граф A является b -хроматическим, а граф B — a -хроматическим.

Доказательство. Неравенство (210) показывает, что эти хроматические числа не могут быть меньше, чем a и b .

Аналогичные соображения можно применить в некоторых случаях при $b \neq 0$ в (214). Всегда должно быть $k(A) = b+1$, так как $k(A) \leq b$ приводит к противоречию, что видно из (210). Но может быть $k(B) \leq a - 1$; это приводит к условию

$$a \geq r + b + 1 \geq 3,$$

не удовлетворяющемуся для $n = 2$. Следовательно, G всегда можно записать в виде суммы (216), где

$$k(A) = b + 1, \quad k(B) = 2,$$

т. е. B двудольный, а A можно взять таким, что

$$k(A) = \frac{1}{2} k, \quad k(A) = \frac{1}{2} (k + 1),$$

зависимости от четности k . Повторяя эту операцию, мы получим следующую теорему:

Теорема 5. k -хроматический граф можно представить как сумму t двудольных графов, где t есть наименьший показатель, для которого $k \leq 2^t$.

2.8.13. Критические графы

Целый ряд свойств хроматических графов получил Дирак при помощи понятия *критического графа*. Граф K называется (вершинно) *критическим*, если удаление любой вершины с ее ребрами уменьшает

хроматическое число $k = k(K)$. Критическим 1-хроматическим графом является одна вершина; критическим 2-хроматическим графом — одно ребро. Критический 3-хроматический граф становится двудольным графом после удаления одной вершины, следовательно, он является простым циклом нечетной длины.

Теорема 1. *Критический граф K имеет следующие свойства:*

1. *Граф K конечен и связан.*
2. *Для любой вершины $\rho(v) \geq k - 1$,* (217)
3. *Граф K не имеет разделяющих вершин.*

Доказательство. 1. Мы видели, что k -хроматический граф содержит конечный подграф с тем же хроматическим числом. Следовательно, каждый k -хроматический граф содержит конечный критический граф с тем же хроматическим числом. Если бы граф K не был связным, то какую-нибудь из его компонент можно было бы удалить, не изменяя k .

2. Удаление вершины и ее ребер может уменьшить хроматическое число не более чем на единицу. Если v удалить из K , то полученный граф будет иметь хроматическое число $k - 1$. Если бы было $\rho(v) < k - 1$, то эту вершину можно было бы вернуть на место без увеличения хроматического числа.

3. Допустим, что K имеет разделяющую вершину s . Тогда K состоит из двух подграфов $K(A)$ и $K(B)$, имеющих только одну общую вершину s . Хроматические числа для $K(A)$ и $K(B)$ не превосходят $k - 1$, и, окрасив s в один цвет в обоих графах, мы получим, что K можно раскрасить в $k - 1$ цветов.

Результат 3 в теореме 1 можно обобщить. Пусть A — подмножество множества вершин V критического графа K . Если A и все ребра от его вершин удалить из K , то останется граф $K(V - A)$, который, вообще говоря, распадается на несколько связанных компонент:

$$K(V - A) = K(C_1) \cup \dots \cup K(C_i), \quad (218)$$

где число $i = i(A)$ есть индекс компонент множества A . По предположению граф $K(A)$ раскрашивается менее чем k цветами. Обозначим через $\mu(A)$ число существенно различных способов раскрашивания $K(A)$, т.е. раскрашиваний, которые нельзя получить друг из друга переименованием цветов. Подсчитывая эти способы, можно всегда предполагать, что какая-то одна вершина $a_1 \in A$ имеет вполне определенный цвет, обозначаемый через 1.

Теорема 2. *Для любого подмножества A множества вершин V критического графа K должно быть*

$$i(A) \leq \mu(A). \quad (219)$$

Доказательство. По условию граф $K(V - C_1)$ можно раскрасить в

$k - 1$ цветов. Поэтому каждый из графов

$$K(C_j \cup A), \quad j = 2, 3, \dots, i(A), \quad (220)$$

можно раскрасить в $k - 1$ цветов так, что множество A имеет одно и то же раскрашивание во всех графах. То же самое можно сделать и для каждого графа $K(V - C_j)$. Таким образом, если число $i(A)$ компонент превосходит число способов $\mu(A)$, которыми может быть раскрашен $K(A)$, то должна найтись пара, скажем $K(V - C_1)$ и $K(V - C_2)$, в которой A имеет одно и то же раскрашивание. Но тогда все графы (220), включая $j = 1$, могут быть $(k - 1)$ -раскрашены с общим раскрашиванием для A . Это дает раскрашивание K в $k - 1$ цветов. Следовательно, (219) должно выполняться.

Если $K(A)$ — полный граф, то имеется только одно существенное раскрашивание; следовательно, справедлива

Теорема 3. *Критический граф не может быть разделен вершинами содержащегося в нем полного графа.*

Каждый критический граф имеет соответствующие простые цепи L_0 максимальной длины и простые циклы C_0 максимальной длины. Обозначим соответственно через $\lambda_k(n)$ и $\gamma_k(n)$ наименьшие длины L_0 и C_0 для различных k -критических графов порядка n . Следующий результат принадлежит Дж. Б. Келли и Л. М. Келли:

Теорема 4.

$$\lim_{n \rightarrow \infty} \lambda_k(n) = \lim_{n \rightarrow \infty} \gamma_k(n) = \infty. \quad (221)$$

Доказательство. Множество A с v вершинами имеет не более чем v^{v-1} существенно различных раскрашиваний. Из теоремы 2 мы получаем, что

$$i(A) \leq v^{v-1}.$$

Следовательно, для простой цепи L длины l

$$i(L) \leq (l + 1)^l,$$

и для критического графа, в котором максимальная длина простой цепи равна l_0 ,

$$i(L) \leq (l_0 + 1)^{l_0}.$$

Теорема 4 показывает, что локальные степени в K для данного l_0 ограничены, и из известного неравенства следует, что при возрастании n должно l_0 стремиться к бесконечности.

Вторая часть (221) вытекает из следующего факта, обнаруженного Дираком.

Теорема 5. *В графе без разделяющих вершин*

$$c_0^2 > l_0, \quad (222)$$

где l_0 и c_0 — соответственно длины длиннейших простых цепей и простых циклов.

Доказательство. Пусть $L_0(a, B)$ —длиннейшая простая цепь. Известно, что существует простой цикл C , проходящий через a и b . Простая цепь L_0 распадается на участки L_1, \dots, L_i так, что каждая L_i или имеет на C только свои концы, или целиком лежит на C . В обоих случаях L_i является цепью на простом цикле, так что длина c_i цепи L_i меньше c_0 . Таким образом,

$$l_0 = \sum c_i < tc_0,$$

и, так как $t < c_0$, мы получаем неравенство (222).

Дальнейшее изучение асимптотического поведения функции $\gamma_k(n)$ из (221) проводили Дирак и Рид.

Свойство 2 в теореме 1 допускает следующее обобщение.

Теорема 6. *Любой критический k -хроматический граф по крайней мере $(k - 1)$ -реберно связан.*

Доказательство. Допустим, что для некоторого множества A существует разложение

$$G = G(A) \cup G(\bar{A}) \cup G(A, \bar{A}), \quad A \cup \bar{A} = V,$$

с

$$\rho = \rho(A, \bar{A}) < k - 1. \tag{223}$$

Графы $G(A)$ и $G(\bar{A})$ имеют функции раскраски f и \bar{f} со значениями $1, \dots, k - 1$. Покажем, что если выполняется (223), то эти функции могут быть объединены в раскрашивание целого графа G в $k - 1$ цветов, что противоречит предположению. Пусть

$$a_1, a_2, \dots, a_e$$

— вершины в A , от которых отходят ребра к \bar{A} . Если среди них имеется ровно r различных цветов $f(a_i)$, то обозначим эти цвета через $1, 2, \dots, r$ и переобозначим вершины так, чтобы было

$$f(a_1) = 1 \leq f(a_2) \leq \dots \leq f(a_e) = r.$$

Затем упорядочим произвольным образом ребра

$$E_t = (a_i, \bar{a}_j), \quad t = 1, 2, \dots, \rho, \quad \rho = \rho(A, \bar{A}),$$

но так, чтобы i не убывало. Для $E_t = (a_i, \bar{a}_j)$ перерасположим значения \bar{f} так, чтобы было $\bar{f}(\bar{a}_j) = 2$. Для последующих ребер будем брать значения

$$\bar{f}(\bar{a}_j) = t + 1,$$

перерасполагая, если нужно, значения \bar{f} на каждом шаге. Так как $\rho < k - 1$, этот процесс может быть проведен для всех ребер E_r . Он приведет к значениям, удовлетворяющим условиям

$$f(a_i) \neq \bar{f}(\bar{a}_i)$$

для каждого ребра E_r . Но тогда f и \bar{f} можно объединить в $(k - 1)$ -раскраску для G .

Среди различных нерешенных проблем для критических графов можно сформулировать следующую важную проблему.

Предположение Хедвигера. Каждый связный конечный k -хроматический граф имеет некоторую часть с полным k -графом в качестве образа при связном гомоморфизме.

Это предположение тривиально выполняется для $k = 1$ и $k = 2$. Для $k = 3$ каждый граф с циклом имеет связный гомоморфизм на треугольник. Для $k = 4$ предположение доказано Дираком в общей форме:

Теорема 7. *Если G — конечный связный граф с однократными ребрами, без разделяющих вершин и с локальными степенями $\rho(v) \geq 3$, то G имеет часть гомоморфную полному 4-графу при связном гомоморфизме.*

Доказательство. Пусть C — простой цикл наибольшей длины γ в G ; по известной теореме $\gamma \geq 4$. Обозначим вершины в C через a_1, \dots, a_γ . Хордой T назовем простую цепь, связывающую две несоседние вершины в C . Докажем сначала, что в каждой вершине на C , например в a_1 , существуют хорды. Имеется хотя бы одно ребро $E = (a_1, b)$, не принадлежащее C . Если b лежит на C , то все доказано; предположим поэтому, что b не принадлежит C . Из теоремы 3 следует, что существует простой цикл C_1 , содержащий E и a_2 . Простая цепь $C_1(a, b, a_2)$ должна иметь первую вершину a_i на C , и, так как C есть длиннейший простой цикл, мы получаем $a_i \neq a_2$ и $a_i \neq a_n$.

Предположим (рис. 44), что C имеет две хорды $T_1(a_i, a_1)$ и $T_2(a_k, a_1)$ с общей вершиной t вне C .

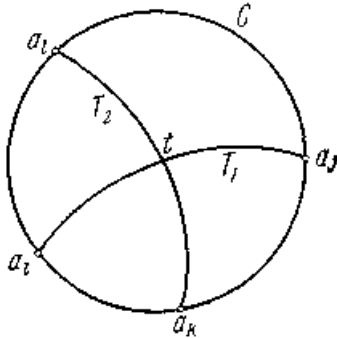


Рис. 44.

Примем за t первую вершину на T_1 , принадлежащую также T_2 . Тогда часть, состоящая из C и T_2 вместе с $T_1(a_i, a_j)$, гомоморфна полному 4-графу при связном гомоморфизме.

Остается рассмотреть случаи, когда все хорды простого цикла C не пересекаются (рис. 45). Любая хорда $T_1(a_i, a_j)$ делит C на две простые цепи.

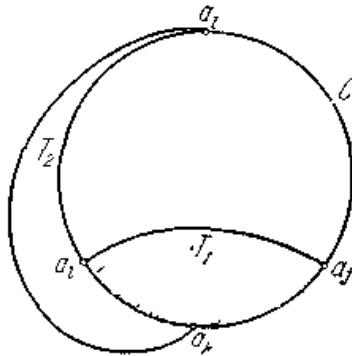


Рис. 45.

Возьмем a_i и a_j , так, чтобы цепь $C(a_i, a_j)$ была как можно короче; возьмем a_k на этой простой цепи. Существует некоторая хорда $T_2(a_k, a_l)$ и, по нашему предположению, вершина a_l не может быть расположена на $C(a_i, a_j)$. Граф, состоящий из C , T_1 и T_2 , будет связно гомоморфен полному 4-графу.

Следующий случай $k = 5$. Для таких графов прогресс очень мал, и это не удивительно, так как можно доказать, что справедливость предположения Хедвигера в этом случае приводила бы к доказательству так называемой гипотезы четырех красок для плоских графов.

2.8.14. Полиномы раскрашиваний

Предположим, что в нашем распоряжении имеется фиксированное число λ цветов для сопоставления их вершинам графа G с множеством вершин V . Если V имеет n элементов, то это можно сделать λ^n способами. Мы хотим определить число $\pi(G; \lambda)$ допустимых раскрасок, удовлетворяющих тому условию, что концы каждого ребра раскрашены различно. Обозначим через $\mu(E)$ число раскрашиваний графа G , в которых концы ребра E имеют один цвет, и вообще через

$$\mu(H) = \mu(E_{i_1}, E_{i_2}, \dots, E_{i_s}) \quad (224)$$

— число раскрашиваний, в которых концы каждого ребра части

$$H = \{ E_{i_1}, \dots, E_{i_s} \} \quad (225)$$

получают один цвет. Известный принцип теории множеств дает формулу

$$\pi(G; \lambda) = \lambda^n - \sum_i \mu(E_i) + \sum_{i,j} \mu(E_i, E_j) - \dots, \quad (226)$$

где суммы берутся соответственно по всем ребрам E_i по всем парам различных ребер E_i, E_j и т. д.

Чтобы вычислить общий член (224) в представлении (226), разложим граф H в (225) на связанные компоненты:

$$H = \bigcup_{i=1}^{\alpha(H)} H_i. \quad (227)$$

Пусть V есть множество вершин H . Если ребра в H имеют одинаковый цвет на обоих концах, то все вершины любой связанной компоненты H_i в (227) должны иметь один и тот же цвет. Поэтому величина (224) равна

$$\mu(H) = \lambda^{\alpha(H)}.$$

Подставляя это в (226), мы получим формулу

$$\pi(G; \lambda) = \sum_{s,c} (-1)^s p(s, c) \lambda^c, \quad (228)$$

где $p(s, c)$ есть число частей H графа G с s ребрами и $c = c(H)$ связными компонентами.

Формула (228) принадлежит Биркгофу. Если этот полином раскрашиваний $\pi(G; x)$ известен, то хроматическое число k_0 графа G будет наименьшим положительным целым числом, для которого

$$\pi(G; k_0) > 0.$$

При фактическом вычислении $\pi(G; \lambda)$ многие члены $\mu(H)$ в (226) исчезают. Возможность таких редукций была систематически

исследована Уитни. Чтобы получить его результаты, расположим ребра G в некотором определенном порядке:

$$E_1, E_2, \dots, E_N, \tag{229}$$

где N есть число ребер в G . Пусть $\{C_i\}$ — семейство всех простых циклов в G . По каждому из них построим *разорванный цикл*

$$P_i = C_i - F_i \tag{230}$$

полученный из C_i после удаления ребра F_i с наибольшим номером в (229). Расположим разорванные циклы лексикографически:

$$P_1, P_2, \dots, P_i \tag{231}$$

по отношению к порядку ребер E_i из (229), которые они содержат. Тогда ребро F_i в (230) не может содержаться ни в каком разорванном цикле P_j , предшествующем P_i , так как F_i имеет больший номер, чем все ребра в P_i и в P_j .

Разобьем, далее, все части графа G на $t + 1$ непересекающихся классов

$$S_1, S_2, \dots, S_{t+1} \tag{232}$$

следующим образом. S_1 состоит из всех частей, содержащих P_1 ; класс S_2 — из всех частей, содержащих P_2 и не содержащих P_1 , и т. д.; S_{t+1} состоит из всех частей, не содержащих разорванных циклов.

Покажем, что все члены в (226), соответствующие графам из S_j , исчезают. Очевидно, графы из S_j могут быть сгруппированы в пары взаимно однозначным образом:

$$H_1 \rightleftharpoons H_1 \cup F_1, \tag{233}$$

где H_1 не содержит F_1 . Два графа в (233) имеют одинаковое число связанных компонент, так как F_1 соединяет в H_1 два конца из P_1 ; следовательно,

$$\mu(H_1) = \mu(H_1 \cup F_1).$$

Эти два члена входят в (226) с противоположными знаками, так как один имеет на одно ребро больше, чем другой.

Аналогично для каждого графа H_2 из S_2 , не содержащего F_2 , в S_2 найдется единственный граф $H_2 \cup F_2$, так как F_2 не содержится в P_1 .

То же рассуждение снова показывает, что соответствующие члены в (226) исчезают. Повторяя этот процесс, мы получаем, что при вычислении (226) можно ограничиваться членами $\mu(H_{t+1})$, в которых H_{t+1} не содержат никакого из разорванных циклов (231).

Эти графы H_{t+1} не могут иметь циклов. Отсюда следует, что

$$c(H_{t+1}) = n - s,$$

и мы получаем выведенное Уитни выражение для полинома раскрасиваний:

$$\kappa(G; x) = \sum_s (-1)^s m(s) x^{n-s}, \tag{234}$$

где $m(s)$ есть число частей графа G с s ребрами, не содержащих разорванных циклов (231). Сравнивая коэффициенты при x^i в двух выражениях (228) и (234) для $\pi(G; x)$, мы получаем формулу

$$(-1)^{n-i} m(n-i) = \sum_s (-1)^s p(s, i). \quad (235)$$

Приведем несколько простых свойств полиномов раскрываний. Если G не имеет ребер, то, очевидно,

$$\pi(G; x) = x^n.$$

Вообще, так какнет частей с $c = 0$, условия

$$p(s, 0) = 0, \quad m(n) = 0$$

выполняются, и $\pi(G; x)$ имеет множитель x . Для одного ребра $H = E$

$$c(E) = n-1,$$

и мы получаем

$$p(1, n-1) = N, \quad p(1, j) = 0, \quad j \neq n-1.$$

Аналогично

$$p(s, n-s) = \binom{N}{s}, \quad p(s, j) = 0, \quad j \neq n-s,$$

если G не имеет простых циклов длины $\leq s$. Следующим соотношением будет

$$\sum_j p(s, j) = \binom{N}{s}, \quad (236)$$

вытекающее из того, что сумма слева представляет собой число частей графа G с s ребрами. Из (14.4.5) мы находим

$$\pi(G; 1) = \sum_{s,c} (-1)^s p(s, c),$$

и, согласно (228), получаем

$$\pi(G; 1) = \sum (-1)^s \binom{N}{s} = (1-1)^N = 0 \quad (237)$$

при $N > 0$.

Определение полинома раскрываний показывает, что если G имеет непересекающееся разложение

$$G = \bigcup_{i=1}^h G_i$$

на связные компоненты, то

$$\pi(G; x) = \pi(G_1; x) \dots \pi(G_h; x). \quad (238)$$

Аналогично пусть

$$G = G_1 \cup G_2$$

— разложение, в котором G_1 и G_2 имеют только одну общую вершину a . Тогда все раскрашивания G получаются из раскрашиваний G_1 и G_2 , в которых a имеет один и тот же цвет. Это сразу приводит к формуле

$$\pi(G; x) = \frac{1}{x} \cdot \pi(G_1; x) \cdot \pi(G_2; x). \quad (239)$$

Теорема 1. Если E — разделяющее ребро для G , то

$$\pi(G; x) = \left(1 - \frac{1}{x}\right) \pi(G - E; x). \quad (240)$$

Доказательство. Разобьем части графа G на два класса

$$\{H\}, \quad \{H \cup E\},$$

где H пробегает все части графа $G - E$. Граф $H \cup E$ имеет на одно ребро больше, чем H , и на одну связную компоненту меньше, так что

$$\pi(G; x) = \sum (-1)^s \mu(H) + \frac{1}{x} \sum (-1)^{s+1} \mu(H).$$

Просуммировав, мы получим (240).

Если применить теорему 1 к дереву T , то получится

$$\pi(T; x) = x(x-1)^{n-1}. \quad (241)$$

Для графа без циклов полином раскрашиваний, согласно (238), является произведением множителей (241).

Очевидно, в простом цикле C с n ребрами часть H с s ребрами имеет

$$c(H) = n - s, \quad s < n,$$

компонент. Это дает формулу

$$\pi(G; x) = (x-1)^n + (-1)^n (x-1). \quad (242)$$

Объединяя формулы (238) и (239) с результатом теоремы 1, мы получаем, что вычисление полинома раскрашиваний графа сводится к нахождению этих полиномов для его блоков.

Полиномы раскрашиваний можно также вычислять последовательными редукциями при помощи формулы

$$\pi(G \cup E; x) = \pi(G; x) - \pi(G'; x),$$

где $G \cup E$ — граф, полученный из G добавлением нового ребра $E = (a, b)$ между двумя его вершинами, а G' получается, если считать, что a и b совпадают. Доказательство является простым следствием определения полиномов раскрашиваний.

2.8.15. Группы автоморфизмов

Каждый граф G , ориентированный или неориентированный, имеет группу автоморфизмов $\Gamma = \Gamma(G)$, состоящую из изоморфизмов G на себя. Это значит, что группа Γ (автоморфизмов) графа G состоит из всех таких взаимно однозначных отображений α множества вершин V на себя, что если $E = (a, b)$ есть (ориентированное) ребро в G , то $E^\alpha = (a^\alpha, b^\alpha)$ также является (ориентированным) ребром, и наоборот. Таким образом, Γ можно рассматривать как группу подстановок на множестве V . Если граф G определяется матрицей смежности (вершин) $M(G)$, то есть автоморфизмами будут те перестановочные матрицы M_α , которые коммутируют с $M(G)$.

(Перестановочной называется матрица, в каждой строке и в каждом столбце которой находится ровно по одному единичному элементу, а все прочие ее элементы равны нулю. Умножение любой матрицы A на перестановочную матрицу P слева означает некоторую перестановку строк матрицы A . Умножение A на P справа равносильно некоторой перестановке столбцов A .)

В дальнейшем будем предполагать, что граф G неориентированный и, если не оговорено противное, — с однократными ребрами и без петель. Первая возникающая здесь задача — это фактическое нахождение группы $T(G)$. Полный граф U_n с n вершинами имеет своей группой симметрическую группу S_n и, наоборот, если группой является S_n , то граф полный. Полезно также отметить, что граф и его дополнение должны иметь одну и ту же группу.

Заметим далее, что нахождение группы можно свести к случаю, когда G связан. Пусть

$$G = G_1 \cup G_2 \cup \dots \cup G_r, \quad G_i = \cup G_{ij} \quad (243)$$

есть разложение G на связанные компоненты G_{ij} , расположенные так, что графы

$$G_{ij}, \quad j = 1, 2, \dots, n_i, \quad (244)$$

изоморфны друг с другом и не изоморфны с другими компонентами. При любом автоморфизме графы G_i должны преобразовываться в себя, так что группа графа G будет прямым произведением

$$\Gamma(G) = \Gamma(G_1) \times \dots \times \Gamma(G_r), \quad (245)$$

соответствующим (243).

Изоморфные компоненты G_{ij} из (244) можно переводить друг в друга, и, наряду с этой операцией, можно к каждой компоненте G_{ij} применить автоморфизм h_j . Таким образом, группа автоморфизмов $\Gamma(G_i)$ в (245) состоит из всех отображений, которые могут быть записаны в виде обобщенной подстановки:

$$P(h) = \begin{pmatrix} X_j \\ X_{k_j} \cdot h_{k_j} \end{pmatrix} \quad (j = 1, 2, \dots, n_1).$$

В обычной теоретико-групповой терминологии она называется *полной мономиальной группой* n , переменных над группой H , изоморфной любой из групп автоморфизмов $\Gamma(G_{ij})$, $j = 1, 2, \dots, n$.

Если G — связный граф, то сразу появляются некоторые ограничения на возможные автоморфизмы. Любая вершина должна переходить в вершину с той же локальной степенью, ребро — в ребро с тем же числом ребер в каждом конце. Вообще, любая часть графа имеет изоморфный образ с изоморфно расположенными соединяющими ребрами и вершинами. Несмотря на эти ограничения, фактическое нахождение группы для больших графов может оказаться очень сложным.

Приведем несколько иллюстраций. Оба графа на рис. 46 и 47 являются однородными степени 3.

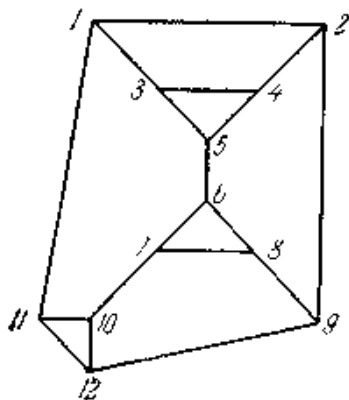


Рис. 46

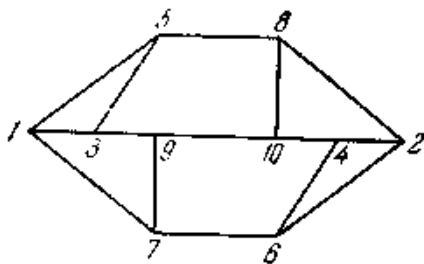


Рис. 47

Первый из них имеет единичную группу, а второй — группу порядка 2, определяемую, как показал Фрухт, подстановкой

$$P = (1, 2)(3, 4) (5, 6) (7, 8) (9, 10).$$

Группой простого цикла является группа диэдра, порождаемая циклической подстановкой вершин $(1, 2, \dots, n)$ и отражением $(1, n - 1) (2, n - 2) \dots$

Группа тетраэдра есть симметрическая группа S_4 на четырех элементах, так как он является полным графом с 4 вершинами. Группы куба и октаэдра обе изоморфны прямому произведению группы S_4 и отражения порядка 2. Группы додекаэдра и икосаэдра изоморфны прямому произведению знакопеременной группы A_5 на 5 элементах и отражения.

Так называемый *граф Петерсена* имеет вид, изображенный на рис. 48.

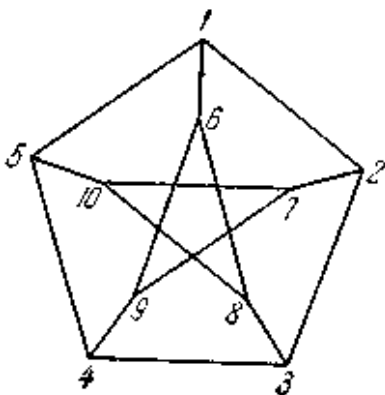


Рис. 48

Это однородный граф степени 3 и порядка 10. Он был впервые введен Петерсеном в качестве примера графа с $\rho = 3$, который не является суммой трех графов первой степени. Как установил Фрухт, его группа изоморфна S_5 .

В геометрических теориях приходится иметь дело с целым рядом конкретных графов. В качестве примера возьмем конфигурации, иллюстрирующие теоремы Паппа и Дезарга в проективной геометрии. Группы этих графов были найдены Каньо. Эта задача сразу сводится к предыдущим случаям, так как граф Дезарга оказывается дополнением графа Петерсена, а граф Паппа имеет дополнение, которое состоит из трех компонент, являющихся треугольниками.

Кроме тривиального случая одной вершины, все графы небольших порядков имеют неединичные группы. Согласно Каньо, граф порядка 6, изображенный на рис. 49, является графом наименьшего порядка с единичной группой.

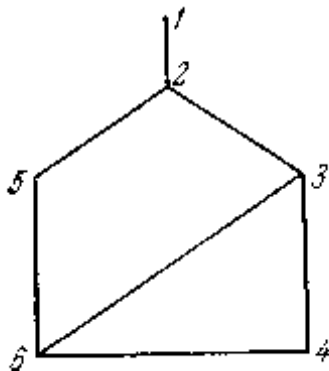


Рис. 49

Каньо исследовал также все графы порядка ≤ 6 , в которых локальные степени удовлетворяют условию $\rho(v) \geq 3$. Все они имеют нетривиальные группы. Графы этого типа порядка 7 могут иметь своей группой только единичную; пример приводится на рис. 15.1.5.

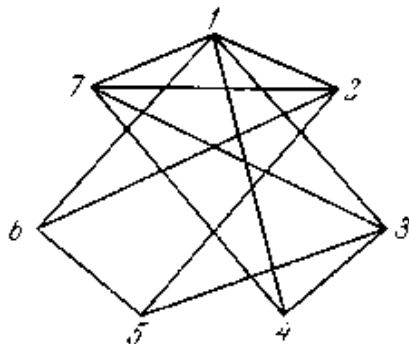


Рис. 50.

Другой иллюстрацией группы автоморфизмов графа будут графы кристаллов и группы кристаллов. Следует заметить, однако, что при определении кристаллографических групп интересуются несколько иной задачей, именно, автоморфизмами, получающимися при ортогональных преобразованиях трехмерного пространства. Наконец, отметим также, что автоморфизмы деревьев были обстоятельно исследованы Пойа.

2.8.16. Цветные графы Кэли для групп

Опишем кратко представление групп при помощи графов, которое было введено Кэли. Пусть Γ есть некоторая группа конечного или бесконечного порядка N . Группе Γ соотнесем некоторый граф G , множество вершин которого совпадает с множеством элементов Γ . Любую пару вершин соединяем ориентированным ребром $E_{ij} = (g_i, g_j)$, так что G является полным ориентированным графом с N входящими и выходящими ребрами в каждой вершине. В частности, между любыми двумя вершинами g_i и g_j имеется пара ребер (g_i, g_j) и (g_j, g_i) с противоположными ориентациями.

Распределим ребра G на N классов или цветов, приписав каждому ребру E_{ij} единственный элемент группы

$$c_{ij} = g^{-1}_i g_j \quad (246)$$

Очевидно, в каждой вершине имеется одно входящее ребро и одно выходящее ребро из каждого класса. Таким образом, ребра любого класса образуют однородный граф первой степени; следовательно, он будет суммой конечных или бесконечных ориентированных простых циклов. На рис. 51 изображен цветной граф симметрической группы S_3 ; его вид слегка упрощен, так как оба ориентированных ребра между двумя его вершинами представлены одной линией.

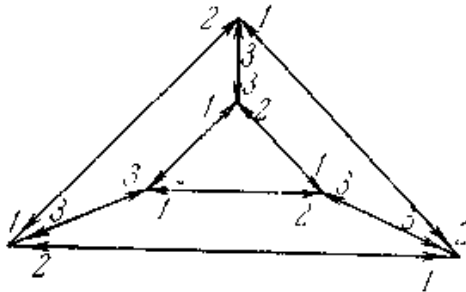


Рис. 51

Пройдем теперь от вершины g_1 в G по ориентированной цепи к некоторой вершине g_n . Тогда по (246) должно быть

$$g_n = g_1 c_{12} c_{13} \dots c_{n-1,n}.$$

Здесь произведение

$$p = c_{12} c_{13} \dots c_{n-1,n}$$

будет единичным элементом e в Γ , если цепь является циклом, и $p \neq e$ в противном случае. Таким образом, цветной граф может оказаться полезным при нахождении соотношений, имеющих место между элементами Γ .

При построении цепей в G на каждом шаге можно выбирать произвольную новую вершину, которая еще не встречалась. Поэтому ясно, что в G можно найти ориентированные гамильтоновы цепи. Чтобы получить некоторую цепь от одной вершины к другой, нет необходимости, чтобы цветной граф содержал ребра всех классов. Достаточно иметь только те, которые соответствуют некоторой системе образующих для Γ , так как остальные элементы получаются как их произведения.

Сказанное приводит к задаче о нахождении этих систем образующих для группы, особенно систем, которые содержат минимальное число элементов или же просты в каком-либо ином смысле, так, чтобы их граф имел гамильтоновы цепи. Некоторые частные случаи были рассмотрены Рапапортом. Между прочим, можно заметить, что задачи такого типа появляются в искусстве колокольного звона, где нужно как-то объединить звонарец, чтобы создать все возможные перестановки на множестве колоколов при помощи переходов от одного к другому в условиях некоторых простых ограничивающих правил.

При применении цветных графов, описанном в следующем пункте, нас будут интересовать те автоморфизмы, которые сохраняют раскраску ребер. Такой автоморфизм получается при умножении каждого элемента из Γ , т. е. каждой вершины из G , слева на фиксированный элемент группы x . Тогда образом ребра $E = (g_i, g_j)$ будут

$$xE = (xg_i, xg_j),$$

и, согласно определению (246), оба эти ребра имеют один цвет. Эти замечания приводят к теореме:

Теорема 1. *Все сохраняющие раскраску ребер автоморфизмы графа Кэли для группы получаются как умножения слева на элементы группы.*

Доказательство. Остается только показать, что все автоморфизмы имеют такой вид. Если какой-нибудь сохраняющий цвета автоморфизм α имеет неподвижную вершину g_i , то он должен также оставлять неизменными все выходящие и входящие ребра в g_i , так как имеется ровно одно такое ребро каждого цвета. Таким образом, все соседние с g_i вершины также инвариантны слева. Продолжая это рассуждение, мы видим, что $\alpha = e$ есть тождественное отображение. Следовательно, все $\alpha \neq e$ должны сдвигать все вершины, и может существовать ровно один такой автоморфизм, переводящий одну вершину в другую. Так как умножения слева обладают этим свойством, только они и являются автоморфизмами, сохраняющими цвета.

2.8.17. Графы с заданными группами

Фрухт получил следующий результат.

Теорема 1. *Для любой конечной группы Γ можно найти такой конечный неориентированный граф G , что группа автоморфизмов графа G изоморфна Γ .*

Доказательство. Первый шаг состоит в построении такого семейства неизоморфных связных графов H_v , что группа каждого H_v является единичной. Это можно сделать многими способами. Фрухт использовал графы, изображенные на рис. 51.

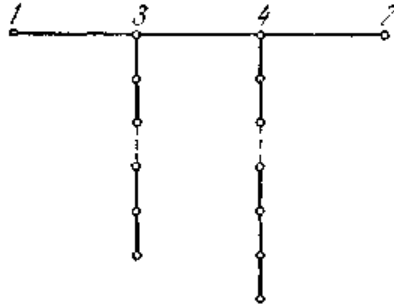


Рис. 51

Такой граф имеет три ребра $(1, 3)$, $(3, 4)$, $(4, 2)$, и от вершин 3 и 4 в H_v , $v = 1, 2, \dots$, отходят простые цепи соответственно длины $2v$ и $2v + 1$. Затем для Γ строится цветной граф Кэли $C(\Gamma)$. Искомый граф G получается из $C(\Gamma)$ такой заменой каждого ориентированного ребра (g_i, g_j) из класса цвета v на граф H_v , что g_i отождествляется с вершиной 1, а g_j — с вершиной 2.

Очевидно, граф G является неориентированным и связным. Остается показать, что его группа изоморфна Γ ; это можно сделать аналогично доказательству теоремы 1 п.2.8.16. Рассмотрим сначала автоморфизм α , оставляющий неподвижной некоторую вершину g_i . В G ребра, отходящие от g_i , имеют вид $(1, 3)$ или $(2, 4)$ в зависимости от направления соответствующего ребра в $C(\Gamma)$. Но если вершина g_i инвариантна при α , то соседние вершины 3 и 4 также должны оставаться неизменными, так как все соответствующие им цепи в H_v имеют различные длины. По той же причине каждый из графов H_v в g_i остается неподвижным; следовательно, такой будет и вершина 2, соответствующая вершине g_j . Последовательно мы получим, что все вершины в G неподвижны, т. е. что α является тождественным отображением.

При автоморфизме графа G любая вершина g_i может переходить только в некоторую вершину g_j , так как все остальные вершины графа H_n имеют другие локальные степени. Кроме того, из предыдущего следует, что для каждой пары (g_i, g_j) может быть не более одного автоморфизма, переводящего одну вершину в другую. Но такой автоморфизм в G индуцируется автоморфизмами графа $C(\Gamma)$, определяемыми умножениями слева на элементы из Γ . Это завершает доказательство теоремы 1.

По существу, проведенное доказательство очень просто, но такая конструкция обычно приводит к графам высокого порядка. Фрухт предложил более экономный в этом отношении метод, построив однородные графы степени 3 с заданной группой. Для любой группы Γ можно найти граф такого вида с $2(n + 2)N$ вершинами, где N есть порядок Γ , а n — число ее образующих.

Требование, чтобы граф имел заданную группу, не налагает существенных ограничений на другие его свойства. Избицкий и Сабидуеси показали, что можно найти однородный граф с произвольными степенью, связностью, хроматическим числом и другими характеристиками. Эти же авторы распространили теорему Фрухта на бесконечные группы.

Группы автоморфизмов частичных упорядочений исследовали Биркгоф и Фрухт; можно показать, что частичное упорядочение P с заданной группой Γ может быть найдено так, чтобы P имело по более N^2 вершин, где N есть порядок группы Γ .

В связи с группами автоморфизмов графов следует также упомянуть о некоторых исследованиях однородных графов степени 3, которые провел Татт. Он рассмотрел задачу о нахождении таких графов, в которых любая ориентированная простая цепь длины s может быть переведена автоморфизмом в любую другую такую простую цепь. Это возможно только при $s \leq 5$. Число s должно также удовлетворять условию

$$s \leq \frac{1}{2} m + 1,$$

где m есть длина кратчайшего простого цикла. Такие специальные графы («клетки»), для которых s принимает это максимальное значение, имеют ряд замечательных свойств. Коксетер доказал, что при $s = 2$ существует бесконечное число однородных графов степени 3 со свойством транзитивности. Пример для случая $s = 1$ был построен Фрухтом.

2.8.18. Реберные отображения

Будем говорить, что два графа G и G' *реберно изоморфны*, если существует такое взаимно однозначное соответствие $E \xleftrightarrow{\quad} E'$ между их ребрами, что если E_1 и E_2 — смежные ребра в G , то соответствующие ребра E'_1 и E'_2 смежны в G' , и наоборот. Ясно, что любой обычный (вершинный) изоморфизм между G и G' определяет также реберный изоморфизм, но обратное, вообще говоря, не имеет места. Этот вопрос был изучен Уитни, и мы приведем его основные результаты в несколько упрощенной форме.

На рис. 53 приведен пример двух графов, которые реберно изоморфны, но не являются вершинно изоморфными.

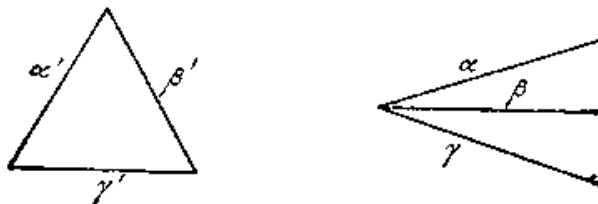


Рис. 53.

На рис. 54 приведены три пары графов порядка 4, которые и вершинно, и реберно изоморфны, но реберный изоморфизм не индуцируется никаким вершинным изоморфизмом.

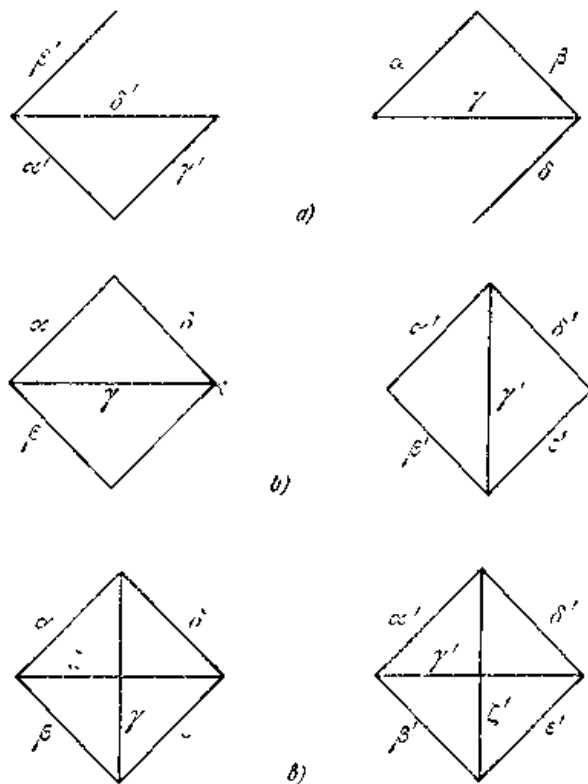


Рис.54

Предоставляем читателю проверить, что для графов, порядок которых не превосходит 4, подобные отображения исчерпываются этими примерами.

Теорема Уитни утверждает следующее:

Теорема 1. Пусть G — конечный связный граф, отличный от графов, изображенных на рис. 53 и 54. Тогда любой реберный изоморфизм графа G на другой граф индуцируется вершинным изоморфизмом.

Доказательство. Согласно предыдущим рассуждениям можно считать, что G имеет не менее пяти вершин. Тогда он содержит некоторую минимальную связную часть, т. е. дерево, с пятью

вершинами. Легко проверить, что для таких графов все реберные изоморфизмы индуцируются вершинными изоморфизмами.

Ощий случай может быть доказан индукцией по числу ребер в G . Предположим, что к G добавляется ребро $E = (a, b)$ так, что новый граф $G \cup E$ связан. Обозначим через

$$E_1^{(a)} = (a, a_1), \quad E_2^{(a)} = (a, a_2), \dots, \quad (247)$$

$$E_1^{(b)} = (b, b_1), \quad E_2^{(b)} = (b, b_2), \dots \quad (248)$$

ребра графа G , инцидентные соответственно a и b . Согласно индуктивному предположению граф, реберно изоморфный $G \cup E$, должен иметь вид $G' \cup E'$, где G' вершинно изоморфен G , а E' есть ребро с такими же смежностями ребер в G' , что и E в G . Поэтому можно считать, что реберно изоморфный граф имеет вид $G \cup E'$, где $E' = (a', b')$.

1. E является концевым ребром, присоединенным в вершине a . Чюбы показать, что $G \cup E$ и $G \cup E'$ вершинно изоморфны, достаточно доказать, что ребро E' также концевое с $a = a'$. Если бы ребро E' не было концевым, то оба его конца a' и b' принадлежали бы G , так что E' было бы смежно некоторым ребрам из G , отличным от (247). Но раз E' концевое, должно быть $a = a'$, так как иначе E' также было бы смежно некоторым ребрам не из (247).

2. E имеет оба конца в G .

(α) $a = a', b \neq b'$. Чюбы доказать, что это невозможно, заметим, что, так как b' должно быть инцидентно всем ребрам (248), может найтись только одно такое ребро

$$E_1^{(b)} = (b, b_1) = (b, b').$$

Но тогда E' в b' смежно некоторому ребру в G , не принадлежащему (247) и (248); такое ребро в b' должно существовать, так как (b, b_1) не может быть изолированным ребром в G .

(β) E и E' не имеют общих концов. E' смежно всем ребрам (247); поэтому может быть не более двух таких ребер

$$E_1^{(a)} = (a, a_1), \quad E_2^{(a)} = (a, a_2), \quad E' = (a_1, a_2).$$

Все ребра (248) также смежны E' . Поэтому может найтись не более двух ребер в b , и они должны соединять b с a_1 и a_2 . В любом случае E' смежно другим ребрам в a_1 и a_2 , так как G связан и имеет более 4 вершин.

Так как нары графов на рис. 54 изоморфны, из теоремы 1 следует

Теорема 2. *Реберно изоморфные графы вершинно изоморфны, за исключением графов на рис. 53.*

Заметим, что два графа реберно изоморфны тогда и только тогда, когда их смежностные графы изоморфны. Таким образом, теорема 2 равносильна следующей теореме.

Теорема 3. *Графы с изоморфными смежностными графами изоморфны, за исключением графов на рис. 53.*

Пусть G — связный граф. Будем говорить, что G *циклически изоморфен* другому графу G' , если существует такое взаимно однозначное реберное отображение между ними, что ребрам, лежащим на простом цикле в одном графе, соответствуют ребра на простом цикле в другом графе. Следующая теорема, по существу, принадлежит Уитни.

Теорема 4. *Пусть граф G связный и без разделяющих вершин. Для того чтобы все циклические изоморфизмы G индуцировались изоморфизмами, необходимо и достаточно, чтобы граф G был 3-вершинно связным.*

Доказательство. Покажем сначала, что если G 3-вершинно связный, то любой циклический изоморфизм индуцируется изоморфизмом. Никакой из исключительных графов на рис. 53 и 54 не является 3-вершинно связным. Поэтому, согласно теореме 1, достаточно, чтобы любой циклический изоморфизм был реберным изоморфизмом. Предположим, что при циклическом изоморфизме двум смежным ребрам

$$E_1 = (a, b), \quad E_2 = (b, c)$$

соответствуют несмежные ребра

$$E'_1 = (a', b'), \quad E'_2 = (c', d').$$

По предположению о связности граф G_1 , полученный из G после удаления ребер, отходящих от b , не может иметь разделяющих вершин. Теорема 3 показывает, что в G_1 существует простой цикл, проходящий через a и c ; следовательно, имеются две непересекающиеся простые цепи $B(a, c)$ и $C(a, c)$, не содержащие E_1 и E_2 . Так как $E_1 \cup E_2 \cup B$ — простой цикл, его образ является простым циклом, проходящим через E'_1 и E'_2 . Изменим обозначения так, чтобы в этом полученном простом цикле a' была связана с c' и b' с d' (рис. 55).

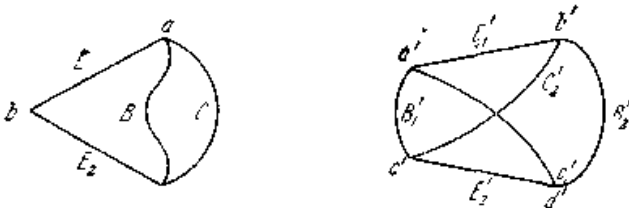


Рис. 55.

$E_1 \cup E_2 \cup C$ также есть простой цикл, которому соответствует простой цикл, проходящий через E'_1 и E'_2 . В этом простом цикле a' должна быть связана с d' раньше, чем с c' (как показано на рисунке), так как непосредственная связь по C' от a' к c' , не содержащая E'_1 и E'_2 , соответствовала бы простому циклу в G , состоящему только из части ребер из B и C . Следовательно, в G' существовал бы простой цикл $E'_1 \cup C'_1 \cup B'_2$. Но прообраз этого простого цикла не является простым циклом, и мы получаем противоречие.

Предположим далее, что вершинная связность G равна $l_0 = 2$. Тогда существует разложение G на два подграфа

$$G = G(A) \cup G(B), \quad A \cap B = \{a, b\}, \quad (249)$$

где a и b — две разделяющие вершины; $G(A)$ и $G(B)$ имеют внутренние вершины. Если существует ребро (a, b) , то отнесем его к $G(A)$. Из разложения (249) мы получим новый граф G^* , поворачивая G относительно вершин a и b , т. е. присоединяя $G(B)$ к $G(A)$ через переставленные разделяющие вершины a и b . Очевидно, граф G^* будет циклически изоморфен G , а ребра из $G(A)$ и $G(B)$ в a и b не останутся смежными.

Уитни доказал, что если $l_0 = 2$, то любой граф, циклически изоморфный графу G , изоморфен некоторому графу G^* , полученному из G последовательными поворотами относительно различными пар разделяющих вершин a и b .

3. Размещение центров и медиан в графе

В практической деятельности постоянно возникают задачи «наилучшего» размещения оборудования (или средств обслуживания) в сетях или графах. В частности, если граф представляет сеть дорог и вершины соответствуют отдельным районам, то можно поставить задачу оптимального размещения больниц, полицейских участков, пожарных частей и многих других крайне необходимых предприятий и служб. В таких случаях критерий оптимальности может состоять в минимизации расстояния (или времени проезда) от пункта обслуживания до самой отдаленной вершины графа, т. е. в оптимизации «наихудшего варианта». В более общей задаче требуется разместить несколько таких пунктов обслуживания (а не только один). При этом самая отдаленная вершина графа должна находиться по крайней мере от одного пункта обслуживания на минимально возможном расстоянии. К таким задачам относятся задачи размещения аварийных

служб, и поэтому объективным требованием здесь является минимизация наибольшего расстояния от произвольной вершины графа до ближайшего к ней пункта обслуживания. По очевидным причинам задачи такого типа называются *минимаксными задачами размещения*. Полученные при решении этих задач места размещения пунктов обслуживания называются *центрами* графа.

В некоторых задачах размещения лучше всего было бы минимизировать сумму всех расстояний от вершин графа до центра обслуживания (если предполагать, что ищется место для размещения только одного такого пункта обслуживания). Такой критерий является наиболее подходящим, например, в задаче о размещении склада в сети дорог, где вершины представляют потребителей, обслуживаемых этим складом, или в задаче размещения телефонных станций в телефонной сети, где вершины представляют абонентов. Задачи такого типа вообще относятся к *минисуммным задачам размещения*, хотя целевая функция является часто не просто суммой расстояний, а суммой различных функций от расстояний. Места размещения пунктов обслуживания, полученные в результате решения минисуммной задачи, называются *медианами* графа.

Целью этой главы является рассмотрение минимаксной задачи размещения для взвешенных графов с весами c_{ij} , соответствующими дугам (и представляющими длины) и другими весами v_p , связанными с вершинами (и представляющими, скажем, размеры или важность объектов). Приводятся алгоритмы определения оптимального размещения центров в таких графах и результаты вычислений для графов средних размеров.

Минисуммная задача размещения рассматривается отдельно. Хотя эти две задачи, очевидно, связаны между собой, но методы их решения различны.

3.1. Разделения

Для любой вершины x_i графа $G = (X, \Gamma)$ пусть $R_\lambda^o(x_i)$ есть множество тех вершин x_j графа G , которые достижимы из вершины x_i с помощью путей со взвешенными длинами $v_j d(x_i, x_j)$, не превосходящими величины λ . Через $R_\lambda^l(x_i)$ будет обозначаться множество тех вершин x_j графа G , из которых вершина x_i может быть достигнута с использованием путей, имеющих взвешенные длины $v_j d(x_j, x_i) \leq \lambda$. Таким образом,

$$R_{\lambda}^o(x_i) = \{x_j \mid v_j d(x_i, x_j) \leq \lambda, x_j \in X\} \quad (1)$$

и

$$R_{\lambda}^i(x_i) = \{x_j \mid v_j d(x_j, x_i) \leq \lambda, x_j \in X\}.$$

Для каждой вершины x_i определим следующие два числа:

$$s_o(x_i) = \max_{x_j \in X} [v_j d(x_i, x_j)]$$

и

$$s_i(x_i) = \max_{x_j \in X} [v_j d(x_j, x_i)]. \quad (2)$$

Числа $s_o(x_i)$ и $s_i(x_i)$ называются соответственно *числом внешнего разделения* и *числом внутреннего разделения вершины x_i* . Следует отметить, что $s_o(x_i)$ является наибольшим числом в строке x_i матрицы $D'(G)$, полученной в результате умножения каждого столбца j матрицы расстояний $D(G)=[d(x_i, x_j)]$ на v_j , и $s_i(x_i)$ является наибольшим числом в столбце x_i матрицы $D''(G)$, полученной после умножения каждой строки j матрицы расстояний $D(G)$ на v_j .

Рассмотрим в качестве примера ориентированный граф, изображенный на рис. 1, и предположим, что все веса вершин и дуг графа равны единице.

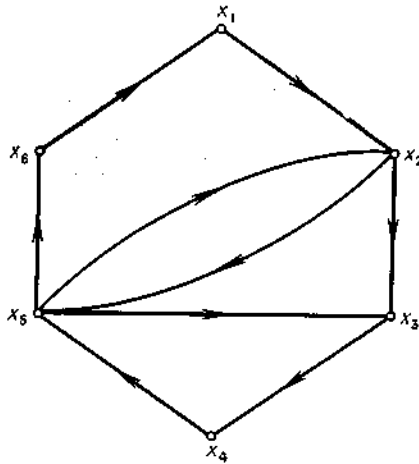


Рис. 1.

Матрица расстояний графа имеет вид

	x_1	x_2	x_3	x_4	x_5	x_6	$s_o(x_i)$
x_1	0	1	2	3	2	3	3
x_2	3	0	1	2	1	2	3
x_3	4	3	0	1	2	3	4
x_4	3	2	2	0	1	2	3
x_5	2	1	1	2	0	1	2*
x_6	1	2	3	4	3	0	4
$s_i(x_i)$	4	3*	3*	4	3*	3*	

Числа внешних и внутренних разделений приведены в присоединенных к матрице столбце и строке соответственно.

Если λ_o — наименьшая длина λ , такая, что для вершины x_i

$$R_\lambda^o(x_i) = X$$

(т. е. все вершины графа G достижимы из x_i с использованием путей, взвешенные длины которых не превосходят λ_o , причем λ_o — наименьшее из таких чисел), то из соотношений (1) и (2) следует равенство

$$s_o(x_i) = \lambda_o. \tag{3}$$

Аналогично, если λ_i — такая наименьшая длина λ , что

$$R_\lambda^i(x_i) = X,$$

то $s_i(x_i) = \lambda_i$.

Совершенно очевидно, что у графа G числа внешнего и внутреннего разделений любой вершины конечны только тогда, когда граф сильно связный, т. е. когда каждая вершина достижима из всякой другой вершины.

3.2. Центр и радиус

Вершина x_o^* , для которой

$$s_o(x_o^*) = \min_{x_i \in X} [s_o(x_i)], \tag{4(a)}$$

называется *внешним центром* графа G ; и аналогично вершина x^* , для которой

$$s_t(x_i^*) = \min_{x_i \in X} [s_t(x_i)], \quad (4(б))$$

называется *внутренним центром* графа G .

У графа может быть несколько (больше, чем один) внешних и внутренних центров. Таким образом они образуют множества внешних и внутренних центров соответственно.

Число внешнего разделения вершины x^* , являющейся внешним центром, называется *внешним радиусом*: $\rho_0 = s_0(x^*)$; число внутреннего разделения внутреннего центра называется *внутренним радиусом*:

$$\rho_t = s_t(x^*_t).$$

У графа, изображенного на рис. 1, с матрицей расстояний, приведенной выше, имеются только один внешний центр (вершина x_5) и четыре внутренних центра, образующих множество $\{x_2, x_3, x_5, x_6\}$. Внешний радиус графа равен 2, а внутренний 3.

3.3. Размещение аварийных служб и пунктов обслуживания

Рассмотрим задачу обслуживания нескольких жилых районов или населенных пунктов (связанных между собой дорожной сетью) каким-либо одним пунктом обслуживания (например, одной больницей, или полицейским участком, или пожарным депо). По некоторым причинам (например, наличие людских ресурсов или другие удобства) пункт обслуживания должен быть размещен в одном из этих районов, а не в произвольной точке какой-либо дороги. Предположим, что «длины» c_{ij} дуг графа G (вершины которого соответствуют районам, а дуги — дорогам) образуют матрицу времен проезда между этими районами. Эта матрица необязательно симметрическая, т.е., вообще говоря, $c_{ij} \neq c_{ji}$, поскольку время проезда может зависеть от уклонов на дороге, наличия улиц с односторонним движением и т. д.

В случае размещения полицейского участка или пожарного депо интересуются временем, необходимым для проезда в наиболее отдаленный из этих районов. Следовательно, задача размещения полицейского участка (или пожарного депо) состоит в минимизации этого времени. Задача становится более реалистичной, если каждой вершине графа приписывается вес (v_j), представляющий вероятность

потребности данного района в соответствующем обслуживании. Эти веса, например, могут быть пропорциональны численности населения каждого района. Вершина, которая минимизирует время проезда до самого отдаленного района, является внешним центром графа.

В случае размещения больницы интересуются временем, необходимым для проезда машины скорой помощи в самый отдаленный район и возвращения ее в больницу. Если определить число внешне-внутреннего разделения вершины x_i с помощью равенства

$$s_{ot}(x_i) = \max_{x_j \in X} \{v_j [d(x_i, x_j) + d(x_j, x_i)]\},$$

то вершину x_{ob}^* на которой достигается минимум выражения $s_{ot}(x_i)$,

можно назвать внешне-внутренним центром.

Для графа, изображенного на рис. 1, с матрицей расстояний $D(G)$, приведенной ранее, внешне-внутренним центром является вершина x_5 . Внешне-внутренний радиус равен 5.

3.3.1. Абсолютный центр

Соотношение (3) определяют числа разделения для любой вершины $x_i \in X$. Мы обобщим теперь это определение на случай «искусственных» точек, которые можно помещать на дугах. Итак, если $a = (x_i, x_j)$ (см. рис. 2) представляет дугу графа с весом (длиной) c_{ij} , то точка y , помещаемая на этой дуге, может быть определена посредством задания длины $l(x_i, y)$ участка (x_i, y) причем должно выполняться равенство

$$l(x_i, y) + l(y, x_j) = c_{ij}. \quad (5)$$

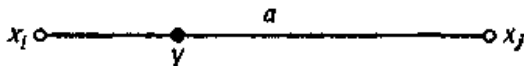


Рис. 2. Размещение на дуге.

Числа разделения $s_o(y)$ и $s_t(y)$ точки y независимо от того, является она вершиной графа G или искусственной точкой дуги графа G , определяются следующим образом:

$$s_o(y) = \max_{x_i \in X} [v_i d(y, x_i)], \quad (6)$$

а для $s_t(y)$ соответствующее выражение получается из соотношения (2).

Точка y^*_o , для которой

$$s_o(y^*_o) = \min_{y \in G} \{s_o(y)\}, \quad (7)$$

называется *абсолютным внешним центром* графа; и аналогично определяется y^*_i — *абсолютный внутренний центр*.

Число внешнего разделения абсолютного внешнего центра называется *абсолютным внешним радиусом*: $r_o = s_o(y^*_o)$, и число внутреннего разделения абсолютного внутреннего центра называется *абсолютным внутренним радиусом*: $r_i = s_i(y^*_i)$.

Пример. Рассмотрим неориентированный граф G , показанный на рис. 3, где все веса вершин и длины ребер равны единице.

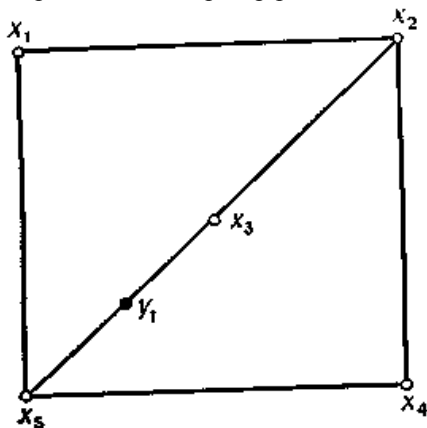


Рис. 3. Абсолютный центр.

Поскольку граф неориентированный, то числа внешних и внутренних разделений одинаковые (для одной и той же вершины). Матрица расстояний графа имеет вид

	x_1	x_2	x_3	x_4	x_5
x_1	0	1	2	2	1
x_2	1	0	1	1	2
$D(G) = x_3$	2	1	0	2	1
x_4	2	1	2	0	1
x_5	1	2	1	1	0

Нетрудно видеть, что центром является каждая вершина и что радиус равен 2.

Если теперь выбрать точку y_1 на ребре (x_3, x_5) так, что $l(x_3, y_1) = \frac{1}{2}$ (и,

следовательно, $l(y_1, x_5) = \frac{1}{2}$), то расстояние от этой точки до вершин графа даются таблицей

	x_1	x_2	x_3	x_4	x_5
y_1	$1\frac{1}{2}$	$1\frac{1}{2}$	$\frac{1}{2}$	$1\frac{1}{2}$	$\frac{1}{2}$

Значит, наибольшее число разделения равно $1\frac{1}{2}$. Таким образом,

точка y_1 более «центральна», чем любая из вершин графа G . В действительности точка y_1 является абсолютным центром графа G — так же, как и все точки, расположенные в середине ребер (x_1, x_2) , (x_2, x_3) , (x_5, x_4) , (x_4, x_2) и (x_5, x_1) . Ни одна из вершин графа не является абсолютным центром. Таким образом, в общем случае может быть один или больше абсолютных центров, которые располагаются или в вершинах, или на дугах графа.

3.4. Алгоритмы нахождения абсолютных центров

Центры и радиусы графа можно найти непосредственно из матрицы взвешенных расстояний, как было показано в разд. 3.1 и 3.2. Приведем два метода нахождения абсолютного центра графа и проиллюстрируем эти методы на примере.

3.4.1. Метод Хакими

Этот метод очень прост и для неориентированного графа состоит в следующем (для ориентированного графа метод остается таким же, надо только каждое «неориентированное» понятие заменить его «ориентированным двойником»).

(i) Для каждого ребра a_k графа найти точки (или точку) y^*_k на a_k , которые имеют наименьшее число деления.

(ii) Из всех точек y^*_k ($k = 1, 2, \dots, m$) в качестве абсолютного центра графа G выбрать точку с наименьшим числом деления.

Первый шаг метода осуществляется следующим образом.

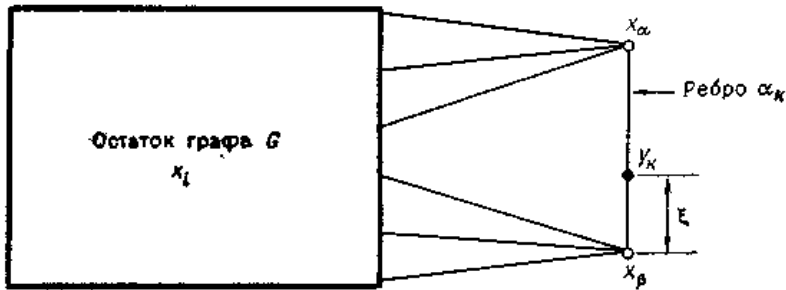


Рис. 4.

Возьмем ребро a_k графа G (см. рис. 5.4). Имеем (из соотношения (6)):

$$\begin{aligned} (y_k) &= \max_{x_i \in X} [v_i d(y_k, x_i)] = \\ &= \max_{x_i \in X} [v_i \min \{l(y_k, x_\beta) + d(x_\beta, x_i), l(y_k, x_\alpha) + d(x_\alpha, x_i)\}], \quad (8) \end{aligned}$$

Поскольку расстояние $d(y_k, x_i)$ равно либо длине маршрута, проходящего через вершину x_α , либо длине маршрута, проходящего через x_β , $l(y_k, x_\alpha)$ и $l(y_k, x_\beta)$ являются длинами соответствующих частей ребра a_k .

Пусть $l(y_k, x_\beta) = l$. Поскольку $l(y_k, x_\alpha) = c_{\alpha\beta} - l(y_k, x_\beta) = c_{\alpha\beta} - l$, то соотношение (8) примет вид

$$s(y_k) = \max_{x_i \in X} \min [v_i \{\xi + d(x_\beta, x_i)\}, v_i \{c_{\alpha\beta} + d(x_\alpha, x_i) - \xi\}]. \quad (9)$$

Для фиксированной вершины x_i и при каждом значении ξ ($0 \leq \xi \leq c_{\alpha\beta}$) можно найти наименьшие значения выражений, заключенных в соотношении (9) в квадратные скобки. Для этого выпишем отдельно два указанных выражения:

$$\begin{aligned} T_i &= v_i \{ \xi + d(x_\beta, x_i) \}, \\ T'_i &= v_i \{ c_{\alpha\beta} + d(x_\alpha, x_i) - \xi \} \end{aligned} \quad (10)$$

и, рассматривая их относительно ξ , строим нижнюю «огibaющую» для соответствующих им прямых линий. (В действительности эта «огibaющая» представляет собой ломаную линию, состоящую из двух «нижних» лучей (от точки пересечения) рассматриваемых прямых линий).

Повторяя эту процедуру для всех вершин $x_i \in X$, мы построим на одном и том же чертеже все остальные нижние «огibaющие». Далее вычертим верхнюю «огibaющую» для всех ранее полученных нижних «огibaющих», которая (в силу соотношения (9)) дает числа разделения $s(y_k)$ для всех значений параметра ξ , т. е. для всех точек y_k ребра a_k . Построенная огibaющая (она составлена из линейных кусков) может иметь несколько минимумов. Точка y_k , соответствующая наименьшему из этих минимумов, является (в силу соотношения (7)) абсолютным центром y^*_k , отвечающим дополнительному ограничению: он должен лежать на ребре a_k . Абсолютным центром графа будет такая точка y^*_k , которой соответствует наименьший из минимумов, определяющих указанные выше абсолютные центры на ребрах a_k ($k = 1, 2, \dots, m$).

3.4.2. Размещение аварийных служб (общий случай)

Рассмотрим еще раз задачу об обслуживании нескольких районов каким-либо одним пунктом обслуживания (одной больницей, или пожарным депо, или полицейским участком). Если опущено ограничение, состоящее в том, что пункт обслуживания должен размещаться в каком-то из жилых районов, то оптимальным размещением пункта обслуживания будет его размещение в любом абсолютном центре соответствующего графа.

Рассмотрим, например, неориентированный граф, приведенный на рис. 5.

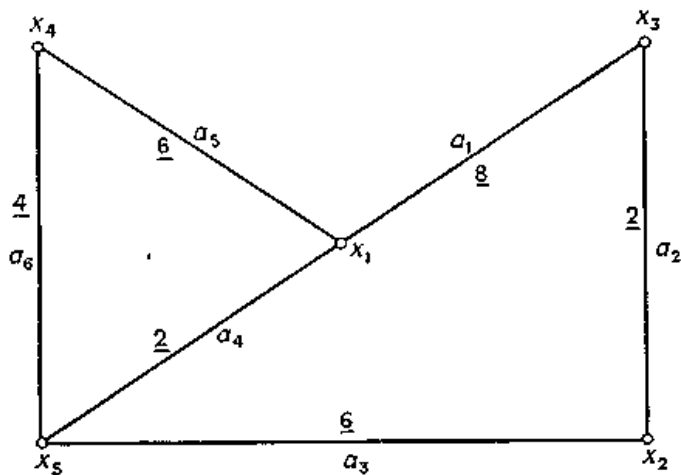


Рис. 5. Граф из примера разд. 3.4.2.

Пусть вершины графа соответствуют жилым районам, а длина ребра (x_i, x_j) равна времени (в минутах), необходимому для проезда из района x_i в район x_j . Предположим, что вершины графа имеют единичные «веса» и матрица «расстояний» (времен) такова:

	x_1	x_2	x_3	x_4	x_5
x_1	0	8	8	6	2
x_2	8	0	2	10	6
$D(G) = x_3$	8	2	0	12	8
x_4	6	10	12	0	4
x_5	2	6	8	4	0

Очевидно, что центром графа является либо вершина x_1 , либо x_5 и что радиус графа равен 8.

Сначала возьмем ребро a_1 (рис. 5), и пусть расстояние ξ измеряется от вершины x_3 до вершины x_1 .

Выражения T_i и T'_i из соотношений (10) (для $i = 1, 2, \dots, 5$) имеют в нашем случае такой вид:

$$T_1 = \xi + d(x_3, x_1) = \xi + 8,$$

$$T'_1 = c_{3,1} + d(x_1, x_1) - \xi = 8 - \xi,$$

аналогично

$$T_2 = \xi + 2, \quad T'_2 = 16 - \xi,$$

$$T_3 = \xi, \quad T'_3 = 16 - \xi,$$

$$T_4 = \xi + 12, \quad T'_4 = 14 - \xi,$$

$$T_5 = \xi + 8, \quad T'_5 = 10 - \xi.$$

Графики этих функций изображены на рис. 6а, для $0 \leq \xi \leq 8$.

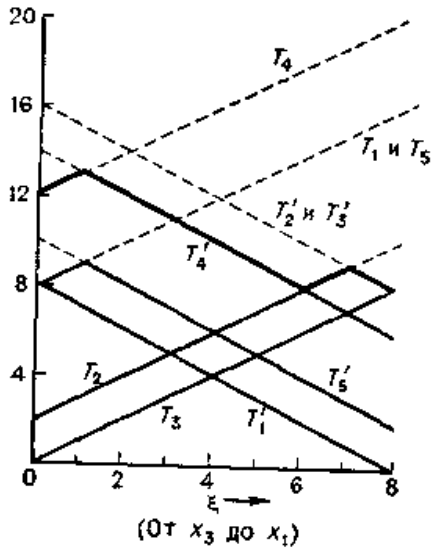


Рис. 6а. Размещение на ребре a_1 .

Имеются два локальных абсолютных центра y^*_1 на расстоянии 6 и 8 минут от x_3 . Число разделения этих двух центров равно 8 минутам. Аналогично строятся графики для ребер a_2, a_3, \dots , см. рис. 6 б—е.

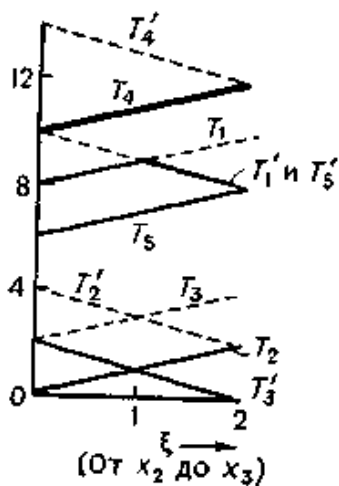


Рис. 6б. Размещение на ребре a_2 .

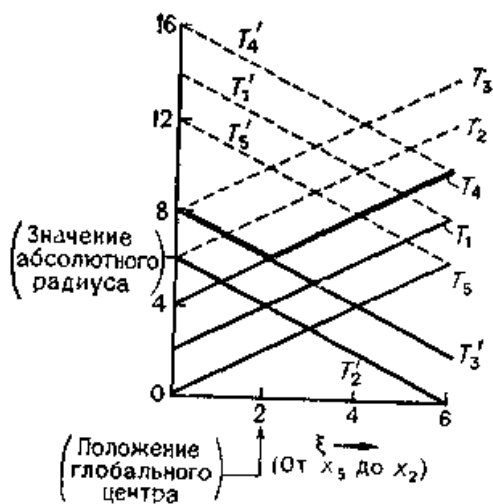


Рис. 6в. Размещение на ребре a_3 .

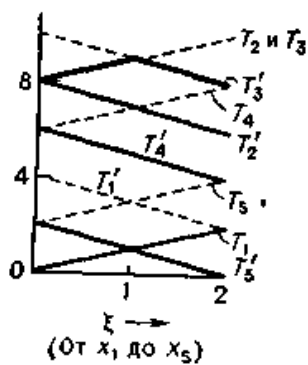


Рис. 6г. Размещение на ребре a_4 .

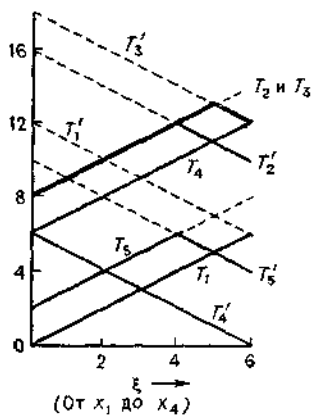


Рис. 6д. Размещение на ребре a_5 .

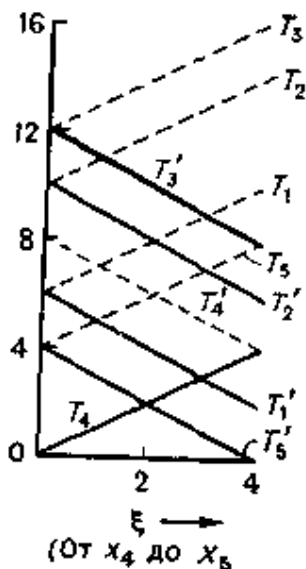


Рис. 6е. Размещение на ребре a_6 .

Абсолютный центр (y^*) этого графа есть такой локальный центр, который имеет наименьшее число разделения. Он, как легко видеть, находится на ребре a_3 (рис. 6в) в двух минутах от x_5 . Абсолютный радиус r (соответствующий y^*), как видно из рис. 6, равен 6.

3.4.3. Модифицированный метод Хакими

В описанном выше методе Хакими поиск локального центра осуществляется вдоль всего ребра графа G . Если в графе много ребер, то время вычисления, требуемое для поиска, может оказаться чрезвычайно большим. Рассматриваемая в данном разделе модификация метода Хакими состоит в вычислении верхних и нижних оценок абсолютных локальных радиусов, соответствующих локальным центрам ребер, и в использовании полученных оценок для уменьшения числа ребер, участвующих в поиске.

Всякому локальному центру, расположенному на ребре (x_i, x_j) , соответствует (как видно из соотношения (8), если положить все длины

(равными нулю) его абсолютный локальный радиус (скажем, r_{ij}), который не меньше, чем p_{ij} , где

$$p_{ij} = \max_{x_s \in X} [v_s \min \{d(x_s, x_i), d(x_s, x_j)\}]. \quad (11)$$

Таким образом, p_{ij} есть *нижняя* оценка абсолютного радиуса графа, если абсолютный центр лежит на ребре (x_i, x_j) . Следовательно, величина

$$P = \max_{(x_i, x_j) \in A} [p_{ij}] \quad (12)$$

(где A —множество ребер графа) является обоснованной нижней оценкой абсолютного радиуса графа.

Допустим, что абсолютный центр расположен в середине ребра (x_i, x_j) .

Тогда, в силу соотношения (8), абсолютный радиус равен

$p_{ij} + \frac{1}{2} v_{s^*} c_{ij}$, где v_{s^*} — вес той вершины x_{s^*} , в которой достигается наибольшее значение величины p_{ij} , указанной в соотношении (11).

Следовательно,

$$H = \min_{(x_i, x_j) \in A} [p_{ij} + \frac{1}{2} v_{s^*} c_{ij}] \quad (13)$$

является обоснованной *верхней* оценкой абсолютного радиуса графа.

Таким образом, всякое ребро (x_i, x_j) , для которого $p_{i,j_0} \geq H$, может не рассматриваться при поиске абсолютного центра.

Для графа, приведенного на рис. 5, у которого $v_{ii} = 1$ для всех $x_i \in X$, получаем (используя формулу (11)) следующие результаты:

центр на $a_1 = (x_3, x_1)$ дает $p_{3,1} = \max(2, 6, 2) = 6$,

центр на $a_2 = (x_2, x_3)$ дает $p_{2,3} = \max(8, 10, 6) = 10$,

центр на $a_3 = (x_5, x_2)$ дает $p_{5,2} = \max(2, 2, 4) = 4$,

центр на $a_4 = (x_1, x_5)$ дает $p_{1,5} = \max(6, 8, 4) = 8$,

центр на $a_5 = (x_1, x_4)$ дает $p_{1,4} = \max(8, 8, 2) = 8$,

центр на $a_6 = (x_4, x_5)$ дает $p_{4,5} = \max(2, 6, 8) = 8$.

Верхняя оценка H равна, следовательно (в силу соотношения (13)),

$$\min(6 + 4, 10 + 1, 4 + 3, 8 + 1, 8 + 3, 8 + 2) = 7,$$

и поэтому ребра a_2, a_4, a_5 и a_6 , для которых $p_{ij} > 7$, могут быть исключены из списка кандидатов на размещение абсолютного центра.

Поиск абсолютного центра нужно вести только на двух оставшихся ребрах (так, как это было продемонстрировано раньше и показано на рис. 6 (а) и 6 (в)). Используя верхние и нижние оценки, удалось уменьшить поиск в 3 раза. Наименьшая нижняя оценка P , вычисленная по формуле (12), равна $P = \min(6, 10, 4, 8, 8, 8) = 4$.

Эта оценка не является точной, поскольку истинная величина абсолютного радиуса r данного графа (найденная в разд. 3.4.2) равна 6.

3.4.4. Итерационный метод

Чтобы сделать изложение более простым, мы опишем данный метод для случая неориентированных графов. Замена каждого «неориентированного» понятия его «ориентированным двойником» не приводит к какому-либо коренному изменению метода.

Пусть $Q_\lambda(x_i)$ — множество всех таких точек y , лежащих на ребрах графа G , из которых вершина x_i достижима со взвешенным расстоянием, не превосходящим λ . Таким образом,

$$Q_\lambda(x_i) = \{y \mid v_i d(y, x_i) \leq \lambda, y \text{ — точка графа } G\}. \quad (14)$$

Это выражение похоже на те (см. (1)), с помощью которых были определены множества $R_\lambda^o(x_i)$ и $R_\lambda^t(x_i)$; отличие состоит в том, что теперь y — любая точка графа G , а необязательно его вершина. Абсолютный радиус r , очевидно, является наименьшим значением λ , при котором из некоторой точки y графа G все вершины графа могут быть достигнуты со взвешенным расстоянием, меньшим или равным λ . Следовательно, r есть такое наименьшее значение λ (скажем, λ_{\min}), что

$$\bigcap_{x_i \in X} [Q_\lambda(x_i)] \equiv Q_\lambda(x_1) \cap Q_\lambda(x_2) \cap \dots \cap Q_\lambda(x_n) \neq \emptyset. \quad (15)$$

Поэтому можно начать с произвольного небольшого значения λ , строить множества $Q_\lambda(x_i)$ для всех $i = 1, 2, \dots, n$ и проверять, выполняется ли соотношение (15). Если оно не выполняется, то надо увеличить немного величину λ , заново построить множества $Q_\lambda(x_i)$ (при новом значении λ) и опять проверить, не выполняется ли соотношение (15). Эту процедуру можно повторять до тех пор, пока не будет выполняться (15). Полученная таким образом величина λ принимается за абсолютный радиус r графа G . Более того, поскольку приращения величины λ малы, то пересечение

$$\bigcap_{x_i \in X} [Q_\lambda(x_i)] \quad (16)$$

при завершении процесса итерации будет содержать только одну точку (этого для практических целей достаточно), и она является как раз абсолютным центром u^* . (Возможно, что будет не одна точка, если существует более чем один абсолютный центр.)

Поскольку $d(x_i, x_j)$ есть наикратчайшее расстояние между двумя произвольными вершинами x_i и x_j , то совершенно очевидно, что если λ меньше половины взвешенного расстояния между x_i и x_j , т. е.

$$\lambda \leq \frac{v_i v_j}{v_i + v_j} d(x_i, x_j), \quad \text{то} \quad Q_\lambda(x_i) \cap Q_\lambda(x_j) = \emptyset$$

и полное пересечение множеств в выражении (16) пусто.

Следовательно, итерационный процесс поиска абсолютного центра можно начать со значения λ , равного

$$\lambda_0 = \max_{x_j, i, j \in X} \left[\frac{v_i v_j}{v_i + v_j} d(x_i, x_j) \right], \quad (17)$$

поскольку радиус r должен быть не меньше λ_0 . С этой точки зрения, значение $\delta = 2\lambda_0$ можно назвать диаметром графа. Нужно, однако, отметить, что совсем необязательно диаметр графа в два раза больше значения абсолютного радиуса.

Детальное описание более общего алгоритма, частью которого является рассмотренный здесь алгоритм, будет дано позже.

3.5. Кратные центры (p -центры)

Понятие центра графа допускает следующее обобщение: можно, рассматривать не отдельную точку (центр), а множество из p точек, которые образуют кратный центр (p -центр).

Пусть X_p — подмножество (содержащее p вершин) множества X вершин графа $G = (X, \Gamma)$. Через $d(X_p, x_i)$ будем обозначать наикратчайшее из расстояний между вершинами множества X_p и вершиной x_i , т. е.

$$d(X_p, x_i) = \min_{x_j \in X_p} [d(x_j, x_i)].$$

Аналогично, символом $d(x_i, X_p)$ обозначается

$$\min_{x_j \in X_p} [d(x_i, x_j)].$$

Подобно тому, как определяются числа разделения вершин, мы можем определить числа разделения для множеств вершин:

$$s_o(X_p) = \max_{x_j \in X} [v_j d(X_p, x_j)]$$

$$s_i(X_p) = \max_{x_j \in X} [v_j d(x_j, X_p)],$$

(18)

где $s_o(X_p)$ и $s_i(X_p)$ — числа внешнего и внутреннего разделения множества X_p .

Множество X_{po}^* , для которого

$$s_o(X_{po}^*) = \min_{X_p \subseteq X} [s_o(X_p)], \quad (19)$$

называется p -кратным внешним центром графа G ; аналогично определяется p -кратный внутренний центр $X_{p_i}^*$.

Ранее указывалось, что центры графа легко могут быть получены из матрицы взвешенных расстояний. Однако находить таким же способом (полным перебором) p -центр можно лишь для небольших графов и для небольших значений величины p . При таком подходе надо построить всевозможные множества вершин $X_p \subseteq X$, содержащие p вершин, а затем, используя формулы (18) и (19), непосредственно найти множества $X_{p_o}^*$ и $X_{p_i}^*$ образующие p -центры. Если предположить, что матрица расстояний уже известна, то непосредственное применение соотношений (18) и (19) потребует выполнить

$$p \cdot (n - p) \cdot \binom{n}{p}$$

сравнений. Это число при $n = 100$ и $p = 5$ равно $3,58 \cdot 10^{10}$.

3.5.1. Задача размещения нескольких пунктов обслуживания

В разд. 3.2.1 была рассмотрена задача размещения одной больницы (или одного полицейского участка, или одного пожарного депо) в графе, представляющем реальную сеть дорог. Однако очень часто имеет место такая ситуация, когда одного пункта обслуживания недостаточно, поскольку он не в состоянии обслужить все поступающие вызовы, и тогда возникает задача о наилучшем размещении нескольких таких пунктов обслуживания. Эту задачу можно сформулировать так.

Найти наименьшее число пожарных депо (например) и такое их размещение, чтобы расстояние от каждого жилого района до ближайшего к нему пожарного депо не превышало наперед заданной величины. Если же число пожарных депо известно, то требуется разместить их так, чтобы было минимально возможным расстояние от любого района до ближайшего к нему депо.

Если предположить, что пожарные должны размещаться в вершинах соответствующего графа G , то задача будет состоять в нахождении p -центров графа для $p = 1, 2, 3, \dots$ и т. д. до тех пор, пока число деления p -центра не станет меньше или равно заданному расстоянию. Полученное (последнее) значение числа p будет наименьшим числом пожарных депо, а p -центр — их оптимальным размещением, удовлетворяющим предъявляемым требованиям.

Алгоритм нахождения p -центров является частным случаем алгоритма решения более общей задачи, состоящей в определении

p -центров, располагающихся, вообще говоря, не в вершинах графа. Рассмотрение общего алгоритма поиска p -центров мы отложим, пока не будет исследована соответствующая более общая задача.

3.6. Абсолютные p -центры

Если ограничение, согласно которому точки p -центра должны лежать в вершинах графа, снято и допускается размещение точек на дугах, то получающееся при этом (более общее) множество из p точек называется *абсолютным p -центром*. Таким образом, тот объект, который в разд. 3.4 был назван абсолютным центром, в соответствии с настоящей терминологией можно назвать абсолютным p -центром.

Задача нахождения абсолютного p -центра может быть сформулирована следующим образом.

(а) Найти оптимальное размещение в любых точках графа заданного числа (например, p) центров при условии, что расстояние (время) до самой отдаленной вершины от ближайшего к ней центра является минимально возможным.

Вторая задача, очень близкая к задаче (а) и которая, как будет показано позже, может быть решена тем же методом, что и задача (а), состоит в следующем.

(б) Для заданного «критического» расстояния найти такое наименьшее число центров и такое их размещение, чтобы все вершины графа лежали в пределах этого критического расстояния (по крайней мере каждая вершина — от ближайшего к ней центра).

Это — общая задача определения абсолютных p -центров. Именно она наиболее часто встречается на практике. Однако решать ее гораздо труднее, чем какой-либо из ее «ограниченных» вариантов. Метод Хакими, приведенный в разд. 3.4 и предназначенный для решения задачи с одним абсолютным центром, не может быть обобщен на случай абсолютных p -центров. Для нахождения таких центров Сингер предложил некоторый эвристический метод.

В данном разделе мы познакомимся с итерационным алгоритмом решения задачи об абсолютных p -центрах графа. Из приведенных далее результатов вычислений видно, что этот алгоритм является быстро сходящимся. Метод обладает следующими двумя преимуществами:

(i) Процесс можно закончить сразу же, как только достигнута необходимая «точность» в расположении центров.

(ii) Метод легко видоизменить таким образом, чтобы можно было находить решения, близкие к оптимальному, и, следовательно, проводить анализ устойчивости решения.

Ради упрощения обозначений мы будем рассматривать только неориентированные графы. Распространение полученных результатов на ориентированные графы осуществляется очевидным образом.

Пусть Y_p — произвольное множество каких-либо p точек на графе G . Число разделения $s(Y_p)$ множества Y_p определяется так:

$$s(Y_p) = \min_{x_j \in X} \{v_j d(Y_p, x_j)\}, \quad (20)$$

где

$$d(Y_p, x_j) = \min_{y_i \in Y_p} [d(y_i, x_j)].$$

Абсолютный p -центр графа G определяется как множество точек Y_p^* , для которого

$$s(Y_p^*) = \min_{Y_p \text{ на } G} [s(Y_p)]. \quad (21)$$

3.7. Алгоритм нахождения абсолютных p -центров

Рассмотрим по очереди каждую вершину x_i и «углубимся» по всем возможным маршрутам, выходящим из нее, на расстояние $\delta_i = \lambda/v_i$, где λ — заданная константа, которую мы будем называть *константой «проникновения»*.

Пусть $Q_\lambda(x_i)$ — множество всех точек y на графе G , из которых вершина x_i достижима в пределах расстояния δ_i при заданном значении λ . Множества $Q_\lambda(x_i)$ определяются с помощью соотношения (14). Эти множества весьма легко можно построить, применяя алгоритм, подобный алгоритму Дейкстры (позволяющему находить кратчайшие пути в графе).

Определим *область* Φ_λ как множество таких точек y на графе G , что из каждой точки y достижимо в пределах расстояния (в дальнейшем (в тех случаях, когда не могут возникнуть недоразумения) слова «в пределах расстояния... при заданном λ » опускаются) δ_i (при заданном λ) одно и то же множество вершин графа G . Область может быть, например, частью ребра или может содержать только одну точку. Рассмотрим в качестве примера граф, изображенный на рис. 7. Пусть все веса вершин равны единице, а длины ребер равны $c_{1,2} = 4$, $c_{2,3} = 8$ и $c_{3,1} = 6$. Возьмем $\lambda = 3$. Тогда $\delta_i = 3$ для всякого i и существует шесть различных областей.

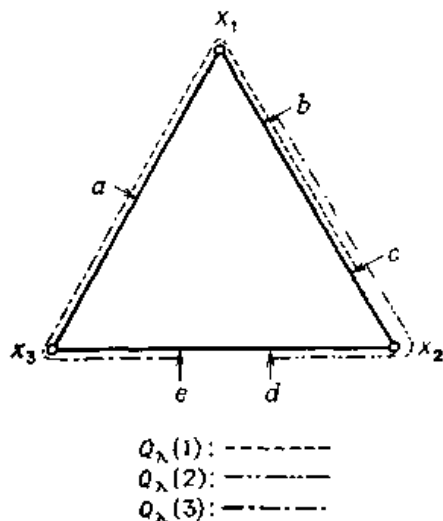


Рис. 7. Области, образованные при $\lambda = 3$.

Область 1. Точка a (единственная точка, из которой в пределах расстояния, равного 3, достижимы вершины x_1 и x_3).

Область 2. Отрезок ребра (x_1, x_2) от b до c . (Из любой точки этой области достижимы вершины x_1 и x_2 .)

Область 3. Отрезки (a, x_1) и (x_1, b) соответствующих ребер (из точек этой области достижима только вершина x_1).

Область 4. Отрезки (a, x_3) и (x_3, e) соответствующих ребер (из каждой точки этой области достижима только x_3).

Область 5. Отрезки (c, x_2) и (x_2, d) (из точек этой области достижима только x_2).

Область 6. Отрезок ребра (x_2, x_3) от e до d (из точек этой области не достижима никакая вершина).

В общем случае все области Φ_λ можно следующим образом построить из достижимых множеств Q_λ .

Области, из которых не достижимы никакие вершины, описываются соотношением

$$\Phi_\lambda(0) = \{y \mid y \text{ на } G\} - \bigcup_i Q_\lambda(x_i), \quad (22)$$

где второй член исключает все области графа G , из которых можно достигнуть хотя бы одну вершину x_i .

(Всюду в этих рассуждениях достижимость берется «в пределах заданного расстояния δ_i ».)

Области, из которых можно достигнуть ровно t вершин $x_{i_1}, x_{i_2}, \dots, x_{i_t}$ (для любого $t = 1, \dots, n$), определяются следующим выражением:

$$\Phi_\lambda(x_{i_1}, x_{i_2}, \dots, x_{i_t}) = \bigcap_{q=1, \dots, t} Q_\lambda(x_{i_q}) - \{[\bigcap_{q=1, \dots, t} Q_\lambda(x_{i_q})] \cap [\bigcup_{q=t+1, \dots, n} Q_\lambda(x_{i_q})]\}, \quad (23)$$

где второй член исключает такие области, из которых достижимы вершины $x_{i_1}, x_{i_2}, \dots, x_{i_t}$, и еще хотя бы одна из оставшихся вершин графа.

С первого взгляда кажется, что число областей из соотношения (23) может быть очень велико. Однако на практике пересечения множеств, входящие в выражение (23), становятся пустыми уже для небольших значений t . Это приводит к вполне обозримому множеству областей. Эффективные (с вычислительной точки зрения) методы построения областей, а также дальнейшего уменьшения числа необходимых областей приводятся в настоящем разделе несколько позже.

3.7.1. Описание алгоритма

Алгоритм построения абсолютного p -центра графа $G = (X, A)$ при заданном p выглядит следующим образом.

Шаг 1. Положить $\lambda = 0$.

Шаг 2. Увеличить λ на небольшую величину $\Delta \lambda$.

Шаг 3. Построить множества $Q_\lambda(x_i)$ для всех $x_i \in X_n$ и найти области Φ_λ .

Шаг 4. Образовать двудольный граф $G' = (X' \cup X, A')$, где X' — множество вершин, каждая из которых соответствует некоторой области Φ_λ , и A' — множество дуг, такое, что дуга между областью-вершиной и вершиной x_i существует тогда и только тогда, когда x_i может быть достигнута из этой области.

Шаг 5. Найти наименьшее доминирующее множество графа G' .

Шаг 6. Если число областей в приведенном выше множестве больше, чем p , то вернуться к шагу 2; в противном случае остановиться. Области этого множества образуют абсолютный p -центр исходного графа G .

Следует отметить, что число областей в наименьшем доминирующем множестве представляет (по определению) наименьшее число точек в G , из которых достигаются все вершины графа в пределах расстояния

проникновения λ , используемого в данной итерации. Надо также отметить, что в процессе построения абсолютного p -центра по указанному выше алгоритму можно заодно получить абсолютные $(n - 1)$ -, $(n - 2)$ - и т. д. центры. Таким образом, если в конце некоторой итерации (продолжающейся с шага 2 до шага 6) число областей в наименьшем доминирующем множестве уменьшается с некоторого уровня l до $l - 1$, то области этого нового множества образуют абсолютный $(l - 1)$ -центр, а величина λ , взятая для этой итерации, будет «абсолютным $(l - 1)$ -радиусом», т. е. она является «критическим» значением λ — при меньших значениях λ не существуют $(l - 1)$ -центры, из которых в пределах взвешенного расстояния λ достижима каждая вершина графа.

Если нужно найти такое наименьшее значение p , что каждая вершина достижима из некоторого центра в пределах заданного критического расстояния, то шаги с 3 по 6 в приведенном выше алгоритме следует выполнять при λ , равном этому «критическому» значению. Соответствующее число областей в наименьшем доминирующем множестве является тогда требуемым значением для p , и области этого множества образуют искомый p -центр.

3.7.2. Вычислительные аспекты

Предположим, что λ зафиксировано и вычислены расстояния $\delta_i = \lambda v_i$. Любое ребро графа либо достижимо целиком, либо частично, либо совсем не достижимо из вершины x_i в пределах расстояния δ_i . Если достижима только часть ребра (от какого-либо конца ребра до некоторой «предельной» точки на нем), то над предельной точкой ставится «метка». Эти метки содержат всю информацию, необходимую для описания множеств $Q_\lambda(x_i)$. Таким образом, $Q_\lambda(x_i)$ состоит из точек всех ребер (или частей ребер), принадлежащих кратчайшим маршрутам между метками и вершиной x_i .

После размещения всех меток (для всех вершин) каждое ребро будет разделено на ряд участков; каждый участок характеризуется теми вершинами, которые из него могут быть достигнуты (то есть из точек этого участка). Таким образом, любой участок можно описать бинарным (единица-ноль) вектором $\{j_1, j_2, \dots, j_n\}$ длины n , в котором $j_k = 1$, если вершина x_k достижима из этого участка, и $j_k = 0$ в противном случае.

Совокупность всех участков с одинаковым бинарным вектором образует область, и, следовательно, области Φ_λ , задаваемые соотношениями (23), могут быть построены с помощью таких меток. Поскольку область достижима только из тех вершин, для которых

$j_k = 1$ (в бинарном векторном представлении этой области), и не из каких других, то в дальнейшем эти бинарные векторы мы будем называть *строгими пересечениями* (SI).

Представление области бинарным вектором не содержит никакой информации о месте ее расположения на графе. Однако такое представление выгодно с вычислительной точки зрения: оно не предъявляет больших требований к памяти машины и позволяет уменьшить время вычисления, особенно при выполнении шага 5 алгоритма. Если абсолютный p -центр описан «на языке строгих пересечений», то дальнейшая процедура осуществляется чрезвычайно просто: надо лишь выяснить, какие участки ребер соответствуют этим строгим пересечениям.

На шаге 4 алгоритма надо строить двудольный граф $G' = (X' \cup X, A')$, у которого вершины из X' представляют области Φ_λ . Это может привести к графу больших размеров, что сильно увеличит время вычисления на шаге 5. Однако с помощью приводимой ниже теоремы удастся уменьшить размеры графа G' : исключаются те области, которые не влияют на получаемое оптимальное решение (но если существует больше одного оптимального решения, то при такой процедуре некоторые из них можно потерять).

Теорема 1. *При заданной величине X для получения некоторого минимального доминирующего множества графа G' можно предварительно исключить из множества X' все вершины, соответствующие тем SI, над которыми доминируют другие SI в X' . Мы говорим, что $(SI)_1$ доминирует над $(SI)_2$, если $(SI)_1 \otimes (SI)_2 = (SI)_2$, где \otimes означает булевское произведение.*

Доказательство этой теоремы очевидно.

В алгоритме из предыдущего раздела параметр λ при каждой очередной итерации увеличивается (по понятным причинам) лишь на небольшую величину. Существует много других более эффективных способов варьирования параметра λ . Результаты вычислений, приведенные в разд. 3.7.4, получены с помощью программы, использующей направленный двоичный поиск по λ .

3.7.3. Пример

Рассмотрим граф G , изображенный на рис. 8.

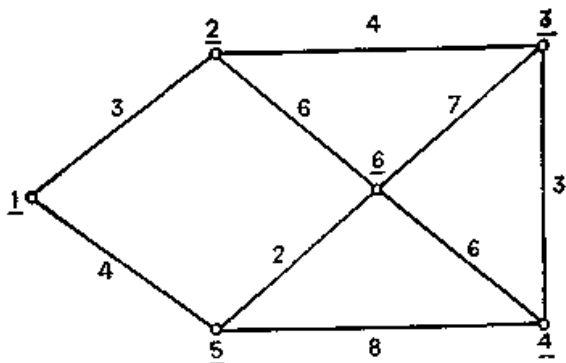


Рис. 8. Граф из примера 3.7.3.

Номера вершин подчеркнуты.

Все другие числа являются длинами ребер.

Число, стоящее около любого ребра, задает его длину; вес каждой вершины графа равен единице. Требуется найти абсолютный p -центр с минимально возможной величиной параметра p и такой, чтобы каждая вершина графа отстояла хотя бы от одного из этих p центров на расстоянии, не превосходящем 3,5 единиц.

Для $\lambda = 3,5$ на рис. 9 показаны метки, определяющие множества $Q_\lambda(x_i)$.

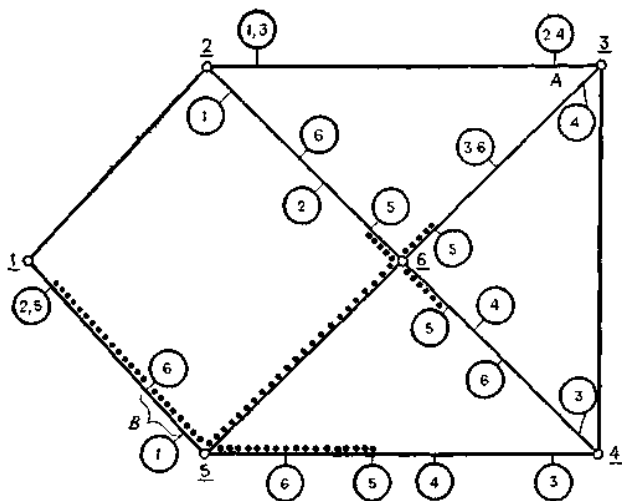


Рис. 9. Метки для достижимых множеств. Множество $Q_\lambda(5)$ отмечено точками.

Они расставлялись сразу же в процессе последовательного «прохождения» ребер. Числа в кружках, стоящие около произвольной метки, указывают «номера» вершин, которым эта метка соответствует. Так, кружок (1,3) на ребре (2,3) означает, что только отмеченная точка достижима из вершин 1 и 3 в пределах расстояния 3,5. На рис. 9 точками около ребер показано достижимое множество $Q_\lambda(5)$, соответствующее вершине 5. Всего в данном примере существует 33 участка ребер, включая пустой участок (из которого ни одна вершина не может достигаться в пределах расстояния 3,5 единиц). Пустой участок расположен между метками 4 и 5 на ребре (4,5). Некоторые из этих 33 участков имеют одинаковые SI; существует всего 18 областей со следующими SI

(1) 000000	(7) 000011	(13) 000101
(2) 010000	(8) 110000	(14) 001001
(3) 001000	(9) 001100	(15) 111000
(4) 000100	(10) 100010	(16) 011100
(5) 000010	(11) 011000	(17) 110010
(6) 000001	(12) 010001	(18) 100011

Так, например, участок между метками 1 и 6 на ребре (1, 5) принадлежит области с SI, равным 100011, поскольку лишь из точек графа, лежащих между этими двумя метками, можно достигнуть вершины 1, 5 и 6 (и никакие другие) в пределах расстояния $\lambda = 3,5$.

После исключения тех SI, которые доминируются другими SI, остается только 7 областей (с номерами SI от 12 до 18). Граф G' после выполнения шага 4 алгоритма выглядит так, как показано на рис. 10.

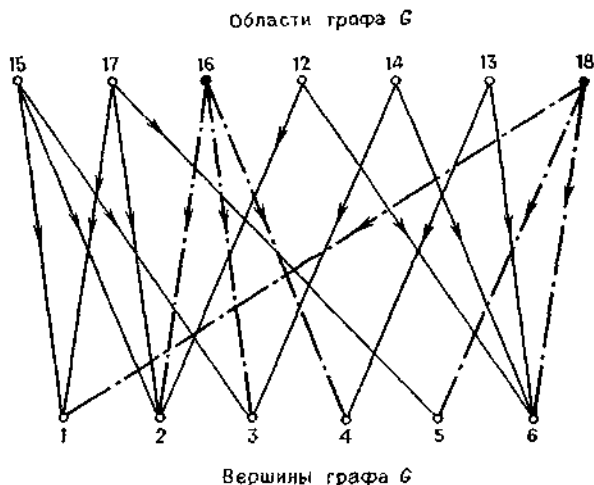


Рис. 10. Граф G' из примера 3.7.3.

- Области в наименьшем доминирующем множестве.

Для нахождения наименьшего доминирующего множества этого графа можно воспользоваться алгоритмом, предназначенным для решения задачи о наименьшем покрытии. Однако в нашем (простом) примере наименьшее множество легко строится с помощью прямого последовательного перебора; оно «порождается» SI с номерами 16 и 18 в приведенном выше списке. Область, соответствующая SI с номером 16, состоит из одной точки, расположенной на ребре (2,3) в том месте, где находится кружок (2,4). На рис. 9 эта точка обозначена буквой A . Область, соответствующая SI с номером 18, представляет собой участок ребра (1,5) между метками 6 и 1; на рис. 9 она обозначена буквой B .

Таким образом, при $\lambda = 3,5$ требуется два центра; один расположен в точке A , а другой — в любой точке области B . Поскольку область A является точкой, то два указанных центра образуют также абсолютный 2-центр и $\lambda = 3,5$ является минимально возможным критическим расстоянием для существования 2-центра. Поэтому при $\lambda < 3,5$ область A исчезнет совсем, и тогда уже нужно строить 3-центр.

3.7.4. Результаты вычислений

Алгоритм построения абсолютного p -центра был опробован на ЭВМ. Каждый бинарный вектор, задающий SI, хранится в одном слове, так что можно использовать булевские функции для более удобного выполнения теоретико-множественных операций.

В некоторых задачах, в частности, когда число областей в искомом абсолютном p -центре (т. е. число p) велико, шаг 5 алгоритма, описанного в разд. 3.7.1, требует значительную часть всего времени, затрачиваемого на решение задачи. В табл. 1 время, необходимое для выполнения этого шага, приведено отдельно от полного времени решения задачи. В этой таблице представлены времена вычисления (в секундах) и числа итераций, требуемые при построении абсолютных 1-, 2-, 3-, 4- и 5-центров для некоторых графов. Здесь мы под числом итераций понимаем число повторений шагов алгоритма с 3 по 6 (для различных значений λ), всех тех повторений, которые необходимо осуществить для получения абсолютного p -центра с заданной точностью. Графы, которые были использованы при составлении табл. 1, выбирались случайным образом и являются связными и неориентированными. Требуемая точность во всех случаях принималась равной 1 % от средней длины ребер рассматриваемого графа.

Как видно из табл. 1, алгоритм, приведенный в разд. 3.7.1, можно весьма эффективно использовать при нахождении абсолютных p -центров достаточно больших графов.

Таблица 1 Вычислительные результаты для выборочных графов

Граф		Число центров в абсолютном p -центре (значение p)														
		5			4			3			2			1		
n	m	A	B	C	A	B	C	A	B	C	A	B	C	A	B	C
10	23	0,44	0,08	7	0,33	0,06	7	0,15	0,04	3	0,30	0,04	6	0,34	---	7
10	30	0,46	0,06	6	0,35	0,05	6	0,46	0,09	7	0,55	0,08	7	0,51	---	7
20	33	14,00	13,00	8	1,36	0,45	7	1,16	0,11	8	1,13	0,05	8	0,86	---	6
20	40	3,51	1,94	7	0,37	0,22	1	1,49	0,43	7	1,73	0,06	6	1,05	---	5
20	60	6,02	4,50	7	2,58	1,06	7	1,29	0,18	5	1,63	0,05	6	1,75	---	7
20	80	16,10	12,00	9	3,60	1,70	8	2,70	0,41	6	3,20	0,10	8	2,90	---	8
20	100	---	---	---	4,25	1,53	6	2,85	0,37	6	5,41	0,19	9	2,20	---	4
20	120	---	---	---	6,38	2,86	6	4,21	0,58	7	4,35	0,09	6	3,32	---	4
30	40	---	---	---	6,00	2,10	11	3,82	0,45	10	3,83	0,05	10	5,00	---	12
30	60	---	---	---	18,10	14,50	7	2,66	0,12	5	3,16	0,06	6	2,22	---	4
30	80	---	---	---	8,40	3,73	6	7,20	1,56	9	6,90	0,17	10	3,55	---	5
30	100	---	---	---	23,20	18,80	6	6,50	1,20	6	7,10	0,25	8	6,60	---	8
40	60	---	---	---	---	---	---	11,80	2,53	10	11,20	0,22	13	12,40	---	14
40	80	---	---	---	---	---	---	27,00	13,50	11	10,00	0,51	8	7,40	---	6
50	80	31,34	25,58	12	24,55	16,60	13	17,70	0,95	12	8,11	0,11	6	17,80	---	12

↑ Число вершин

↓ Число ребер

A Полное время вычисления (секунды)

B Время вычисления для шага 5 алгоритма

C Число итераций

3.7.5. Применение общего алгоритма для поиска p -центров

Алгоритм из разд. 3.7.1 можно, очевидно, использовать при решении более узкой задачи, приведенной в разд. 3.7.6, т. е. при нахождении p -центра. Для этого надо только изменить шаг 6 настоящего алгоритма (см. разд. 3.7.1) таким образом, чтобы «контролировалось» не число областей в наименьшем доминирующем множестве, найденном на шаге 5, а число вершин графа G , содержащихся в этих областях. Если полученное число вершин больше p , то нужно возвратиться к шагу 2. В противном случае алгоритм заканчивает работу, и p -центром графа будет множество вершин, содержащихся во всех тех областях, которые образуют доминирующее множество.

Поскольку в этом случае необходимы только вершины, содержащиеся в областях, задаваемых соотношениями (22) и (23), то алгоритм из разд. 3.7.1 можно без изменений применять для нахождения p -центров, если области на шаге 3 строить с помощью соотношений (22) и (23), используя вместо множеств $Q_\lambda(x_i)$ множества $R_\lambda(x_i)$. Более того, если все $n(n-1)$ расстояний в матрице расстояний с самого начала расположить в строку в порядке неубывания $[f_1, f_2, \dots, f_{n(n-1)}]$, то на шаге 2 алгоритма значения для параметра λ , должны выбираться лишь из этой строки, и их не надо увеличивать на произвольно малую величину в каждой итерации. Поиск начинается с $1 = f_1$, а затем продолжается при $\lambda = f_2, f_3$ и т. д. в соответствии с указанным выше упорядочением. Следует заметить, что здесь, очевидно, было бы весьма кстати бинарное представление семейства $\{f_1, f_2, \dots, f_{n(n-1)}\}$. (В семействе (в отличие от множества) могут встречаться одинаковые элементы)

В ряде задач о размещении пунктов обслуживания требуется так расположить пункт обслуживания на графе, чтобы сумма кратчайших расстояний от этого пункта до вершин графа была минимально возможной. Оптимальное в указанном смысле место расположения пункта называется *медианой* графа. Исходя из природы целевой функции, такие задачи называют *минисуммными задачами размещения*. Эти задачи в различных формах часто встречаются на практике: при выборе места расположения коммутаторов в телефонной сети, подстанций в электросетях, баз снабжения (складов) в сети дорог (где вершины представляют потребителей) или отделов сортировки в почтовой связи.

С этим типом задач контрастирует *минимаксная задача размещения*, возникающая при выборе места для таких пунктов обслуживания, как

пожарное депо, полицейский участок или амбулатория, и рассмотренная ранее. Ниже рассматривается минисуммная задача размещения. В частности, обсуждается задача о нахождении *p*-медианы данного графа *G*; это задача об оптимальном размещении заданного числа (скажем, *p*) пунктов обслуживания, при котором сумма кратчайших расстояний от вершин графа *G* до ближайших к ним пунктов принимает минимально возможное значение.

(Суммирование ведется по всем вершинам. Для каждой вершины берется кратчайшее расстояние от нее до ближайшего к ней пункта обслуживания.)

Задача нахождения *p*-медианы может быть несколько обобщена, если каждой вершине x_i сопоставить некоторый вес v_i (представляющий, например, ее размеры или важность); тогда целевой функцией, подлежащей минимизации, станет сумма «взвешенных» расстояний.

3.8. Медиана графа

Пусть дан граф $G = (X, \Gamma)$. Для каждой вершины $x_i \in X$ определим два числа, которые назовем *передаточными* числами:

$$\left. \begin{aligned} \sigma_o(x_i) &= \sum_{x_j \in X} v_j d(x_i, x_j), \\ \sigma_i(x_i) &= \sum_{x_j \in X} v_j d(x_j, x_i), \end{aligned} \right\} \quad (24)$$

где $d(x_i, x_j)$ — кратчайшее расстояние от вершины x_i до вершины x_j . Числа $\sigma_o(x_i)$ и $\sigma_i(x_i)$ называются соответственно *внешним* и *внутренним передаточными числами* вершины x_i . Число $\sigma_o(x_i)$ есть сумма элементов строки x_i матрицы, полученной после умножения каждого столбца матрицы расстояний $D(G) = [d(x_i, x_j)]$ на вес соответствующей этому столбцу вершины, т. е. j -й столбец умножается на v_j ; число $\sigma_i(x_i)$ есть сумма элементов столбца x_i матрицы, полученной в результате умножения каждой строки матрицы расстояний $D(G)$ на соответствующий этой строке вес (j -я строка умножается на v_j).

Вершина \bar{x}_o , для которой

$$\sigma_o(\bar{x}_o) = \min_{x_i \in X} [\sigma_o(x_i)], \quad (25)$$

называется *внешней медианой* графа *G*, а вершина \bar{x}_i , для которой

$$\sigma_i(\bar{x}_i) = \min_{x_i \in X} [\sigma_i(x_i)], \quad (26)$$

называется *внутренней медианой* графа G .

Рассмотрим еще раз граф, изображенный на рис. 1 (для которого все v_i и c_{ij} приняты равными единице), и вычислим внешние и внутренние передаточные числа вершин. Эти числа приведены в присоединенных к матрице расстояний строке и столбце:

	x_1	x_2	x_3	x_4	x_5	x_6	$\sigma_o(x_i)$
$D(G) =$	0	1	2	3	2	3	↓ 11
x_2	3	0	1	2	1	2	9
x_3	4	3	0	1	2	3	13
x_4	3	2	2	0	1	2	10
x_5	2	1	1	2	0	1	7*
x_6	1	2	3	4	3	0	13
$\sigma_i(x_i) \rightarrow$	13	9*	9*	12	9*	11	

По полученным передаточным числам видно, что внешней медианой является x_5 с $\sigma_o(x_5) = 7$ и что существуют три внутренние медианы (x_2, x_3 и x_5), каждая с внутренним передаточным числом, равным 9.

3.8.1. Выбор места для склада

Рассмотрим задачу снабжения ряда потребителей товарами, поступающими с одного склада. Потребителей можно объединить в группы таким образом, чтобы каждую группу обслуживало *целое число грузовиков*. Машина выезжает со склада, обслуживает некоторую группу потребителей и возвращается на склад. Группы потребителей можно представить вершинами графа, а сеть дорог — его дугами. На практике каждой группе потребителей присваивается вес v_i , представляющий ее «важность» (например, v_i может быть числом, пропорциональным годовому потреблению или частоте, с которой транспорт должен объезжать эту группу потребителей, чтобы удовлетворить их потребности).

В этом случае задача состоит в определении такого места для склада, чтобы общее расстояние, проходимое транспортом, было бы минимально возможным. Если матрица расстояний $D(G)$ задает действительные расстояния в километрах, то требуемым местом

расположения склада будет такая вершина $\bar{x}_{o,t}$, для которой сумма внешних и внутренних передаточных чисел будет наименьшей. Вершина $\bar{x}_{o,t}$ может быть названа *внешне-внутренней медианой* и найдена из соотношения, аналогичного (24). В следующем разделе будет показано, что для любой точки (вершины или произвольной точки дороги), выбранной для размещения склада, общий километраж, покрываемый транспортом, не может быть меньше, чем для вершины $\bar{x}_{o,t}$.

3.9. Кратные медианы (p -медианы) графа

Ранее, обобщая понятие центра, мы ввели понятие p -центра. Подобным же образом можно обобщить понятие медианы, определив p -медиану.

Пусть X_p — подмножество множества вершин X графа $G=(X, \Gamma)$, и предположим, что X_p содержит p вершин. Как и прежде, введем следующие обозначения:

$$d(X_p, x_j) = \min_{x_i \in X_p} [d(x_i, x_j)] \quad (27 \text{ а})$$

$$d(x_j, X_p) = \min_{x_i \in X_p} [d(x_j, x_i)]. \quad (27 \text{ б})$$

Если x'_i — вершина из X_p , на которой достигается минимум в (27 а) или (27 б), то говорят, что вершина x_j *прикреплена* к x'_i . Передаточные числа множества вершин X_p определяются так же, как и для одиночной вершины:

$$\left. \begin{aligned} \sigma_o(X_p) &= \sum_{x_j \in X} v_j d(X_p, x_j), \\ \sigma_i(X_p) &= \sum_{x_j \in X} v_j d(x_j, X_p), \end{aligned} \right\} \quad (28)$$

где $\sigma_o(X_p)$ и $\sigma_i(X_p)$ — соответственно внешнее в внутреннее передаточные числа множества вершин X_p .

Множество $\bar{X}_{p,o}$, для которого

$$\sigma_o(\bar{X}_{p,o}) = \min_{X_p \subseteq X} [\sigma_o(X_p)], \quad (29)$$

называют *внешней p -медианой* графа G ; аналогично определяется *внутренняя p -медиана*.

Как и в случае p -центров, рассмотренных ранее, даже для графов средних размеров с вычислительной точки зрения нецелесообразно использовать при нахождении p -медиан непосредственно выражения (27), (28) и (29). Алгоритмы построения p -медиан будут даны в разд. 3.11.

3.9.1. Абсолютные p -медианы

С целью упрощения изложения рассмотрим неориентированный граф G . Индексы o и t будут отсутствовать. Разберем сначала случай медианы (1-медианы). Спрашивается, существует ли такая точка y на некотором ребре (не обязательно совпадающая с вершиной графа G), для которой передаточное число

$$\sigma(y) = \sum_{x_j \in X} v_j d(y, x_j)$$

меньше, чем для медианы графа G . Точку \bar{y} , на которой достигается минимум величины $\sigma(y)$, будем называть *абсолютной медианой* графа G .

Сейчас мы докажем, что не существует точки \bar{y} , для которой $\sigma(\bar{y}) < \sigma(\bar{x})$, т. е. здесь ситуация противоположна той, которая имела место при рассмотрении центров.

Теорема 1. *Какова бы ни была точка y графа $G = (X, A)$, в нем найдется по крайней мере одна вершина x , для которой $\sigma(x) \leq \sigma(y)$.*

Доказательство. Пусть y — точка ребра (x_a, x_b) , расположенная на расстоянии ξ от x_a . Тогда

$$d(y, x_j) = \min [\xi + d(x_a, x_j), c_{ab} - \xi + d(x_b, x_j)], \quad (30)$$

где c_{ab} — длина ребра (x_a, x_b) .

Пусть X_a — множество тех вершин x_j , для которых первый член в (30) не больше второго, а X_b — множество вершин, для которых второй член меньше первого. Мы можем тогда написать

$$\begin{aligned} \sigma(y) = \sum_{x_j \in X} v_j d(y, x_j) &= \sum_{x_j \in X_a} v_j [\xi + d(x_a, x_j)] + \\ &+ \sum_{x_j \in X_b} v_j [c_{ab} - \xi + d(x_b, x_j)]. \end{aligned} \quad (31)$$

Поскольку из неравенства треугольника следует, что

$$d(x_a, x_j) \leq c_{ab} + d(x_b, x_j), \quad (32)$$

то, заменяя $c_{ab} + d(x_b, x_j)$ на $d(x_a, x_j)$ в выражении (31), получаем

$$\sigma(y) \geq \sum_{x_j \in X_a} v_j [\xi + d(x_a, x_j)] + \sum_{x_j \in X_b} v_j [d(x_a, x_j) - \xi]. \quad (33)$$

Так как $X_a \cup X_b = X$, то, сделав перегруппировку в (33), имеем:

$$\sigma(y) \geq \sum_{x_j \in X} v_j d(x_a, x_j) + \xi \left[\sum_{x_j \in X_a} v_j - \sum_{x_j \in X_b} v_j \right]. \quad (34)$$

Поскольку для каждого ребра (x_a, x_b) мы вправе сами решать, какую вершину называть x_a и какую x_b , то всегда можно добиться выполнения неравенства

$$\sum_{x_j \in X_a} v_j \geq \sum_{x_j \in X_b} v_j.$$

Заметив, что первый член в правой части неравенства (34) равен $\sigma(x_a)$, получаем из (34) такое соотношение:

$$\sigma(y) \geq \sigma(x_a).$$

Таким образом, для вершины x_a величина $\sigma(x_a)$ не превышает $\sigma(y)$ и, следовательно, теорема доказана.

Теорема 1 довольно просто обобщается на случай p -медиан.

Теорема 2. *Каково бы ни было множество Y_p , состоящее из p точек графа $G = (X, A)$, т. е. из p точек ребер и реришин, найдется по крайней мере одно подмножество $X_p \subset X$, содержащее p вершин, для которого $\sigma(X_p) \leq \sigma(Y_p)$.*

В теоремах 1 и 2 предполагалось, что передаточные числа $\sigma(x)$ и $\sigma(X_p)$ определены с помощью выражений (24) и (28). Леви, Голдман и Мейерса показали, что эти теоремы остаются в силе и в тех случаях, когда передаточные числа определяются как суммы произвольных, вогнутых функций от взвешенных расстояний.

Из теорем 1 и 2 следует, что понятие абсолютной медианы не представляет особого интереса (в противоположность ситуации с абсолютными центрами). Поэтому в остальной части этой главы основное внимание уделяется задаче о p -медиане.

3.10. Обобщенная p -медиана графа

Задача нахождения p -медианы графа является центральной в общем классе задач, встречающихся в литературе под названием «распределение и размещение центров обслуживания» или «размещение складов». Эти задачи являются до некоторой степени более общими, чем задача о p -медиане, в которой вершинам

сопоставлены фиксированные стоимости f_i . *Обобщенная задача о p -медиане* может быть сформулирована следующим образом.

Для заданного графа $G = (X, A)$ с матрицей кратчайших расстояний $[d(x_i, x_j)]$, с весами вершин v_i и с фиксированными стоимостями вершин f_i задача состоит в нахождении такого подмножества \bar{X}_p из p вершин, для которого величина

$$z = \sum_{x_i \in \bar{X}_p} f_i + \sigma(\bar{X}_p) \quad (35)$$

принимает минимально возможное значение.

Таким образом, в этом случае минимизации подлежит не просто передаточное число $\sigma(\bar{X}_p)$ множества \bar{X}_p , а полная функция цели,

содержащая фиксированные стоимости f_i вершин x_i подмножества \bar{X}_p .

На практике f_i представляет, например, фиксированную стоимость строительства пункта обслуживания (склада, фабрики, подстанции) при размещении его в вершине x_i . Задача о p -медиане соответствует такому случаю, когда все f_i одинаковы (например, равны f), так что первый член в (35) равен постоянной величине $\bar{p}f$ независимо от выбора подмножества \bar{X}_p .

Очень близкой к сформулированной выше задаче об обобщенной p -медиане является такая задача, в которой число вершин $|\bar{X}_p|$ не обязательно равно p , а может быть некоторым числом, не превосходящим p . Задача минимизации выражения (35) при условии $|\bar{X}_p| \leq p$ является разновидностью задачи об обобщенной p -медиане и часто встречается на практике.

В задачах размещения складов неизменно приходится сталкиваться с ограничениями, которых нет в «чистой» задаче о p -медиане. Чаще всего встречаются ограничения на наибольшее и наименьшее значения, которые может принять выражение

$$x_j, \text{ прикрепленные к } x_i \quad (36)$$

для любой медианной вершины $x_i \in \bar{X}_p$. Выражение (36) определяет количество товара, вывозимого из вершины x_i , и, следовательно, является мерой вместимости склада x_i .

Очевидно, что такое же обобщение, возникающее при введении ограничений на величину (36), возможно и в задачах о нахождении p -центров и абсолютных p -центров, рассмотренных ранее. Однако для

практических задач размещения энергетических объектов, именно тех задач, которые связаны с нахождением p -центров, указанное обобщение имеет небольшое значение, так как всегда эти ограничения легко вводятся в алгоритм, приведенный ранее.

Сложность решения практических задач размещения складов определяется не изменениями и дополнительными ограничениями, рассмотренными выше, а свойственна природе самой задачи о p -медиане. В свете этого рассмотрим несколько методов, пригодных для определения p -медианы графа. Для упрощения изложения индексы i и t , указывающие связь объектов соответственно с внешними и внутренними медианами, будут опускаться, поскольку излагаемые методы применимы к обеим задачам.

3.11. Методы решения задачи о p -медиане

3.11.1. Формулировка задачи в терминах целочисленного программирования

Пусть $[\xi_{ij}]$ — матрица распределения, в которой

$$\xi_{ij} = \begin{cases} 1, & \text{если вершина } x_j \text{ прикреплена к вершине } x_i, \\ 0 & \text{в противном случае.} \end{cases}$$

Далее примем $\xi_{ii} = 1$, если вершина x_i является медианной вершиной и $\xi_{ij} = 0$ в противном случае. Задача о p -медиане может быть сформулирована тогда следующим образом.

Минимизировать функцию

$$z = \sum_{i=1}^n \sum_{j=1}^n d_{ij} \xi_{ij} \quad (37)$$

при ограничениях

$$\sum_{i=1}^n \xi_{ij} = 1 \quad \text{для } j = 1, \dots, n, \quad (38)$$

$$\sum_{i=1}^n \xi_{ii} = p, \quad (39)$$

$$\xi_{ij} \leq \xi_{ii} \text{ для всех } i, j = 1, \dots, n \quad (40)$$

$$\xi_{ij} = 0 \text{ или } 1, \quad (41)$$

где $[d_{ij}]$ — матрица взвешенных расстояний графа (она получается из матрицы расстояний после умножения каждого j -го столбца на соответствующий вес v_j). Соотношения (38) гарантируют

выполнимость следующего условия: любая заданная вершина x_j прикрепена к одной и только к одной медианной вершине x_i . Выражение (39) гарантирует существование в точности p медианных вершин. Из ограничений (40) следует, что $\xi_{ij} = 1$ только тогда, когда $\xi_{ji} = 1$, т. е. прикрепления осуществляются только к вершинам медианного множества. Если $[\bar{\xi}_{ij}]$ является оптимальным решением задачи, определяемой соотношениями (37)—(41), то p -медиана имеет вид

$$\bar{X}_p = \{x_i \mid \bar{\xi}_{ii} = 1\}.$$

Если ограничения (41) вышеприведенной задачи заменить на

$$\xi_{ij} > 0, \tag{42}$$

то возникает задача линейного программирования (ЛП). Ее решение получить нетрудно. (Отметим, что для величины ξ_{ij} указывать верхнюю границу не нужно, поскольку из соотношений (38) следует, что $\xi_{ij} \leq 1$.)

Решение задачи ЛП не обязательно будет целочисленным, ξ_{ij} может принимать и дробные значения. Ревель и Свэйи показали, что дробные значения встречаются крайне редко, и поэтому в большинстве случаев для получения p -медиан можно использовать язык и методы линейного программирования. В том случае, когда некоторые ξ_{ij} являются дробными, решение можно получить с помощью древовидного поиска (имеется в виду «поиск, использующий дерево решений»), причем одной ветви, соответствующей какой-либо дробной величине (переменной) ξ_{ij} , придается значение 0, а другой ветви — значение 1. Затем для каждой из полученных ветвей (подзадач) можно продолжить поиск решения (как задачи ЛП) и действовать подобным образом до тех пор, пока все ξ_{ij} не примут значения из множества $\{0, 1\}$. Другой подход, отличный от метода линейного программирования, предложен Марстеном, который показал, что решение $[\bar{\xi}_{ij}]$ задачи (37) — (41), соответствующее p -медиане графа, является экстремальной точкой некоторого многогранника H , причем многогранник будет одним и тем же для всех p , удовлетворяющих условию $1 \leq p \leq n$. Используя множители Лагранжа и параметрическое линейное программирование, Марстен предложил метод «прокладки пути» между несколькими экстремальными точками многогранника H . При «прокладке» этого пути последовательно, в порядке убывания p , порождаются некоторые p -медианы графа. Кое-какие p -медианы (для отдельных значений p) могут быть не построены, и, наоборот, могут быть «порождены» такие экстремальные точки многогранника H , которые не соответствуют p -

медианам графа G , так как содержат дробные значения величин ξ_{ij} . Хотя этот метод привлекателен как с теоретической, так и с вычислительной точек зрения, но все же он не всегда приводит к построению p -медианы графа для требуемого значения p .

3.11.2. Алгоритм направленного древовидного поиска

Вместо того чтобы задачу нахождения p -медианы толковать как задачу целочисленного программирования, можно для ее решения использовать прямое дерево поиска. Этот подход лучше всего соответствует структуре рассматриваемой задачи. В настоящем разделе будет описан один из таких методов, в котором каждая подзадача, возникающая при ветвлении в каком-либо узле дерева, определяется тем, что (при заданной вершине x_j) переменная ξ_{ij} полагается равной 1 для некоторой вершины x_i . Равенство $\xi_{ij} = 1$ означает, что вершина x_j прикреплена к вершине x_i , а также, очевидно, что x_i является медианной вершиной.

Этот поиск удобно выполнять следующим образом.

Построим матрицу $M = [m_{kj}]$, j -й столбец которой содержит все вершины графа G , расположенные в порядке неубывания их расстояний от вершины x_j . Таким образом, если $m_{kj} = x_i$, то вершина x_i будет k -й ближайшей к x_j вершиной. Очевидно, что первой ближайшей к x_j вершиной является она сама, т. е. $m_{1j} = x_j$.

Поиск начинается с последовательного просмотра всех вершин графа, с x_1 по x_n . Вершина x_j вначале прикрепляется к вершине m_{1j} , затем к m_{2j} и т. д., пока не будут перебраны все возможности. Необходимо сделать несколько замечаний.

1. Поскольку оптимальное решение состоит из p медианных вершин, то прикрепление вершины в этом решении к какой-либо медианной вершине должно быть наилучшим из p возможных прикреплений. Иными словами, для каждой вершины должно существовать по крайней мере еще $p - 1$ возможностей прикрепления с меньшей стоимостью, чем у выбранной возможности. Следовательно, из матрицы M можно удалить $p - 1$ последних строк, и это не отразится на оптимальном решении задачи о p -медиане

2. Предположим, что вершину x_j прикрепили к вершине m_{kj} (которая является, например, вершиной x_i). Пусть вершина x_i будет k -й ближайшей вершиной к некоторой еще не прикрепленной вершине $x_{j'}$ ($j' < j \leq n$), соответствующей j -му столбцу матрицы M , т. е. $m_{ki} = x_i$. Тогда, очевидно, все элементы m_{ij} с $l > k$ могут быть исключены из дальнейшего рассмотрения (отмечены), поскольку, прикрепив $x_{j'}$ к x_i , мы тем самым причисляем x_i к медианным вершинам и поскольку

вершина x_j может быть прикреплена к x_i с меньшей стоимостью, чем к любой другой вершине m_{ij} при $l = k + 1, k + 2, \dots$. Ясно также, что если на некотором шаге возвращения процедуры поиска изменится распределение вершины x_j относительно x_i , то элементы m_{ij} (неисключенные из рассмотрения, неотмеченные) должны быть рассмотрены заново.

3. Пусть вершина $x_{j'}$ прикреплена к вершине $m_{k'j'}$. Тогда можно предположить, что все вершины $m_{1j'}, m_{2j'}, \dots, m_{(k-1)j'}$ не являются медианными вершинами, поскольку в противном случае вершина $x_{j'}$ была бы расположена по крайней мере от одной из них не дальше, чем от $m_{k'j'}$ (с меньшей) результирующей стоимостью). Эти вершины, следовательно, могут быть отмечены (исключены из рассмотрения) во всех столбцах с номерами $j' > j'$. Нужно иметь в виду, что вершины эти исключаются временно и их надо «восстанавливать» всякий раз при изменении распределения вершины $x_{j'}$ относительно $m_{k'j'}$.

4. Если из рассмотрения исключены t верхних элементов j -го столбца, соответствующего нераспределенной (неприкрепленной) вершине x_j , и $(t + 1)$ -й элемент, т. е. $m_{(t+1)j}$ является медианной вершиной, то x_j должна быть прикреплена к вершине $m_{(t+1)j}$. В этом случае нет необходимости рассматривать какие-либо другие возможности до тех пор, пока некоторые из t верхних элементов не окажутся «восстановленными» (вследствие изменения в предыдущих распределениях вершин, что отразится на множестве «исключенных» вершин). Этот вывод является прямым следствием вышеприведенных замечаний 2 и 3.

5. Если на некотором этапе q проводимого поиска будет построено множество из p медианных вершин (в результате применения «процедуры прикрепления» вершин), то каждую оставшуюся нераспределенную вершину можно прикрепить к ближайшей медианной вершине. Очевидно, что это является оптимальным завершением частного решения, соответствующего осуществленному прикреплению вершин. На следующем шаге процедуры — шаге возвращения — должно изменяться распределение вершин, полученное в конце q -го этапа.

Вычисление нижней границы. Замечания 1—5 можно использовать для ограничения размера дерева поиска путем уменьшения числа возможных прикреплений вершины x_j на любом этапе процедуры. Кроме того, на некоторых этапах, когда последней распределяемой вершиной является вершина x_j , для ограничения дальнейшего поиска может быть использована нижняя граница величины полного оптимального решения (полученного для уже осуществленных прикреплений).

Предположим, что выполненные прикрепления (включая прикрепление вершины x_j) дали p' ($p' < p$) медианных вершин. Тогда оставшиеся прикрепления должны привести к нахождению остальных $p - p'$ медианных вершин. Пусть J — множество индексов еще не распределенных вершин. В общем случае J есть множество индексов j , для которых $j' < j \leq n$, за исключением индексов тех вершин, распределение которых индуцировано распределением первых j' вершин $x_1, x_2, \dots, x_{j'}$ (в соответствии с замечанием 4).

Пусть $m_{\alpha_j j}$ и $m_{\beta_j j}$ — самый верхний и следующий за ним элементы j -го столбца, которые являются неотмеченными (не исключенными из рассмотрения). Тогда наилучшим распределением вершины x_j будет ее прикрепление к вершине $m_{\alpha_j j}$. Если число различных вершин $m_{\alpha_j j}$ для $j \in J$ равно h и $h = p - p'$, то все эти наилучшие распределения еще не прикрепленных вершин являются осуществимыми (т. е. получается множество, содержащее p медианных вершин). Эти распределения образуют оптимальное пополнение частного решения, полученного при распределении вершин $x_1, x_2, \dots, x_{j'}$. В таком случае записывается результат и шаг возвращения может быть предпринят с текущего частного решения.

Если, однако, $h > p - p'$, то для получения всех p медианных вершин надо заменить по крайней мере $h - p + p'$ лучших назначений на вторые лучшие или худшие. Тогда наименьшая дополнительная стоимость распределения является суммой $h - p + p'$ наименьших разностей

$$v_j [d(x_j, m_{\beta_j j}) - d(x_j, m_{\alpha_j j})] \tag{43}$$

по всем вершинам $x_j, j \in J$.

Нижняя граница стоимости оптимального решения, даваемая текущим частным решением, получается прибавлением к сумме стоимостей уже выполненных распределений

$$\sum_{j \in J} v_j d(x_j, m_{\alpha_j j})$$

суммы $h - p + p'$ наименьших разностей, задаваемых выражением (43). Может случиться, что h меньше $p - p'$. Тогда наилучшее пополнение текущего частного решения дает медианных вершин меньше чем p . Но поскольку очевидно, что передаточное число $\sigma(\bar{X}_p)$ оптимальной p -медианы \bar{X}_p монотонно убывает с увеличением p , то при $h < p - p'$ текущее частное решение наверняка не является частью оптимального p -медианного решения и тогда можно выполнить шаг возвращения.

3.11.3. Другой алгоритм направленного поиска

Рассмотрение альтернативных распределений, возникающих при выборе конкретных значений для переменных ξ_{ij} , приводит в процедуре поиска из разд. 3.10 к частным подзадачам. Укажем еще одну процедуру поиска; она состоит в построении бинарного дерева поиска, причем это построение осуществляется путем последовательной проверки условия: является или нет данная вершина медианной вершиной? Процедура поиска в этом случае задается множествами S^+ , S^- и F . Первое множество соответствует тем вершинам, которые в текущий момент являются медианными, второе соответствует немедианным вершинам, а третье — вершинам, о которых пока нельзя сказать ничего определенного.

Нижняя граница стоимости оптимального решения, определяемая уже осуществленным распределением некоторых вершин по множествам S^+ или S^- , может быть вычислена способом, аналогичным приведенному в предыдущем разделе.

Итак, вершина $x_j \in S^-$ должна быть прикреплена к некоторой вершине множества $S^+ \cup F$. Наилучшее из этих распределений имеет стоимость

$$u'_j = \min_{x_i \in S^+ \cup F} [v_j d(x_j, x_i)] \quad (44)$$

С другой стороны, вершина $x_j \in F$, которая не может стать медианной вершиной, должна обладать стоимостью, равной по крайней мере

$$u''_j = \min_{\substack{i \neq j \\ x_i \in S^+ \cup F}} [v_j d(x_j, x_i)]. \quad (45)$$

Нижняя граница вычисляется по формуле

$$\sum_{x_j \in S^-} u'_j + U'', \quad (46)$$

где U'' есть сумма $n - p - |S^-|$ наименьших чисел u''_j . Следует отметить, что $n - p - |S^-|$ есть число вершин, которые пока еще свободны, но могут быть медианными вершинами окончательного решения и, следовательно, должны быть сопоставлены другим вершинам.

Оценка из выражения (46) используется при древовидном поиске специального типа (равностоймостного). При таком поиске ветвление производится в узлах дерева, соответствующих наименьшей нижней границе решения. Методы поиска с приоритетом по глубине,

аналогичные тем, которые рассматривались в начале настоящего раздела и в которых используются оценки, подобные приведенной в выражении (46), — несколько модифицированные, чтобы можно было решать обобщенные p -медианные задачи, возникающие на практике при размещении складов (пунктов обслуживания), — встречаются также в ряде работ.

3.11.4. Приближенный алгоритм

Тэйтц и Барт предложили эвристический метод для нахождения p -медианы. Метод состоит в следующем: случайным образом выбираются p вершин, они и образуют начальное множество S , аппроксимирующее p -медианное множество \bar{X}_p . Затем выясняется, может ли некоторая вершина $x_j \in X - S$ заменить вершину $x_i \in S$ (как медианная вершина), для чего строится новое множество $S' = (S \setminus \{x_i\}) \cup \{x_j\}$ и сравниваются передаточные числа $\sigma(S')$ и $\sigma(S)$. Если $\sigma(S') < \sigma(S)$, то вершина x_i замещается вершиной x_j и из множества S получается множество S' , которое лучше аппроксимирует p -медианное множество \bar{X}_p . Затем исследуется уже множество S' , аналогично тому как исследовалось S , и т. д., пока не будет построено множество \bar{S} , такое, что ни одну его вершину нельзя заместить вершиной из $X - \bar{S}$ и получить при этом множество с меньшим передаточным числом, чем $\sigma(\bar{S})$. Множество \bar{S} берется в качестве требуемого приближения к множеству \bar{X}_p ,

Описание алгоритма

Шаг 1. Выбрать некоторое множество S из p вершин в качестве начального приближения к p -медиане» Назовем все вершины $x_j \notin S$ «неопробованными».

Шаг 2. Взять произвольную «неопробованную» вершину и для каждой вершины $x_i \in S$ вычислить «приращение» Δ_{ij} , соответствующее замене вершины x_i вершиной x_j , т. е. вычислить

$$\Delta_{ij} = \sigma(S) - \sigma((S \setminus \{x_i\}) \cup \{x_j\}). \quad (47)$$

Шаг 3. Найти

$$\Delta_j = \max_{x_i \in S} \{\Delta_{ij}\}.$$

(i) Если $\Delta_{i_0j} \leq 0$, то назвать вершину x_j «опробованной» и перейти к шагу 2.

(ii) Если $\Delta_{i,j} > 0$, то $S \leftarrow (S \cup \{x_j\}) - \{x_i\}$ назвать вершину x_j «опробованной» и перейти к шагу 2.

Шаг 4. Повторять шаги 2 и 3 до тех пор, пока все вершины из $X - S$ не будут опробованы. Эта процедура оформляется как цикл. Если при выполнении последнего цикла совсем не будет замещений вершин на шаге 3 (i), то перейти к шагу 5. В противном случае, т. е. если осуществлено некоторое замещение, назвать все вершины «неопробованными» и вернуться к шагу 2.

Шаг 5. Стоп. Текущее множество S является подходящей аппроксимацией p -медианного множества X_p .

Очевидно, что приведенный выше алгоритм не во всех случаях дает оптимальный ответ. Действительно, рассмотрим неориентированный граф, изображенный на рис. 11.

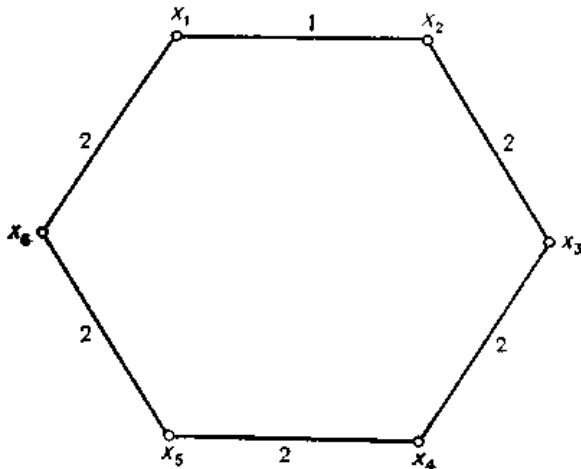


Рис. 11. Контрпример к приближенному алгоритму разд. 3.11.4.

Числа, стоящие около ребер, равны соответствующим реберным стоимостям. Считаем, что все вершины имеют единичные веса. Если искать 2-медиану и в качестве начального множества S взять $\{x_3, x_6\}$ с передаточным числом $\sigma(S) = 8$, то никакое замещение только одной вершины не приводит к множеству с меньшим передаточным числом. Однако множество $\{x_3, x_6\}$ не является 2-медианой данного графа. Существуют два 2-медианных множества: $\{x_1, x_4\}$ и $\{x_2, x_5\}$ с передаточными числами $\lambda(\bar{X}_2) = 7$.

Алгоритм, описанный в настоящем разделе, является в действительности лишь одним из целого семейства алгоритмов, бази-

рующихся на локальной оптимизации и на идее λ -оптимальности, впервые введенной Лином для задачи коммивояжера, а впоследствии развитой и использованной в других работах при исследовании различных комбинаторных проблем.

Вернемся к нашей задаче. Множество S из p вершин называется λ -оптимальным ($\lambda \leq p$), если замена любых λ вершин из множества S любыми λ вершинами, не принадлежащими S , не может породить нового множества с передаточным числом, меньшим, чем у S . Очевидно, что если S является p -медианным множеством \bar{X}_p рассматриваемого графа, то S p -оптимально. Совершенно очевидно также, что если S' является λ -оптимальным множеством и S'' — λ -оптимальное множество, то $\sigma(S'') \leq \sigma(S')$ при $\lambda'' > \lambda'$.

Согласно приведенным выше определениям решение, полученное с помощью приведенного выше алгоритма, можно назвать 1-оптимальным. Подобные алгоритмы можно дать и для порождения 2-оптимальных, 3-оптимальных и т. д. решений. Чтобы убедиться в λ -оптимальности множества S , нужно выполнить

$$\binom{p}{\lambda} \binom{n-p}{\lambda}$$

возможных замещений (а, следовательно, и вычислений передаточных чисел σ). Это число довольно быстро растет с ростом λ , а поэтому такие алгоритмы практически нельзя использовать для значений λ , больших 3.

Итерационный процесс для решения задачи построения 1-медианы в случае бесконечного графа, вложенного в евклидову плоскость, состоит в следующем.

Пусть G — бесконечный полный граф, представляющий евклидову плоскость, и пусть S , — заданное подмножество вершин, состоящее из n точек с координатами (x_i, y_i) , $i = 1, 2, \dots, n$. Требуется выбрать некоторую точку (x_m, y_m) для расположения в ней медианной вершины и чтобы при этом выражение

$$\sigma_m = \sum_{i=1}^n v_i d_{im}$$

было минимальным; здесь v_i — «вес» точки с координатами (x_i, y_i) и d_{im} евклидово расстояние:

$$d_{im} = \sqrt{[(x_i - x_m)^2 + (y_i - y_m)^2]}.$$

В точке минимума мы имеем:

$$\frac{\partial \sigma_m}{\partial x_m} = \sum_{i=1}^n \left[\frac{v_i (x_i - x_m)}{d_{im}} \right] = 0 \quad (48)$$

и

$$\frac{\partial \sigma_m}{\partial y_m} = \sum_{i=1}^n \left[\frac{v_i (y_i - y_m)}{d_{im}} \right] = 0. \quad (49)$$

Эти уравнения можно решить итерационным методом, представив их предварительно в следующем виде:

$$x_m = \sum_{i=1}^n \left[\frac{v_i x_i}{d_{im}} \right] / \sum_{i=1}^n \left[\frac{v_i}{d_{im}} \right], \quad (50)$$

$$y_m = \sum_{i=1}^n \left[\frac{v_i y_i}{d_{im}} \right] / \sum_{i=1}^n \left[\frac{v_i}{d_{im}} \right]. \quad (51)$$

Начиная с некоторой точки (x_m, y_m) , можно вычислить расстояния d_{im} , а затем, используя выражения (50) и (51), найти новое лучшее приближение для (x_m, y_m) . Затем заново вычисляются расстояния d_{im} и координаты x_m и y_m . Процесс продолжается до тех пор, пока не стабилизируются величины x_m и y_m .

Результат в этом итерационном процессе при нахождении глобального оптимума для (x_m, y_m) не зависит от выбора начального расположения этой точки.

4. Частичная упорядоченность

4.1. Графы частичных упорядочений

Ранее мы ввели понятие частичного упорядочения. Отношение *частичного упорядочения* $a \geq b$ характеризуется тремя свойствами:

1. Из $a \geq b$ и $b \geq a$ следует $a = b$.
2. Рефлексивность: $a \geq b$.
3. Транзитивность.

Обращение $a \leq b$ частичного упорядочения есть частичное упорядочение. Каждому частичному упорядочению $a \geq b$ соответствует *строгое частичное упорядочение* $a > b$, являющееся пересечением двух отношений

$$a \geq b, \quad a \neq b.$$

Граф G строгого частичного упорядочения характеризуется свойствами:

1. G есть ориентированный граф с однократными ребрами.
2. Ациклический.

3. Транзитивный.

Граф соответствующего частичного упорядочения получается добавлением петель в каждой вершине.

Как и выше для ациклических графов, будем говорить, что вершина b непосредственно следует за вершиной a , если $a > b$, и не существует такой вершины c , что $a > c > b$, т. е., другими словами, если ребро (a, b) существенное. Ориентированная цепь $P(a, b)$ максимальна, когда все ее ребра существенные. Для таких максимальных цепей имеют место свойства Жордана — Гёльдера.

Частичное упорядочение является транзитивным замыканием любого своего порождающего графа. Частичное упорядочение имеет не более одного базисного графа B , и если такой граф B существует, то он состоит из существенных ребер. Следует отметить, что, когда частичное упорядочение представляют в виде графа, обычно изображают не сам граф частичного упорядочения, а его базисный граф B . Базисный граф существует тогда и только тогда, когда для любого ребра (a, b) вершины a, b связаны максимальной цепью $P(a, b)$. В частности, это имеет место, когда цепи между a и b конечны.

4.2. Представления в виде сумм упорядоченных множеств

Обратимся теперь к частям не более чем первой степени и покажем, что особые свойства частичных упорядочений позволяют вывести более конкретные результаты, чем те, которые получены для общих ориентированных графов. Представим ориентированный граф G в виде двудольного графа $G(V, V')$, где V' есть копия V , с взаимно однозначным соответствием $a \longleftrightarrow a'$ между вершинами этих множеств. Каждому ребру (a, b) в G соответствует единственное ребро (a, b') в $G(V, V')$, и наоборот. Любому паросочетанию M в $G(V, V')$ соответствует часть H в G не более чем первой степени. Компоненты C_i графа H являются ориентированными простыми циклами и ориентированными простыми цепями. Пусть V_i означает множество вершин компоненты C_i . Тогда множества V_i не пересекаются, и имеют место разложения

$$H = \bigcup C_i, \quad V = \bigcup V_i. \quad (1)$$

Некоторые из множеств V_i могут состоять из одной вершины.

Предположим теперь, что G есть ациклический граф. Тогда компоненты C_i графа H в (1) будут конечными или бесконечными ориентированными простыми цепями, следовательно, подграфы $G(V_i)$

представляют собой квазиупорядочения. Поэтому любому паросочетанию M в $G(V, V')$ соответствует разложение (1) множества V , в котором каждое $G(V_i)$ является квазиупорядочением со счетным числом вершин. Если G есть частичное упорядочение, то множества V_i оказываются упорядоченными.

Дефицитными вершинами H являются те, в которых нет выходящих ребер, так что этими вершинами будут конечные вершины простых цепей C_i . Обратное дефицитными вершинами являются те, в которых нет входящих H -ребер; следовательно, таковыми будут начальные вершины простых цепей C_i . Если все простые цепи C_i конечны, то они будут иметь как начальные, так и конечные вершины; следовательно, в этом случае имеет место взаимно однозначное соответствие между дефицитными и обратное дефицитными вершинами. Если граф G конечен, то минимальное число дефицитных вершин равно максимальному дефициту G :

$$\delta_0 = \delta_0^*, \tag{2}$$

Отсюда следует

Теорема 1. Пусть в конечном ациклическом графе G с множеством вершин V

$$V = \bigcup V_i \tag{3}$$

есть такое непересекающееся разложение V , что для каждого класса V_i подграф $G(V_i)$ является квазиупорядоченным множеством. Тогда наименьшее число компонент V_i в любом таком разложении равно максимальному дефициту (2) графа G .

Заметим, что если $\nu_e(H)$ — число ребер части H не более чем первой степени, содержащей максимальное число ребер, то

$$\nu_e(H) = \nu(V) - \delta_0. \tag{4}$$

Это показывает, что графы H с минимальным числом квазиупорядоченных компонент будут также графами с максимальным числом ребер.

Исследуем более подробно случай строгого частичного упорядочения G . Подмножество S множества V будет называться *нижним отрезком*, если из $x < s$ для некоторого $s \in S$ следует $x \in S$. Аналогично определяется *верхний отрезок*. Каждое множество B порождает нижний отрезок $S(B)$, состоящий из всех $x \leq b$ для какого-нибудь $b \in B$; таким образом, B будет порождающим множеством для $S(B)$. *Минимальное порождающее множество* B_0 есть такое множество, что никакое его подмножество не порождает $S(B_0)$. Множество B_0 является минимальным порождающим множеством тогда и только

тогда, когда оно будет *независимым множеством*, т. е. когда ни для какой пары b_1 и b_2 его элементов не выполняется соотношение $b_1 > b_2$. Предположим сначала, что G — конечное частичное упорядочение. Если A — подмножество множества вершин V , то его дефицит дается формулой

$$\delta(A) = v(A) - v(G[A]),$$

где $G[A]$ есть множество вершин v с $v < a$ для некоторого $a \in A$, т. е.

$$S(A) = A \cup G[A].$$

Пусть для некоторого такого v мы имеем $v \notin A$. Так как G транзитивно,

$$G\{v\} \subseteq G[A];$$

следовательно,

$$\delta(A \cup v) = \delta(A) + 1.$$

Это показывает, что любое множество с максимальным дефицитом должно быть нижним отрезком. Пусть B_0 есть минимальное порождающее множество для такого отрезка A . Из

$$B_0 = A - G[A]$$

мы заключаем, что

$$\delta(A) = v(B_0),$$

т. е. что дефицит отрезка равен числу вершин в его минимальном порождающем множестве S_0 . Как следствие мы получаем теорему:

Теорема 2. *В конечном частичном упорядочении максимальный дефицит равен*

$$\delta_0 = v(B_0),$$

где B_0 — *независимое множество с максимальным числом элементов*. Критические множества, т. е. множества с максимальным дефицитом, являются отрезками, для которых B_0 есть наибольшее независимое множество с $\gamma(B_0) = \delta_0$. Критические множества образуют кольцо множеств; в частности, существуют наименьшее множество $N_0 = S(B_0)$, содержащееся во всех остальных, и наибольшее; множество $M_0 = S(B'_0)$, содержащее все остальные. Соответственно среди наибольших независимых множеств существуют единственное самое нижнее множество B_0 и единственное самое верхнее множество B'_0 , так что все элементы любого другого наибольшего множества содержатся между B_0 и B'_0 .

Объединяя результаты теорем 1 и 2, мы получаем для конечных частичных упорядочений теорему:

Теорема 3. *Пусть G — частичное упорядочение, в котором наибольшие независимые множества имеют δ элементов. Тогда множество вершин V имеет такое непересекающееся разложение (3)*

на δ компонент V_i , в котором каждое $G(V_i)$ является графом упорядочения.

Теорему 3 доказал Дилворт, распространивший ее также на бесконечные частичные упорядочения. Доказательство опирается на конечный случай и использует индукцию относительно максимального числа δ независимых вершин. Ясно, что для данного δ не может быть непересекающегося разложения на упорядочения менее чем с δ членами, так как это помешало бы существованию δ независимых элементов. Для $\delta = 1$ теорема тривиальна.

В общем случае каждое конечное подмножество F множества V имеет непересекающееся разложение на упорядочения с δ членами

$$F = \bigcup F_i, \tag{5}$$

где некоторые F_i могут быть пустыми. Будем называть множество C *сильно зависимым*, если для любого конечного множества F существует такое разложение (5) на упорядочения, что

$$C \cap F \subseteq F_1$$

для подходящей компоненты F_1 . Одна вершина сильно зависима. Если это условие применить к двум элементам, то мы получим, что множество C должно быть упорядоченным. Сумма

$$C_0 = \bigcup C_i$$

любого упорядоченного по включению семейства $\{C_i\}$ сильно зависимых множеств также будет сильно зависима, так как должно быть

$$C_0 \cap F \subseteq C_k \cap F$$

для некоторого подходящего множества C_k . Из принципа максимальной заключаем, что существуют максимальные сильно зависимые множества.

Чтобы завершить индукцию, покажем, что если C есть максимальное сильно зависимое множество, то остающееся множество

$$V_1 = V - C$$

содержит максимальное число $\delta_1 = \delta - 1$ независимых вершин. Не может быть $\delta_1 < \delta - 1$, так как это приводило бы к разложению V на упорядочения менее чем с δ членами. Пусть поэтому $\delta_1 = \delta$, так что V_1 содержит δ независимых элементов

$$a_1, a_2, \dots, a_\delta. \tag{6}$$

Так как множества $C \cup a_i$ не могут быть сильно зависимыми, для каждого a_i найдется такое конечное множество S_i , что не существует

разложения α упорядочения, в котором $(C \cup a_i) \cap S_i$ содержится в одной компоненте. Очевидно, $a_i = S_i$, так что конечное множество

$$S = S_1 \cup S_2 \cup \dots \cup S_\delta$$

содержит все вершины (6). Пусть

$$S = T_1 \cup T_2 \cup \dots \cup T_\delta$$

есть такое прямое разложение множества S на упорядочения, что

$$C \cap S \equiv T_1.$$

Так как S содержит независимые вершины (6), существует некоторое $a_i \in T_i$, Но тогда

$$S_i = (S_i \cap T_1) \cup \dots \cup (S_i \cap T_\delta)$$

является прямым разложением множества S_i на упорядочения c

$$(C \cup a_i) \cap S_i \cap T_1 \subseteq S_i \cap T_1,$$

что противоречит определению S_i .

Теорема Дилворта относится к частично упорядоченным множествам, для которых наибольшие независимые множества имеют ограниченное число элементов. При некоторых условиях эта теорема может быть распространена на частичные упорядочения с бесконечными независимыми множествами.

Разложение (3) частичного упорядочения на непересекающиеся упорядоченные множества называется *независимым*, если существует независимое множество

$$B_0 = \{b_i\}, \quad b_i \in V_i.$$

Если независимое разложение существует, то любое другое независимое множество может иметь не более одной вершины в каждом V_i . Мы видим, что теорема 3 равносильна следующей:

Т е о р е м а 4. *Частичное упорядочение, в котором наибольшие независимые множества состоят из δ элементов, имеет независимое разложение на упорядочения.*

Доказательство. При данном условии независимое разложение содержит по крайней мере δ упорядочений и не может иметь больше. С другой стороны, когда теорема 3 справедлива, должно быть по одной из δ вершин в каждом V_i .

Частичное упорядочение G называется *локально конечным*, если для данного v существует только конечное число вершин x , удовлетворяющих условиям $v > x$ или $v < x$. Отсюда следует, что все простые цепи в G конечны. Так как двудольный граф $G(V, V')$, отвечающий G , также будет локально конечен, то следует, что он

имеет максимальное наросочетание M . Исходя из этого свойства, можно доказать следующее:

Теорема 5. Любое локально конечное частичное упорядочение имеет независимое разложение на упорядочения.

4.3. Структуры и структурные операции

Отношения замыкания. Предположим, что частичное упорядочение P обладает следующим свойством: для каждой пары элементов a и b существует единственный элемент

$$m = a \cup b,$$

называемый *структурным объединением* a и b , для которого

$$x \geq a, \quad x \geq b$$

тогда и только тогда, когда $x \geq m$.

Из этого определения легко получить свойства:

1. Идемпотентность: $a \cup a = a$.
2. Коммутативность: $a \cup b = b \cup a$. (7)
3. Ассоциативность: $(a \cup b) \cup c = a \cup (b \cup c)$.

Если в P существует структурное объединение, то, его можно рассматривать как алгебраическую операцию, при которой любым двум элементам a и b отвечает вполне определенный третий элемент $a \cup b$. Кроме того, $a \geq b$ в P тогда и только тогда, когда

$$a \cup b = a. \quad (8)$$

Операцию, удовлетворяющую условиям (7), будем называть *структурной операцией*. Для нее можно доказать также обратный результат:

Теорема 4. Структурная операция определяет частичное упорядочение со структурным объединением $a \cup b$, если считать $a \geq b$, когда выполняется соотношение (8).

Доказательство простое и может быть предоставлено читателю в качестве упражнения.

Можно считать, что вместо структурного объединения частичное упорядочение имеет однозначно определенный элемент $a \cap b$, называемый *структурным пересечением*, для которого

$$y \leq a, \quad y \leq b$$

тогда и только тогда, когда

$$y \leq a \cap b.$$

Структурное пересечение обладает теми же свойствами (7), что и структурное объединение. Теорема, аналогичная теореме 4, также справедлива, если считать, что $a \geq b$, когда

$$a \cap b = b. \tag{9}$$

Частичное упорядочение называется *структурой*, если любая пара элементов a и b имеет как структурное объединение $a \cup b$, так и структурное пересечение $a \cap b$. Вместо того, чтобы вводить структуру при помощи исходного частичного упорядочения, можно просто рассматривать структуру как алгебраическую систему с операцией структурного объединения $a \cup b$ и операцией структурного пересечения $a \cap b$. Эти операции удовлетворяют двойственным системам аксиом:

$$\begin{aligned} a \cup a &= a, & a \cap a &= a, \\ a \cup b &= b \cup a, & a \cap b &= b \cap a, \\ a \cup (b \cap c) &= (a \cup b) \cap c, & a \cap (b \cup c) &= (a \cap b) \cup c, \\ a \cup (a \cap b) &= a, & a \cap (a \cup b) &= a. \end{aligned} \tag{10}$$

Первые три аксиомы из каждой системы в (10), как мы видели, определяют частичное упорядочение. Последние две аксиомы суть *законы поглощения*. Они гарантируют, что два частичных упорядочения $a \geq b$, определенные соответственно формулами (8) и (9), совпадают. Так, из (8) следует

$$a \cap b = (a \cup b) \cap b = b$$

и, обратно, (8) следует из (9).

Теория структур пронизывает большую часть математики, но скольконибудь подробное описание ее применений выходит за пределы нашего изложения, и мы отсылаем читателя к специальным работам на эту тему. Добавим только несколько утверждений.

Мы определили структуру как такое частичное упорядочение, в котором любые два элемента a и b имеют точную верхнюю границу $a \cup b$ и точную нижнюю границу $a \cap b$. Следовательно, эти понятия определены для любого конечного множества элементов. Структура называется *полной*, если любое множество

$$A = \{a_i\}, \quad i \in I,$$

имеет структурное объединение, или точную верхнюю границу и структурное пересечение, или точную нижнюю границу:

$$\mathbf{V} = \bigvee_{i \in I} a_i, \quad \mathbf{\Lambda} = \bigwedge_{i \in I} a_i.$$

Тогда каждая из систем условий

$$x \geq a_i, \quad y \leq a_i, \quad i \in I,$$

будет выполняться тогда и только тогда, когда соответственно

$$x \geq \mathbf{V}, \quad y \leq \mathbf{\Lambda}.$$

Полная структура L имеет единственный максимальный элемент m_1 , являющийся структурным объединением всех элементов L , и единственный минимальный элемент m_0 , являющийся структурным пересечением всех элементов.

Для полных структур система аксиом может быть сокращена. Достаточно предполагать, что:

1. Каждое множество имеет структурное пересечение.
2. Существует единственный максимальный элемент m_1 , содержащий все остальные.

Структурное объединение множества A можно тогда определить как структурное пересечение

$$\bigvee_{a_i \in A} a_i = \bigwedge_{u \in U} u,$$

где U есть непустое множество всех элементов u , которые содержат все $a_i \in A$.

Полные структуры связаны с *отношениями замыкания* для множеств. Пусть S — некоторое основное множество. Отношение замыкания в S есть отображение

$$A \rightarrow \Gamma(A),$$

которое каждому подмножеству A множества S ставит в соответствие некоторое множество $\Gamma(A)$, причем это соответствие обладает следующими свойствами:

1. $\Gamma(A) \geq A$.
2. $\Gamma(\Gamma(A)) = \Gamma(A)$.
3. Из $A \geq B$ следует $\Gamma(A) \geq \Gamma(B)$.

Множества-образы $\Gamma(A)$ называются *замкнутыми множествами* для этого отношения. Условие 1 показывает, что $S = \Gamma(S)$ является замкнутым множеством.

Теорема 5. *Замкнутые множества для отношения замыкания Γ образуют полное кольцо R пересечений множеств, содержащее S , и замыкание $\Gamma(A)$ множества A есть наименьшее множество в R , содержащее A . Обратно, любое полное кольцо пересечений R , содержащее S , определяет отношение замыкания указанным способом.*

Доказательство. Под *полным кольцом пересечений* R множеств понимается такое семейство множеств, что пересечение любого его подсемейства принадлежит R . Пусть

$$\{D_i\}, \quad \Gamma(D_i) = D_i,$$

— семейство замкнутых множеств для отношения замыкания Γ . Чтобы показать, что их пересечение

$$D = \bigcap D_i$$

замкнуто, заметим, что из

$$D \subseteq D_i, \quad \Gamma(D) \subseteq \Gamma(D_i) = D_i$$

следует

$$\Gamma(D) = \bigcap D_i = D.$$

Обратно, пусть R — полное кольцо пересечений множеств, содержащее S . Можно определить

$$\Gamma(A) = \bigcap C_i,$$

где C_i пробегает все множества из R , содержащие A ; легко видеть, что $\Gamma(A)$ удовлетворяет трем условиям для отношения замыкания.

Теория полных структур и теория отношений замыкания равносильны в следующем смысле. Замкнутые множества для отношения замыкания образуют полную структуру. Структурным пересечением семейства замкнутых множеств $\{\Gamma(A_i)\}$ является их пересечение, а структурным объединением — замыкание их суммы

$$\bigvee_i \Gamma(A_i) = \Gamma\left(\bigcup_i A_i\right). \quad (11)$$

Обратно, в полной структуре множества $\Gamma(a)$ всех $x \leq a$ образуют кольцо пересечений множеств, следовательно, определяют отношение замыкания в множестве вершин V .

Частичное упорядочение является стругаурой, если существует единственный минимальный элемент $a \cup b$, содержащий a и b , и единственный максимальный элемент $a \cap b$, содержащийся в a и b . В более общем случае может быть такое минимальное множество элементов $\{c_i\}$, что $x \geq a$, $x \geq b$ тогда и только тогда, когда $x \geq c_i$ для некоторого i , и аналогично для $x \leq a$, $y \leq b$. Это наводит на мысль об определении частичного упорядочения P как мультиструктуры, в которой заданы многозначные операции

$$a \cup b = \{c_i\}, \quad a \cap b = \{d_i\}.$$

Существуют соответствующие аксиомы для этого случая.

4.4. Размерность в частичном упорядочении

Ранее мы установили, что каждый ациклический граф G содержится в квазиупорядочении O с тем же множеством вершин. Если граф G транзитивен и, следовательно, является частичным упорядочением, то он будет содержаться в транзитивном замыкании для O . Отсюда мы получаем следующий результат, впервые доказанный Шпильрайном. Всякое частичное упорядочение содержится в некотором упорядочении. В этом случае то же рассуждение дает теорему Душника и Миллера:

Теорема 6. Любое пересечение упорядочений на множестве V является частичным упорядочением, и любое частичное упорядочение есть пересечение содержащих его упорядочений.

На основе этой теоремы можно сформулировать следующее определение. *Порядковая размерность $d_0(G)$ частичного упорядочения G есть наименьшее кардинальное число упорядочений O_i на V , для которого*

$$G = \bigcap_i O_i, \quad (12)$$

Можно также ввести понятие размерности частичного упорядочения и другим способом, который более сходен с введением обычной декартовой размерности для произведения пространств. Предположим, что $\{O_i\}$ есть семейство упорядочений, определенных соответственно на множествах V_i ; индекс i может при этом пробегать произвольное множество индексов I . Введем в рассмотрение произведение множеств

$$V = V_1 \times V_2 \times \dots \times V_k = \prod_{i \in I} V_i, \quad (13)$$

состоящее из всех k -наборов

$$v = (v_1, v_2, \dots, v_k), \quad v_i \in V_i.$$

Упорядочения $\{O_i\}$ определяют частичное упорядочение Q , называемое *произведением упорядочений*

$$Q = \prod_{i \in I} Q_i \quad (14)$$

на произведении множеств (13): если

$$w = (w_1, w_2, \dots, w_k),$$

то $v > w$ в Q тогда и только тогда, когда для каждого i

$$v_i \geq w_i,$$

где точное равенство не может выполняться одновременно для всех индексов. Фактически частичное упорядочение Q есть структура с операциями

$$v \cup w = (\max(v_i, w_i)), \quad v \cap w = (\min(v_i, w_i)).$$

Кроме того, эта структура является *дистрибутивной*, т. е. структурные операции удовлетворяют в ней двойственным друг другу законам *дистрибутивности*

$$\begin{aligned} u \cap (v \cup w) &= (u \cap v) \cup (u \cap w), \\ u \cup (v \cap w) &= (u \cup v) \cap (u \cup w), \end{aligned}$$

в чем читатель может убедиться.

Введем следующее определение. *Мультипликативной размерностью* $d_p(G)$ для частичного упорядочения G называется наименьшее кардинальное число упорядочений Q_i , для которого G изоморфно подграфу произведения этих упорядочений Q .

Теорема 7. *Порядковая размерность и мультипликативная размерность частичного упорядочения G совпадают:*

$$d_o(G) = d_p(G). \quad (15)$$

Доказательство. Предположим сначала, что G имеет представление (12) в виде пересечения $k = d_o(G)$ упорядочений на его множестве вершин V . Введем k копий V'_i множества V и на каждой из них — упорядочение O'_i , изоморфное O_i . Каждому $a \in V$ будет соответствовать единственный элемент a'

$$a \leftrightarrow a' = (a'_i) \quad (16)$$

в произведении пространств

$$V' = \prod_i V'_i. \quad (17)$$

Согласно представлению (12) мы имеем $a > b$ в G тогда и только тогда, когда $a > b$ для каждого O_i в V , следовательно, когда $a'_i > b'_i$ для каждого O'_i в V'_i . Это означает, что взаимно однозначное соответствие (16) таково, что $a > b$ в G тогда и только тогда, когда $a' > b'$ для произведения упорядочений

$$G' = \prod_i O'_i. \quad (18)$$

Значит, G изоморфно подграфу произведения упорядочений O' с k множителями. Отсюда следует, что

$$d_o(G) \geq d_p(G).$$

Пусть теперь Q есть произведение (14) $l = d_p(G)$ упорядочений $\{O_i\}$, которые определены на множествах V_i , имеющих своим произведением множество (13). Построим такое семейство упорядочений $\{O_i\}$ на V , что Q имеет представление (12). Для этого мы вполне упорядочим множество индексов I для упорядочений Q_i и определим каждое упорядочение O_i на V следующим образом. Положим $v > w$ для двух элементов v и w из V , если либо $v_i > w_i$, либо в случае $v_i = w_i$,

$v_j > w_j$, где j есть первый индекс в I , для которого $v_j \neq w_j$. Читатель может убедиться, что O_i действительно является упорядочением на V . Чобы показать, что

$$Q = \bigcap_i O_i, \tag{19}$$

заметим, что $v > w$ в Q может иметь место только в том случае, когда это включение выполняется для каждого упорядочения O_i , и наоборот. Предположим, что G есть подграф в Q . Тогда (19) дает

$$G = \bigcap_i (G \cap O_i);$$

следовательно, G является пересечением I упорядочений. Отсюда следует

$$d_0(G) \leq d_p(G),$$

и мы получаем (15).

Из теоремы 7 следует, что можно говорить просто о размерности частичного упорядочения G и обозначать ее через $d(G)$. Приведем некоторые свойства этой размерностной функции.

Теорема 8. *Размерность произведения упорядочений Q из (14) с k множителями равна $d(Q) = k$.*

Доказательство. Мы уже убедились в том, что Q может быть в данном случае представлено как пересечение (19) k упорядочений. Поэтому $d(Q) \leq k$. Чобы доказать, что $d(Q) = k$, достаточно установить, что Q не может быть пересечением меньшего чем k числа упорядочений на V . Для проверки этого выберем две вершины $a_i > b_i$ в множестве V_i для каждого упорядочения Q_i и построим вершины в V :

$$\alpha^{(i)} = (\dots b_x \dots a_i \dots b_y \dots),$$

$$\beta^{(i)} = (\dots a_x \dots b_i \dots a_y \dots),$$

где все компоненты $\alpha^{(i)}$, кроме a_i , равны b_x , а все компоненты $\beta^{(i)}$, кроме b_i , равны a_x . В Q

$$\beta^{(j)} > \alpha^{(i)}, \quad i \neq j,$$

а $\alpha^{(i)}$ и $\beta^{(i)}$ несравнимы.

Так как Q является пересечением упорядочений, существует некоторое упорядочение O , на V с

$$\alpha^{(i)} > \beta^{(i)},$$

а также некоторое упорядочение O^*_i с обратным соотношением. Но ни в каком O_i не может быть

$$\alpha^{(j)} > \beta^{(i)}, \quad j \neq i,$$

так как это привело бы к ориентированному циклу

$$\alpha^{(i)} > \beta^{(1)} > \alpha^{(1)} > \beta^{(i)} > \alpha^{(i)}.$$

Поэтому для каждого i должно существовать хотя бы одно O_i в представлении в виде пересечения (19) для Q . Это завершает доказательство теоремы 8. Частным ее случаем оказывается теорема, доказанная Коммом:

Теорема 9. *Размерность структуры L всех подмножеств множества S равна мощности S .*

Доказательство. Семейство подмножеств S изоморфно произведению множеств (13), в котором компонентами являются $v_i = 0$ или $v_i = 1$.

4.5. Соответствия Галуа

Будем обозначать через P и Q два частично упорядоченных множества. Предположим, что существует отображение ρ , переводящее P в Q :

$$p \rightarrow \rho(p) \in Q, \quad (20)$$

а также отображение π , переводящее Q в P :

$$q \rightarrow \pi(q) \in P. \quad (21)$$

Эти отображения будем называть парой *соответствий Галуа* между P и Q , если выполняются следующие два условия:

1. Если $p_1 \geq p_2$ — два элемента из P и $q_1 \geq q_2$ — элементы из Q , то

$$\rho(p_1) \leq \rho(p_2), \quad \pi(q_1) \leq \pi(q_2). \quad (22)$$

2. Для любых элементов $p \in P$ и $q \in Q$

$$\rho\pi(q) \geq q, \quad \pi\rho(p) \geq p. \quad (23)$$

Если определена пара соответствий Галуа, то будем говорить, что между P и Q имеет место *связь Галуа*. Эта терминология заимствована из теории Галуа для уравнений. В ней изучается соответствие между подполями данного поля F и подгруппами группы автоморфизмов A поля F . Для каждого множества элементов $X \subseteq F$ существует подгруппа $A(X)$, состоящая из тех автоморфизмов, которые оставляют инвариантным каждый элемент из X . Обратно, для каждого подмножества $Y \subseteq A$ существует поле $F(Y)$, состоящее из тех элементов F , которые инвариантны при автоморфизмах из Y . Можно проверить, что условия

$$A(X) \cong A(X_1) \text{ при } X \supseteq X_1,$$

$$F(Y) \cong F(Y_1) \text{ при } Y \supseteq Y_1$$

выполнены; кроме того,

$$F(A(X)) \cong X, \quad A(F(Y)) \cong Y,$$

так что определено соответствие Галуа между подмножествами множеств F и A .

Из многочисленных примеров соответствий Галуа упомянем следующие.

1. В векторном пространстве V каждому подмножеству A множества V отвечает множество $O(A)$, состоящее из всех векторов, ортогональных векторам из A . Нетрудно проверить, что

$$O(A) \subseteq O(B) \text{ при } A \supseteq B, \\ O(O(A)) \supseteq A,$$

так что определена связь Галуа для подмножеств V .

2. В группе G каждому подмножеству A соответствует централизатор $C(A)$. Здесь $C(A)$ состоит из всех элементов c , коммутирующих с каждым элементом a из A :

$$a \cdot c = c \cdot a.$$

3. В кольце каждому подмножеству A отвечает аннулятор $N(A)$, состоящий из элементов n , для которых $n \cdot a = 0$ для каждого $a \in A$. Исследуем свойства соответствий Галуа. Условия (22) показывают, что соответствия Галуа обращают включение. Из свойств (22) и (23) следует, что для любого $p \in P$ должно быть одновременно

$$\rho \pi(p) \supseteq \rho(p), \quad \pi \rho(p) \subseteq \rho(p).$$

Таким образом, мы получаем теорему:

Теорема 10. *Если ρ и π — соответствия Галуа, то*

$$\rho \pi(p) = \rho(p), \quad \pi \rho(q) = \pi(q). \quad (24)$$

Отображения $\pi \rho$ и $\rho \pi$ переводят P и Q в себя. Их можно назвать операциями замыкания Галуа, а образы

$$\Gamma_P(p) = \pi \rho(p), \quad \Gamma_Q(q) = \rho \pi(q) \quad (25)$$

— замыканиями Галуа соответственно для p и q . Читатель может проверить, что они обладают следующими свойствами:

Теорема 11. *Замыкания Галуа (25) имеют три свойства замыкания в P :*

1. *Если $p_1 \supseteq p_2$, то $\Gamma_P(p_1) \supseteq \Gamma_P(p_2)$.*
2. *$\Gamma_P(p) \supseteq p$.*
3. *$\Gamma_P \Gamma_P(p) = \Gamma_P(p)$,*

и аналогично в Q .

Следует отметить, что элементы

$$\bar{p} = \Gamma_P(p), \quad \bar{q} = \Gamma_Q(q)$$

являются замкнутыми элементами в P и Q :

Теорема 12. *Замкнутые элементы p и q в P и Q — это те элементы, которые являются образами элементов другого множества:*

$$\bar{p} = \pi(q), \quad \bar{q} = \rho(p), \quad (26)$$

и связь Галуа определяет взаимно однозначные отображения для множеств замкнутых элементов.

Доказательство. Элементы p вида (26) замкнуты согласно (24). С другой стороны, если p — замкнутый элемент, то он является образом при π :

$$\bar{p} = \pi(\rho(p)).$$

Наконец, если

$$\bar{p} = p, \quad \bar{q} = q$$

в (26), то одно из соотношений

$$\bar{p} = \pi(\bar{q}), \quad \bar{q} = \rho(\bar{p})$$

влечет за собой другое.

Пусть P_1 и Q_1 — множества замкнутых элементов соответственно в P и Q . Связь Галуа называется *совершенной* в P , если $P = P_1$, т. е. если каждое $p \in P$ замкнуто:

$$\pi\rho(p) = p.$$

Связь называется *совершенной*, если она совершенна в P и в Q . В ряде вопросов, например в теории Галуа для уравнений, важно знать когда это имеет место. Оказывается полезным следующий критерий.

Теорема 13. Связь Галуа между P и Q совершенна в P тогда и только тогда, когда любые два различных элемента $p_1 > p_2$ имеют различные образы:

$$\rho(p_1) < \rho(p_2).$$

Доказательство предоставляется читателю.

Рассмотрим общее построение связей Галуа. Множество P_1 замкнутых элементов является таким частично упорядоченным подмножеством множества P , что для каждого $p \in P$ существует единственный минимальный элемент $\bar{p} \in P_1$ с $\bar{p} \geq p$. Аналогично замкнутые элементы \bar{q} образуют такое подмножество Q_1 множества Q , что для любого $q \in Q$ существует минимальный элемент $\bar{q} \in Q_1$ с $\bar{q} \geq q$. Между P_1 и Q_1 имеется обратный изоморфизм, т. е. взаимно однозначное отображение π

$$P_1 = \pi(Q_1), \quad Q_1 = \pi^{-1}(P_1),$$

для которого из $q_1 > q_2$ в Q_1 следует

$$\pi(q_1) < \pi(q_2)$$

в P_1 , и наоборот. Обратное, если существуют такие подмножества P_1 и Q_1 в P и Q , то соответствия Галуа могут быть определены.

В большинстве приложений множества P и Q являются полными структурами. Установим, что тогда множеству P_1 и Q_1 также будут полными структурами. Пусть $\{\bar{p}_i\}$ — некоторое множество замкнутых элементов. Наибольшим элементом из P , содержащимся во всех \bar{p}_i , будет структурное пересечение

$$\bar{d} = \bigwedge \bar{p}_i.$$

Но из

$$d \leq \bar{p}_i, \quad \bar{d} \leq \bar{p}_i,$$

закключаем, что

$$\bar{d} \leq \bigwedge \bar{p}_i = d, \quad \bar{d} = d.$$

Так как d замкнуто, структурные пересечения в P и в P_1 совпадают. Наименьшим элементом в P , содержащим все \bar{p}_i , является структурное объединение

$$u = \bigvee \bar{p}_i.$$

Поэтому наименьший замкнутый элемент, содержащий все \bar{p}_i , есть замыкание

$$\bar{u} = \overline{\bigvee \bar{p}_i} = \bigvee \bar{p}_i.$$

Мы видим, что структурные объединения в P и в P_1 , вообще говоря, не совпадают. Будем называть P_1 *полной подструктурой P относительно структурного пересечения*, и аналогично для Q_1 в Q .

Полные структуры P и Q имеют максимальные и минимальные элементы:

$$m^{(P)}, m^{(Q)}, n^{(P)}, n^{(Q)}.$$

Точно так же P_1 и Q_1 имеют максимальные и минимальные элементы:

$$m^{(P_1)}, m^{(Q_1)}, n^{(P_1)}, n^{(Q_1)}.$$

Ясно, что

$$m^{(P)} \bar{m}^{(P)} = m^{(P_1)}, \quad m^{(Q)} \bar{m}^{(Q)} = m^{(Q_1)} \tag{27}$$

и что

$$m^{(P)} \rightarrow n^{(Q_1)}, \quad m^{(Q)} \rightarrow n^{(P_1)}$$

при связи Галуа.

Обратно, пусть P и Q имеют такие полные подструктуры P_1 и Q_1 относительно структурного пересечения, что выполняется (27). Когда P_1 и Q_1 обратно изоморфны, связь Галуа между P и Q определяется,

если каждому $p \in P$ соотнести в качестве замыкания \bar{p} наименьший элемент в P_1 , содержащий p , и аналогично для Q . Соответствия Галуа определяются формулами

$$\begin{aligned} \rho(\bar{p}_1 \cup \bar{p}_2) &= \overline{\bar{q}_1 \cup \bar{q}_2}, \\ \rho(\overline{\bar{p}_1 \cup \bar{p}_2}) &= \bar{q}_1 \cap \bar{q}_2 \end{aligned}$$

п аналогично для π .

Эти рассуждения справедливы, в частности, при $P = Q$, т. е. для связей Галуа в полной структуре P . Могут представиться различные частные случаи. Может быть $P_1 = Q_1$, и соответствие Галуа будет *двойственным себе*; это предполагает, что P_1 имеет *обратный*, или *двойственный автоморфизм*, т. е. взаимно однозначное отображение, переводящее структурные объединения в структурные пересечения и наоборот. Такая связь Галуа будет *инволютивной*, если двойственный автоморфизм π есть *инволюция*, т. е. такое отображение порядка 2, что

$$\bar{p} \rightarrow \bar{q} = \pi(\bar{p}), \quad \bar{q} \rightarrow \pi(\bar{q}) = \bar{p}.$$

Наконец, пусть P_1 есть *структура с дополнениями*, так что каждое $p_1 \in P_1$ имеет хотя бы одно *дополнение* p_1 , определяемое формулами

$$p_1 \cup p_1' = m_1, \quad p_1 \cap p_1' = n_1.$$

Полярность, или *ортогональность*, есть инволютивная связь Галуа, которая переводит каждое $p_1 \in P_1$ в его особое *полярное*, или *ортогональное, дополнение* p_1^* , обладающее следующими свойствами:

$$\begin{aligned} p_1 \rightarrow p_1^* &= \pi(p_1), \quad p_1^{**} = p_1, \\ p_1 \cup p_1^* &= m_1, \quad p_1 \cap p_1^* = n_1. \end{aligned}$$

4.6. Связи Галуа для бинарных отношений

Проиллюстрируем теорию связей Галуа, применяя ее к бинарным отношениям. Пусть V и V' — два множества, которые могут и совпадать. Как известно, бинарное отношение R есть соотношение

$$aRa', \tag{28}$$

которое выполняется для некоторых пар элементов (a, a') в произведении $V \times V'$. Обозначим через $R(a)$ подмножество множества V' , состоящее из всех вершин a' , для которых выполняется соотношение (28). Обратное отношение R^* определяем, как и раньше, полагая

$$a'R^*a,$$

если выполняется (28). Множество $R^*(a')$ состоит из всех $a \in V$, удовлетворяющих (28).

Отношение R ставит в соответствие каждому множеству A в V два производных множества

$$\hat{R}(A) = \bigcap_{a \in A} R(a), \quad \check{R}(A) = \bigcup_{a \in A} R(a), \quad (29)$$

содержащихся в V' . Множество $\hat{R}(A)$ состоит из всех $a' \in V'$, для которых (28) выполняется при каждом $a \in A$; множество $\check{R}(A)$ состоит из всех a' , для которых (28) выполняется хотя бы при некотором $a \in A$.

Мы проведем исследование только множеств $\hat{R}(A)$. Определение (29) показывает, что они образуют полное кольцо пересечений множеств.

Минимальное множество этого кольца есть $\check{K}(V)$, состоящее из всех таких a' , что (28) выполняется при каждом a . Максимальное множество есть

$$\hat{R}(\emptyset) = V'.$$

Для обратного отношения получается аналогичная картина. Каждому подмножеству A' множества V' отвечает множество $\check{K}^*(A')$, состоящее из всех $a \in V$, для которых (28) выполняется при каждом $a' \in A'$. Как и в (29), мы имеем

$$\check{R}^*(A') = \bigcap_{a' \in A'} R^*(a'). \quad (30)$$

Эти множества образуют полное кольцо пересечений с $\check{K}^*(V)$ в качестве минимального элемента. Из (29) и (30) следует

$$\hat{R}(A_1) \subseteq \hat{R}(A_2), \quad A_1 \supseteq A_2,$$

и аналогично для \check{K}^* ; кроме того,

$$\hat{R}^* \hat{R}(A) \supseteq A, \quad \hat{R} \hat{R}^*(A') \supseteq A',$$

так что можно утверждать следующее:

Теорема 14. *Каждое бинарное отношение R и его обращение R^* определяют соответствия Галуа*

$$A \rightarrow \hat{R}(A), \quad A' \rightarrow \hat{R}^*(A') \quad (31)$$

для подмножеств множеств V и V' .

Из теории связей Галуа, в частности, следует

Теорема 15. *Для бинарного отношения R и его обращения R^* мы имеем*

$$\hat{R} \hat{R}^* \hat{R}(A) = \hat{R}(A), \quad \hat{R}^* \hat{R} \hat{R}^*(A') = \hat{R}^*(A'). \quad (32)$$

Замыканиями (25) множеств A и A' в этом случае будут

$$\Gamma(A) = \widehat{R}^* \widehat{R}(A), \quad \Gamma^*(A') = \widehat{R} \widehat{R}^*(A').$$

Важно отметить, что теорема 14 допускает обращение:

Теорема 16. Любая связь Галуа Γ между структурами V и Q всех подмножеств множеств V и V' получается из отображений (31) для подходящего отношения R .

Доказательство. Пусть P_I и Q_I — подструктуры замкнутых множеств в Γ со связывающим их двойственным изоморфизмом α :

$$\bar{A} \rightarrow \alpha(\bar{A}), \quad \bar{A}' \rightarrow \alpha^{-1}(\bar{A}').$$

Введем между V и V' отношения

$$a R a', \quad a' R^* a, \tag{33}$$

которые выполняются соответственно при

$$a' \in \alpha(\bar{a}), \quad a \in \alpha^{-1}(\bar{a}'). \tag{34}$$

Первое из написанных условий дает нам

$$\bar{a}' \subset \alpha(\bar{a}),$$

и, применяя α^{-1} , мы получаем

$$\alpha^{-1}(\bar{a}') \supset \bar{a} \supset a,$$

так что R^* есть обращение R . Кроме того,

$$\widehat{R}(A) = \bigcap_{a \in A} \widehat{R}(\bar{a}) = \bigcap_{a \in A} \alpha(\bar{a}),$$

и при обратном изоморфизме мы получим

$$\widehat{R}^*(A) = \alpha(\bar{V} \bar{a}) = \alpha(\bar{A}).$$

Это показывает, что замкнутые множества для Γ те же самые, что для R и R^* .

Будем далее предполагать, что R есть бинарное отношение на множестве V . Для того чтобы R было двойственным себе, т. е. для того, чтобы структуры замкнутых множеств совпадали, необходимо и достаточно, чтобы при каждом a было

$$R(a) = \bigcap_{b \in B_a^*} R^*(b),$$

$$R^*(a) = \bigcap_{b \in B_a} R(b),$$

для подходящих множеств B_a и B_a^* . Это свойство можно выразить в следующей форме:

Теорема 17. Для того чтобы отношение R определяло двойственную себе связь Галуа, необходимо и достаточно, чтобы для каждого $a \in V$ существовали такие множества B_a и B_a^* , что соотношения

$$aR^*x, \quad aR^*y$$

выполняются тогда и только тогда, когда,

$$B_a^*R^*x, \quad B_aRy.$$

Ввиду сказанного такие отношения можно также называть *слабо симметрическими*.

Пусть теперь $R R^*$ — симметрическое отношение. В этом случае (32) сводится к

$$\widehat{R}\widehat{R}\widehat{R}(A) = \widehat{R}(A).$$

Это означает, что соответствие между замкнутыми множествами имеет порядок 2 и, следовательно, является инволюцией. Обратно, каждое инволютивное соответствие Галуа между подмножествами V может быть определено симметрическим отношением, так как в этом случае условия (34) сводятся к одному условию, и соотношения (33) совпадают.

Теорема 18. Симметрическое отношение определяет инволютивную связь Галуа в семействе всех подмножеств основного множества V ; обратно, каждая такая связь Галуа получается из симметрического отношения.

Отношение R рефлексивно, если aRa для каждого a , и антирефлексивно, если это не имеет места ни для какого a . Обратное отношение R^* рефлексивно или антирефлексивно вместе с R . Если R антирефлексивно, то $R(a)$ не содержит a ; следовательно,

$$a \cap \widehat{R}(A) = \emptyset$$

для любого множества A и, в частности,

$$\widehat{R}(A) \cap \widehat{R}\widehat{R}(A) = \emptyset.$$

Для соответствия Галуа, определяемого отношением R , из этого последнего свойства замкнутых множеств следует, что для образов при двойственных изоморфизмах должно быть

$$\widehat{R}^*\widehat{R}(A) \cup \widehat{R}^*\widehat{R}\widehat{R}(A) = V$$

в Q_1 . В P_1 мы аналогично получаем ^{*}

$$\widehat{R}\widehat{R}^*(A) \cup \widehat{R}\widehat{R}^*\widehat{R}^*(A) = V.$$

Предположим теперь, что отношение R симметрическое, и, следовательно, для замкнутых множеств мы имеем $P_1=Q_1$. Установленные результаты сводятся к равенствам

$$\widehat{R}(A) \cap \widehat{R}\widehat{R}(A) = \emptyset, \quad \widehat{R}(A) \cup \widehat{R}\widehat{R}(A) = V.$$

Таким образом, структура замкнутых множеств будет структурой с дополнениями, и R определяет в ней полярность. Как и выше, из (22) и (34) легко получить обратное. Тем самым установлена

Теорема 19. *Симметрическое антирефлексивное отношение определяет связь Галуа с полярностью для семейства всех подмножеств V ; обратно, любая такая полярность получается из подходящего симметрического антирефлексивного отношения в V .*

Из различных отношений, которые можно получить при помощи связей Галуа, упомянем замкнутые отношения, при которых

$$\hat{R}^*(\hat{R}(A)) = A. \quad (35)$$

Соответствующие множества $\hat{R}(A)$, определенные в (29) для отношения R , имеют свойства, аналогичные свойствам, полученным для $\check{R}(A)$. Обозначим здесь через $S = \bar{R}$ дополнительное отношение для R , т.е. такое отношение, что

$$aSa'$$

выполняется тогда и только тогда, когда (28) не выполнено. В этом случае можно проверить, что для S соответствующими производными множествами являются

$$\hat{S}(A) = C'(\check{R}(A)), \check{S}(A) = C'(\hat{R}(A)),$$

где C' обозначает дополнительное множество в V' . Для обращения S имеют место аналогичные формулы.

4.7. Отношения чередующегося произведения

Произведение двух отношений R и S определяется так же, как и произведение графов. Именно, отношение

$$aR\check{S}a''$$

выполняется тогда и только тогда, когда существует такая вершина $a' \in V'$, что

$$aRa', a'Sa'',$$

другими словами, существует маршрут (a, a') , (a', a'') длины 2 от a к a'' . Произведение нескольких отношений определяется аналогично, так что

$$R_1 R_2 \dots R_n [A] = \check{R}_n \dots \check{R}_1 (A)$$

есть множество тех вершин, которые могут быть достигнуты от вершин из A маршрутами из k ребер, лежащих по порядку в R_1, \dots, R_k . Если все R_i равны $R_i = R$, то мы получаем множества $R^n[A]$, определяемые посредством степени отношения R^n .

Рассмотрим различные отношения, характеризующиеся свойствами произведений. Простейшим из них является *транзитивное отношение* в множестве V , определяемое условием

$$R^2 \subseteq R, \quad (36)$$

Обозначим через E *тождественное отношение* $a=b$ в V . Тогда *рефлексивные отношения* определяются условием

$$R \cup E = R, \quad (37)$$

а *антирефлексивные отношения* — условием

$$R \cap E = \emptyset,$$

где \emptyset есть *нулевое отношение*, не выполняющееся ни для каких пар вершин. Отношение будет *ациклическим*, если

$$R^n \cap E = \emptyset, \quad n = 1, 2, \dots \quad (38)$$

Таким образом, *строгое частичное упорядочение* определяется условиями (36) и (38), а *отношение эквивалентности* характеризуется условиями $R = R^*$ и (36), (37). *Транзитивное замыкание* отношения R есть объединение

$$\Gamma(R) = R \cup R^2 \cup R^3 \cup \dots, \quad (39)$$

при котором

$$a\Gamma(R)b$$

выполняется тогда и только тогда, когда существует ориентированная цепь из R -ребер от a к b .

Пусть R — отношение между элементами V и V' , а S — отношение между элементами V' и V . Произведения

$$RS, \quad SR \quad (40)$$

являются соответственно отношениями в V и в V' . Для их транзитивных замыканий

$$\Gamma(RS), \quad \Gamma(SR)$$

будет выполнено

$$a\Gamma(RS)b, \quad a'\Gamma(SR)b',$$

если существует четная цепь от a к b , состоящая поочередно из ребер в R и в S , соответственно четная цепь от a' к b' из S - и R -ребер.

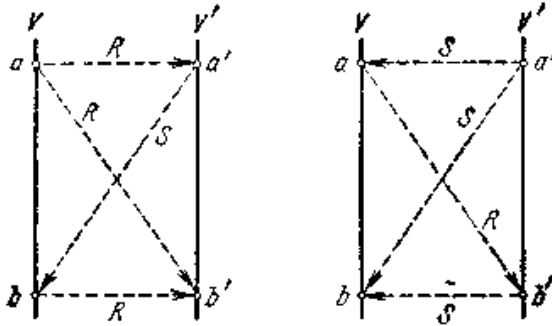


Рис. 1.

Два отношения R и S будут называться *взаимно транзитивными*, если

$$RSR \subseteq R, \quad SRS \subseteq S. \quad (41)$$

Это — условия четырехугольника для ребер из R и S (рис. 1). Если существуют ребра

$$E_1 = (a, a') < R, \quad E_3 = (b, b') < R, \quad E_2 = (a', b) < S,$$

то существует и ребро $E_4 = (a, b')$ в R , и аналогично для S . Если отношения R и S взаимно транзитивны, то отношения R^* и S^* также взаимно транзитивны. Каждое из условий (41) обеспечивает транзитивность обоих произведений отношения (40). Если $V = V'$ и $R = S$, то условия (41) превращаются в

$$R^3 \subseteq R.$$

Это условие *слабой транзитивности*, по-видимому, является естественной заменой транзитивности в двудольных графах. Такие отношения изучал Рэйнич.

Назовем отношение *самотранзитивным*, если R и R^* взаимно транзитивны. В этом случае условия (41) также сводятся к одному:

$$RR^*R \subseteq R. \quad (42)$$

Такие взаимно транзитивны отношения изучали, в частности, Жакотен-Дюбрейль и Риге; последний обозначал их термином «*бифункциональные отношения*». Если выполнено (42), то отношения R^*R и RR^* оба транзитивны и симметричны, и будет

$$aR^*R^*a,$$

если существует R -ребро от каждого a к каждому a' . Сказанное приводит к основному результату для самотранзитивных отношений:

Теорема 20. *Все самотранзитивные отношения получаются следующей конструкцией: рассматриваются такие разбиения основных множеств V и V' :*

$$V = \cup B_i, \quad V' = \cup B'_i,$$

что кардинальное число классов B_i и B'_i одно и то же, и в каждой паре соответствующих классов B_i и B'_i каждая вершина $b_i \in B_i$ соединяется R -ребром с каждой $b'_i \in B'_i$ и наоборот.

Таким образом, эти отношения можно рассматривать как соответствия между двумя отношениями эквивалентности.

4.8. Отношения Феррерса

Пусть R и S — отношения между V и V' . Построим *чередующиеся произведения*

$$\begin{aligned} \Delta(R, S) &= (R \cap \bar{S}) \cdot (S^* \cap \bar{R}^*), \\ \Delta(S^*, R^*) &= (S^* \cap \bar{R}^*) \cdot (R \cap \bar{S}), \end{aligned} \quad (43)$$

где, как и прежде, \bar{R} есть дополнительное отношение для R . Очевидно, $\Delta(R, S)$ — отношение в V , а $\Delta(R^*, S^*)$ — отношение в V' . Кроме того,

$$a \Delta(R, S) b$$

выполняется тогда и только тогда, когда существует такое $a' \in V'$, что

$$a (R \cap \bar{S}) a', \quad a' (S^* \cap \bar{R}^*) b,$$

т. е. когда существуют ребро (a, a') из R , не принадлежащее S , и ребро (a', b) из S^* , не принадлежащее R^* .

Транзитивное замыкание $\Gamma(\Delta(R, S))$ состоит из вершин в V , которые могут быть достигнуты из вершины a четными чередующимися цепями с ребрами, принадлежащими двум отношениям:

$$R \cap \bar{S}, \quad S^* \cap \bar{R}^* = (S \cap \bar{R})^*. \quad (44)$$

Вершины $a' \in V'$, которые могут быть достигнуты из a нечетными чередующимися цепями такого рода, удовлетворяют соотношению

$$a \Gamma(R, S) \cdot R \cap \bar{S} a'.$$

Если отношения (44) взаимно транзитивны, то выполняются условия:

$$\begin{aligned} (R \cap \bar{S}) \cdot (S \cap \bar{R})^* \cdot (R \cap \bar{S}) &\subset R \cap \bar{S}, \\ (S \cap \bar{R})^* \cdot (R \cap \bar{S}) \cdot (S \cap \bar{R})^* &\subset (S \cap \bar{R})^*. \end{aligned} \quad (45)$$

Представляет некоторый интерес особый случай $S = \bar{R}$. Условия (45) сводятся к

$$R \bar{R}^* R \subset R, \quad \bar{R}^* R \bar{R}^* \subset \bar{R}^*, \quad (46)$$

т. е, отношения R и \bar{R}^* взаимно транзитивны. Отношения с этим особым свойством известны под названием *отношений Феррерса*; их изучали Винер и Риге. Можно проверить, что если одно из отношений

$$R, \bar{R}, R^*, \bar{R}^*$$

есть отношение Феррерса, то таковыми же являются и остальные три, и условия (46) сводятся к одному условию, если только R и R^* определены на целых множествах V и V' . Эти отношения можно охарактеризовать другим способом:

Теорема 21. *Отношение R является отношением Феррерса тогда и только тогда, когда соотнесенные множества $R(a)$ образуют упорядоченное по включению семейство.*

Доказательство. Если R есть отношение Феррерса, для которого

$$aRa', b\bar{R}a',$$

то

$$a' \in R(a), a' \notin R(b), \tag{47}$$

и условие (46) показывает, что

$$R(a) \supset R(b). \tag{48}$$

Обратно, если множества $R(a)$ образуют упорядоченное по включению семейство и выполняется (47), то выполнены и (48), (46).

Отношения Феррерса появились впервые в связи с аддитивными разбиениями целых чисел. Предположим, что для некоторого целого N мы имеем

$$N = \sum r_i, \quad r_1 \geq r_2 \geq \dots \geq r_m.$$

Это разбиение может быть представлено при помощи матрицы инцидентий, или *диаграммы Феррерса*

$$\begin{array}{c}
 r_1^* \quad r_2^* \quad \dots \\
 \begin{array}{c} r_1 \\ r_2 \\ \vdots \\ \vdots \\ \vdots \end{array} \left| \begin{array}{cccccc} 1 & 1 & 1 & 0 & \dots & 0 \\ 1 & 1 & 0 & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & 0 \end{array} \right.
 \end{array}$$

где в i -й строке первые n элементов равны 1, а остальные 0. Тогда суммы по столбцам также образуют невозрастающую последовательность

$$r_1^* \geq r_2^* \geq \dots,$$

определяющую двойственное, или обратное разбиение

$$N = \sum r_i.$$

Кроме уже приведенных условий изучались также многие другие формальные условия. Говорят, что два отношения *коммутируют*, если

$$R \cdot S = S \cdot R.$$

Это означает, что если

$$aRb, bSc,$$

то существует некоторое b' , для которого

$$aSb', b'Re,$$

и наоборот. *Слабая коммутативность* определяется условием *шестиугольника Дюбрейля*

$$RSR = SRS.$$

Более топологический характер носит T_0 -условие: из $R(a)=R(b)$ следует $a = b$.

4.9. Матричный анализ связей в частично ориентированных графах

Ниже даётся матричный метод анализа связей в частично ориентированных графах с применением к комбинаторике, анализу релейно-контактных схем и исследованию разложимости матриц, в том числе стохастических. Предварительно вводятся некоторые алгебраические понятия.

1. Множество A вместе с операциями $x+y$ и $x \cdot y$ на нём мы назовём *нумероидом* (от латинского *numerus* -число; в понятии нумероида обобщаются некоторые свойства натуральных чисел), если $x+y$ и $x \cdot y$ ассоциативны и коммутативны, $x \cdot y$ дистрибутивна относительно $x + y$ и в A существует такой единственный элемент 0 , что $0+x=x+0=x$, и такой единственный элемент 1 , что $1 \cdot x = x \cdot 1 = x$. Примеры нумероидов: булева алгебра, множество натуральных чисел (включая нуль), поле. Пусть $\|a_{ij}\|$ есть p^2 -матрица над нумероидом. Матрицу, получаемую из неё вычёркиванием k -й строки и l -го столбца, мы обозначим через $\|a_{ij} \nearrow_{kl}\|$ не меняя нумерации оставшихся рядов. *Квазиминомом* $\|a_{ij} \nearrow_{kl}\|$ где $k \neq l$, матрицы $\|a_{ij}\|$ назовём сумму $\sum a_{ki} a_{i_1 i_2} \dots a_{i_r l}$, где суммирование производится по всем размещениям

$$i_1 i_2 \dots i_r \quad (r = 0, 1, \dots, p - 2)$$

из номеров рядов матрицы $\| a_{ij} \rangle_{kk} \rangle_{ll} \|$; при $k = l$ пусть $| a_{ij} \rangle_{kk} |_{kl} = a_{kk}$.

Теорема 1. Если $k \neq l$, то $| a_{ij} \rangle_{kk} |_{kl} = \sum_{m \neq k} a_{km} A_{km}^{(l)}$, где $A_{km}^{(l)} = 1$

при $m = l$ и

$$A_{km}^{(l)} = | a_{ij} \rangle_{kk} \rangle_{km} |_{ml} |$$

при $m \neq l$.

В самом деле, по определению $| a_{ij} \rangle_{kk} |_{kl} = \sum_{i_1, i_2, \dots, i_r} a_{ki_1} a_{i_1 i_2} \dots a_{i_r l}$. Правую

часть этого равенства можно представить в виде

$$a_{kl} + \sum_{i_1 \neq k, l} a_{ki_1} \sum_{i_2, \dots, i_r} a_{i_1 i_2} \dots a_{i_r l}$$

или, заменив i_1 на m , в виде $a_{kl} + \sum_{m \neq k, l} a_{km} \sum_{i_2, \dots, i_r} a_{mi_2} \dots a_{i_r l}$. Но

вследствие способа суммирования, предусмотренного определением квазиминора,

$$\sum_{i_2, \dots, i_r} a_{mi_2} \dots a_{i_r l} = | a_{ij} \rangle_{kk} \rangle_{lm} |_{ml} = | a_{ij} \rangle_{kk} \rangle_{km} |_{ml}$$

что и доказывает теорему. Теорема 1 позволяет вычислять квазиминоры разложением их на меньшие квазиминоры. Так, вычислим квазиминоры

$| a_{ij} \rangle_{43} |_{34}$ для матрицы

$$\| a_{ij} \| = \left\| \begin{array}{ccccc} 0 & 1 & 1 & 1 & 0 \\ 2 & 0 & 3 & 1 & 1 \\ 0 & 3 & 0 & 0 & 2 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \end{array} \right\| \quad (1)$$

над натуральными числами. Номера её рядов пишем по краям квазиминоров. По теореме 1

$$\begin{aligned}
 |a_{ij} \nearrow_{43} |_{34} &= \begin{vmatrix} 1 & 2 & 4 & 5 \\ 1 & 0 & 1 & 1 & 0 \\ 2 & 2 & 0 & 1 & 1 \\ 3 & 0 & 3 & 0 & 2 \\ 5 & 0 & 1 & 0 & 0 \end{vmatrix}_{34} = 3 \begin{vmatrix} 1 & 4 & 5 \\ 1 & 0 & 1 & 0 \\ 2 & 2 & 1 & 1 \\ 5 & 0 & 0 & 0 \end{vmatrix}_{24} + 2 \begin{vmatrix} 1 & 2 & 4 \\ 1 & 0 & 1 & 1 \\ 2 & 2 & 0 & 1 \\ 5 & 0 & 1 & 0 \end{vmatrix}_{54} = \\
 &= 3 \cdot 2 \begin{vmatrix} 4 & 5 \\ 1 & 1 & 0 \\ 5 & 0 & 0 \end{vmatrix}_{14} + 3 + 3 \begin{vmatrix} 1 & 4 \\ 1 & 0 & 1 \\ 5 & 0 & 0 \end{vmatrix}_{54} + 2 \begin{vmatrix} 1 & 4 \\ 1 & 0 & 1 \\ 2 & 2 & 1 \end{vmatrix}_{24} = \\
 &= 3 \cdot 2 + 3 + 2 \cdot 2 \begin{vmatrix} 4 \\ 1 & 1 \end{vmatrix}_{14} + 2 = 15.
 \end{aligned}$$

2. Напомним, что *граф* есть пара множеств U и V , в которой каждому элементу из U поставлены в соответствие два элемента из V ; элементы из U называются *ребрами* графа, элементы из V — *вершинами* графа; вершины, поставленные в соответствие ребру, называются его *концами*. Ребро с совпадающими концами называется *петлёй*. В дальнейшем под ребром понимается лишь ребро с разными концами. Назовём граф *частично ориентированным*, если из двух концов каждого его ребра выделено не более одного конца. Ребро с одним выделенным концом называется *ориентированным* от выделенного конца к невыделенному. Если ни одно ребро графа не ориентировано, то граф называется *неориентированным*, а если все его рёбра ориентированы, то граф называется *ориентированным*.

Пусть kl — ребро с концами k и l . Множество рёбер $i_0i_1, i_1i_2, \dots, i_{r-1}i_r$, где все поршины i_0, \dots, i_r различны и в ребре $i_{h-1}i_h$ не выделен конец i_h , есть *путь из вершины $i=i_0$ в вершину $j=i_r$* , а число r — *длина* пути. При $r = 1$ путь называется *непосредственным*. Пусть вершины конечного частично ориентированного графа произвольно занумерованы, и пусть a_{ij} и \bar{a}_{ij} суть соответственно число непосредственных и число всех путей из вершины i в вершину j , где $i \neq j$. Пусть $a_{ii} = \bar{a}_{ii} = 0$.

Теорема 2. $\bar{a}_{kl} = |a_{ij} \nearrow_{ik}|_{kl}$.

В самом деле, пусть $k \neq l$. Так как через размещение вершин $ki_1i_2 \dots i_r l$ идёт $a_{ki_1} a_{i_1i_2} \dots a_{i_r l}$ путей из k в l , то $\bar{a}_{kl} = \sum a_{ki_1} a_{i_1i_2} \dots a_{i_r l}$, где суммирование производится по всем размещениям $i_1i_2 \dots i_r$ ($r=0, 1, \dots, p-2$) вершин, отличных от k и l . Через p обозначено число вершин в графе. Но эта сумма есть $|a_{ij} \nearrow_{ik}|_{kl}$. При $k = k$ теорема очевидна.

Вычисление $|a_{ij} \nearrow_{ik}|_{kl}$ даёт также для каждого $r < p - 1$ число $\bar{a}_{kl}^{(r)}$ путей длины r из k в l . Оно равно сумме чисел, получаемых на r -м шаге разложения $|a_{ij} \nearrow_{ik}|_{kl}$ по теореме 1. Число $p - 1$ есть максимальная

длина пути в графе с p вершинами. Для графа с $\|a_{ij}\| = (1)$, например, $a^{(1)}_{34}=a_{34}=0$, $a^{(2)}_{34}=3$, $a^{(3)}_{34}=8$, $a^{(4)}_{34}=4$. По индексам членов разложения квазиминора $|a_{ij} \nearrow ik|kl$ легко найти и сами пути. Так, для графа с $\|a_{ij}\| = (1)$ пути из вершины 3 в вершину 4 суть: 3 — 2 — 1 — 4 (6 вариантов), 3 — 2 — 4 (3 варианта), 3 — 2 — 5 — 1 — 4 (4 варианта) и 3 — 5 — 2 — 4 (2 варианта).

3. Теорема 2 позволяет систематически решать различные задачи комбинаторики и теории игр. Пусть, например, из трёх сосудов ёмкостью 8 л, 5 л и 3 л первый полон, а другие пусты. Сколькими способами можно перелить 4 л из первого сосуда во второй, пользуясь только этими тремя сосудами и не прибегая к измерениям? Так как при переливаниях, удовлетворяющих этим условиям, хотя бы один сосуд всегда пуст или полон, то мыслимо лишь 16 наполнений сосудов: 800, 710, 701, 620, 602, 530, 503, 440, 413, 350, 323, 251, 233, 152, 143, 053 (пишем 800 вместо 8 л, 0 л, 0 л и т. д.). Изобразив эти наполнения точками 1,2, ..., 16 и соединяя точку i с

точкой j вектором \overrightarrow{ij} , если наполнение j возникает из наполнения i одним переливанием, мы получим чертёж ориентированного графа. В ребре \overrightarrow{ij} выделяем конец i . Искомое число есть $a_{18} = |a_{ij} \nearrow s1|_{18}$.

Применяя теорему 1, мы найдём, что $a_{18} = 16$ и что единственный кратчайший путь из вершины 1 в вершину 8 есть 1—10—11 — 4 — 5 — 14—15 — 8. Ему отвечает последовательность из семи переливаний 800 - 350 - 323 - 620 - 602 - 152 - 143 - 440. Непосредственное разыскание путей на чертеже довольно затруднительно. Читатель легко воспроизведёт чертёж и проверит сказанное.

4. Пусть b_{ij} — число рёбер между вершинами i и j , где $i \neq j$, и пусть $b_{ij} = 0$. Матрицы $\|a_{ij}\|$ и $\|b_{ij}\|$ вместе определяют граф с точностью до эквивалентности. В неориентированном графе $b_{ij} = a_{ij} = a_{ji}$, а в ориентированном $b_{ij} = a_{ij} + a_{ji}$, так что в этих случаях граф задаётся только матрицей $\|a_{ij}\|$. Нетрудно видеть, что $c_{ij} = a_{ij} + a_{ji} - b_{ij}$ есть число неориентированных рёбер между вершинами i и j .

Множество различных рёбер $i_0i_1, i_1i_2, \dots, i_{r-2}i_{r-1}, i_{r-1}i_0$ с различными вершинами i_0, \dots, i_{r-1} называется *неориентированным контуром длины r* , проходящим через эти рёбра и вершины. Отметив один из концов каждого ребра в неориентированном контуре так, чтобы ни одна вершина не была отмеченным концом двух рёбер сразу, мы получим из неориентированного контура два *ориентированных контура*. Говорят, что ориентация контура совпадает с ориентацией его ребра, если неотмеченный конец ребра не выделен.

Теорема 3. *Через вершину k графа проходит*

$$\sum_{i=1}^p b_{ik} | b_{ij} \wedge_{11} \wedge_{22} \wedge_{\dots} \wedge_{(k-1)(k-1)} \wedge_{ik} |_{k1} - \sum_{i=1}^p b_{ik}$$

ориентированных контуров, где p — число вершин в графе.

В самом деле, пусть превращение всех ориентированных рёбер в неориентированные делает из графа G граф G^* и из чисел a_{ij} , a_{ij} числа a_{ij}^* , a_{ij}^* . Тогда всякий ориентированный контур, проходящий в графе G через A , разлагается в графе G^* на непосредственный путь из некоторой вершины l в k и путь из k в l , не содержащий ребра, вошедшего в путь из l в k . Отсюда в графе G через lk идёт a_{lk}^* ($a_{lk}^* - 1$) ориентированных контуров, что ввиду $a_{ij}^* = b_{ij}$ и теоремы 2 доказывает утверждение.

Следствие. Число ориентированных контуров в графе равно

$$\sum_{k=1}^{p-1} \sum_{i=k}^p b_{ik} (| b_{ij} \wedge_{11} \wedge_{22} \wedge_{\dots} \wedge_{(k-1)(k-1)} \wedge_{ik} |_{k1} - 1).$$

В самом деле, разобьём контуры на: 1) проходящие через вершину 1; 2) проходящие через вершину 2, но не через вершину 1; 3) проходящие через вершину 3, но не через вершину 1 и 2; ...; $p - 1$) проходящие через вершину $p - 1$, но не через вершины 1 и 2; ...; $p - 2$. Граф без вершин 1, 2, ..., $k - 1$, где $1 \leq k \leq p - 1$, описывается матрицей

$$\| | b_{ij} \wedge_{11} \wedge_{22} \wedge_{\dots} \wedge_{(k-1)(k-1)} \wedge_{ik} |_{k1} \| ,$$

получаемой из $\|b_{ij}\|$ вычёркиванием строк и столбцов с номерами 1, 2, ..., $k - 1$.

Отсюда по теореме 3 через вершину k , но не через вершины 1, 2, ..., $k - 1$, проходит

$$\sum_{i=k}^p b_{ik} (| b_{ij} \wedge_{11} \wedge_{22} \wedge_{\dots} \wedge_{(k-1)(k-1)} \wedge_{ik} |_{k1} - 1)$$

ориентированных контуров. Суммирование по k даёт приведённую формулу.

Число неориентированных контуров есть половина числа ориентированных. Теорема 3 даёт аналитический критерий принадлежности графов к деревьям, деревьям Хусими и т. п. классам графов. Подсчитывая согласно теореме 3 контуры, проходящие через вершину k , одновременно можно для каждого $r \leq p$ получить число контуров длины r , проходящих через k . Контур длины p называется *гамильтоновой линией* графа. Таким образом, теорема 3 даёт средство подсчёта и отыскания гамильтоновых линий графа.

Вариантом теоремы 3 является

Теорема 4. *Через вершину k графа проходит $\sum_{l=1}^p a_{lk} |a_{lj} \wedge l_k|_{kl} - \sum_{l=1}^p c_{lk}$ ориентированных контуров, у которых ориентация совпадает с ориентацией всех входящих в них ребер.*

Докажем её. Всякий такой контур разлагается на непосредственный путь из некоторой вершины l в l и путь из k в l , не содержащий ребра, вошедшего в путь из l в k . Так как в противоположные пути может войти только неориентированное ребро, то через lk идёт

$a_{lk}a_{kl} - c_{lk}$ контуров. Применение теоремы 2 завершает доказательство.

5. Коэффициентом непосредственной связи вершин i и j частично ориентированного графа ($i \neq j$) назовём булев элемент δ_{ij} , равный булевой единице 1 при $a_{ij} > 0$ и булеву нулю 0 при $a_{ij} = 0$. Пусть $\delta_{ii} = 1$. Булева матрица $\|\delta_{ij}\|$ есть матрица непосредственных связей графа. Полным коэффициентом связи вершин i и j назовём булев элемент Δ_{ij} , равный 1 при $a_{ij} > 0$ и равный 0 при $a_{ij} = 0$. Пусть $\Delta_{ii} = 1$. Вершина i связана с вершиной j , если $\Delta_{ij} = 1$; вершины i и j сообщаются, если $\Delta_{ij} = \Delta_{ji} = 1$. $\|\Delta_{ij}\|$ есть полная матрица связей графа.

Теорема 5. $\Delta_{kl} = |\delta_{ij} \wedge l_k|_{kl}$.

В самом деле, $|a_{ij} \wedge l_k|_{kl} > 0$ тогда и только тогда, когда хотя бы для одного размещения $i_1 i_2 \dots i_r$, допускаемого определением квазиминора, $a_{k i_1} a_{i_1 i_2} \dots a_{i_{r-1} i_r} > 0$. Но в этом случае $\delta_{k i_1} \delta_{i_1 i_2} \dots \delta_{i_{r-1} i_r} = 1 \cdot 1 \cdot \dots \cdot 1 = 1$ и, следовательно, $|\delta_{ij} \wedge l_k|_{kl} = 1$. Отсюда $\Delta_{kl} = |\delta_{ij} \wedge l_k|_{kl}$ ввиду теоремы 2.

Отношение сообщения вершин рефлексивно, симметрично и транзитивно. Множество вершин распадается на такие классы K_1, \dots, K_r , что вершины одного класса сообщаются, а вершины разных классов не сообщаются, притом либо все вершины в K_i связаны со всеми вершинами K_j , либо все вершины в K_i связаны со всеми вершинами в K_j , либо K_i и K_j изолированы друг от друга. Для неориентированных графов классы суть (максимальные) связные части. Множество классов K_i и множество упорядоченных пар (K_i, K_j) , где K_i связан с K_j , образуют ориентированный граф, который мы назовём *каркасом* исходного графа. Каркас эквивалентен своему каркасу. Каркас частично ориентированного графа Γ характеризует общую картину связей в Γ . В частности, по каркасу легко решать различные вопросы, касающиеся базисов вершин графа.

Для построения каркаса достаточно вычислить $\|\Delta_{ij}\|$ и $\|\Delta_{ij} \Delta_{ji}\|$. i -й ряд в $\|\Delta_{ij} \Delta_{ji}\|$ даёт перечень элементов класса, к которому принадлежит вершина i . Найдя же из $\|\Delta_{ij} \Delta_{ji}\|$ число и состав классов K_1, \dots, K_r , можно сравнить Δ_{ij} для представителей этих классов. Граф называется *связным*, если все его вершины сообщаются.

Теорема 6. *Граф связан тогда и только тогда, когда*

$$\prod_{k=2}^p \Delta_{1k} \Delta_{k1} = \prod_{k=2}^p |\delta_{ij \nearrow k1} |_{1k} | \delta_{ij \nearrow k1} |_{k1} = 1.$$

В самом деле,

$$\prod_{k=2}^p \Delta_{1k} \Delta_{k1} = 1$$

равносильно $\Delta_{12} = \Delta_{21} = \dots = \Delta_{1p} = \Delta_{p1} = 1.$

6. Задача о числе путей между двумя вершинами частично ориентированного графа встречается при анализе контактных схем с вентилями или без них (кроме настоящих вентиляей (полупроводников), односторонней структурной проводимостью обладают также контакты, присоединённые непосредственно к источнику напряжения.)

Контактные схемы суть графы, рёбрами («ветвями») которых служат контакты и вентили, а вершинами («узлами») — точки соединения контактов и вентиляей. Один из основных методов анализа контактных схем состоит в нахождении всех путей между двумя узлами и выписывании проводимостей этих найденных путей. Однако непосредственное отыскание путей в сложной мостиковой схеме затруднительно. Подсчёт путей может служить средством контроля.

Квазиминоры применимы и для отыскания самих структурных проводимостей. Структурные проводимости суть булевы функции состояния приёмных элементов. Каждое состояние приёмных элементов даёт свою картину непосредственных и полных связей $\|\delta_{ij}\|$, $\|\Delta_{ij}\|$ между узлами схемы. Поэтому непосредственную проводимость f_{ij} между узлами i и j можно рассматривать как переменный коэффициент δ_{ij} , а полную проводимость F_{ij} между i и j - как переменный коэффициент Δ_{ij} . Поэтому ввиду теоремы 5 справедлива

Теорема 7. $F_{kl} = |f_{ij \nearrow k1} |_{kl}$.

Теорема 7 даёт метод анализа контактных схем, представляющий собой алгебраизацию упомянутого выше метода анализа посредством нахождения всех путей. Метод анализа схем вычислением булевых миноров основан на равенстве $|f_{ij \nearrow k1} |_{kl} = |f_{ij \nearrow k1}|$, справедливым для любых нормированных булевых матриц. Метод булевых миноров даёт те же результаты, что и метод квазиминоров, и в конечном счёте также равносильен перебору путей, но при вычислении миноров по обыкновенным правилам обычно появляются лишние, устранимые члены. Булево сложение дистрибутивно относительно булева умножения. Поэтому можно ввести булевы квазиминоры второго рода

и применить их к анализу контактных схем по условиям несрабатывания.)

7. Метод квазиинверсов применим к исследованию условий разложимости и неразложимости матриц над нумероидом. Понятие разложимости, неразложимости и нормального вида матрицы мы заимствуем из [13].

Теорема 8. *Разложимость матрицы $\|u_{ij}\|$ над нумероидом равносильна разложимости булевой матрицы $\|\omega_{ij}\|$, где ω_{ij} — булев элемент, равный 0, если $u_{ij} = 0$, и равный 1, если $u_{ij} \neq 0$.*

В самом деле, разложимость зависит только от расположения нулей в $\|u_{ij}\|$, а оно такое же, как в $\|\omega_{ij}\|$.

Диаграммой булевой p^2 -матрицы $\|\eta_{ij}\|$ назовём ориентированный граф с p вершинами и матрицей непосредственных путей $\|a_{ij}\|$, где $a_{ij} = 1$, если $i \neq j$ и $\eta_{ij} = 1$, и $a_{ij} = 0$ в противном случае. Диаграммой матрицы $\|u_{ij}\|$ над нумероидом назовём диаграмму матрицы $\|\omega_{ij}\|$, описанной в теореме 8.

Теорема 9. *$\|u_{ij}\|$ неразложима тогда и только тогда, когда её диаграмма связна, т. е. когда*

$$\prod_{k=2}^p \varepsilon_{1j \rightarrow k} \varepsilon_{k \rightarrow 1} \varepsilon_{ij \rightarrow k} \varepsilon_{k \rightarrow l} = 1,$$

где $\varepsilon_{ij} = \omega_{ij}$ при $i \neq j$ и $\varepsilon_{ij} = 1$ при $i = j$.

В самом деле, рассмотрим матрицы связей $\|\delta_{ij}\|$ и $\|\Delta_{ij}\|$ для диаграммы матрицы $\|\omega_{ij}\|$. Ввиду $\omega_{ij} = \varepsilon_{ij}$ разложимость $\|\omega_{ij}\|$ равносильна разложимости $\|\delta_{ij}\|$. Пусть $\|\delta_{ij}\|$ разложима и после перестановки рядов содержит в правом верхнем углу нулевую $q \times (p - q)$ -подматрицу. Пусть $k \leq q$ и $l > q$. Так как любое размещение $k i_1 i_2 \dots i_l$ содержит хотя бы одну пару соседних чисел i_h, i_{h+1} из которых $i_h < q$ и $i_{h+1} > q$, то любое произведение $\delta_{k i_1} \delta_{i_1 i_2} \dots \delta_{i_l} = 0$ и, следовательно, $\Delta_{kl} = 0$. Поэтому $\|\Delta_{ij}\|$ также содержит в правом верхнем углу нулевую $q \times (p - q)$ -подматрицу. Значит, диаграмма содержит не менее двух классов вершин и не может быть связной. Но перестановка рядов в $\|\delta_{ij}\|$ была равносильна перенумерации вершин диаграммы и не повлияла на её связность. Тем самым теорема доказана в одну сторону. Но если диаграмма несвязна, то она содержит не менее двух классов вершин. Один из них не связан с другим. Пусть в этом первом классе q вершин. Занумеруем эти вершины числами $1, 2, \dots, q$, а прочие вершины числами $q+1, q+2, \dots, p$. Тогда для $k \leq q, l > q$ все коэффициенты Δ_{kl} и тем более δ_{kl} суть нули, т. е. $\|\delta_{ij}\|$ разложима. Это окончательно доказывает теорему.

Следствие. *Нормальный вид матрицы непосредственных связей графа однозначно характеризует его каркас.*

В самом деле, каждое диагональное поле матрицы, приведённой к нормальному виду, отвечает одному из классов K_1, \dots, K_r , а прочие поля задают связи между классами.

Понятие разложимой матрицы ввёл Г. Фробениус при исследовании корней неотрицательной действительной матрицы. Он же дал следующий критерий: для разложимости неотрицательной матрицы A необходимо и достаточно, чтобы один из главных миноров определителя $|\lambda E - A|$ был равен нулю для $\lambda=r$, где r — максимальный корень матрицы A . Однако этот критерий громоздок. Критерий, вытекающий из теоремы 9, проще, так как работает с булевыми элементами.

8. Частным случаем неотрицательных матриц являются стохастические матрицы. Многие важные свойства марковских цепей связаны с разложимостью и неразложимостью стохастических матриц, представляющих законы этих цепей. Таким образом, теорему 9 можно применить к качественному исследованию марковских цепей. Марковская цепь есть ориентированный граф, вершинами которого служат состояния (события), образующие цепь, а рёбрами — возможные (т. е. обладающие положительной вероятностью) переходы между этими состояниями. Этот граф является диаграммой стохастической матрицы, изображающей закон цепи, и для наглядности может быть изображён геометрически. Таким образом, приведённые матрицы переходных вероятностей к нормальному виду равносильно анализу связей графа, образуемого цепью. Это обстоятельство лежит в основе зависимости между концепциями А. Н. Колмогорова и В. И. Романовского при классификации марковских цепей.

Заметим также, что теорема 4 может быть использована для исследования цикличности стохастических матриц.

5. Кратчайшие пути

Пусть дан граф $G = (X, \Gamma)$, дугам которого приписаны веса (стоимости), задаваемые матрицей $C = [c_{ij}]$. Задача о кратчайшем пути состоит в нахождении кратчайшего пути от заданной начальной вершины $s \in X$ до заданной конечной вершины $t \in X$, при условии, что такой путь

существует, т. е. при условии $t \in R(s)$. Здесь $R(s)$ — множество, достижимое из вершины s . Элементы c_{ij} матрицы весов C могут быть положительными, отрицательными или нулями. Единственное ограничение состоит в том, чтобы в G не было циклов с суммарным отрицательным весом. Если такой цикл Φ все же существует и x_i — некоторая его вершина, то, двигаясь от s к x_i , обходя затем Φ достаточно большое число раз и попадая наконец в t , мы получим путь со сколь угодно малым ($\rightarrow -\infty$) весом. Таким образом, в этом случае кратчайшего пути не существует.

Если, с другой стороны, такие циклы существуют, но исключаются из рассмотрения, то нахождение кратчайшего пути (простой цепи) между s и t эквивалентно нахождению в этом графе кратчайшего гамильтонова пути с концевыми вершинами s и t . Это можно усмотреть из следующего факта. Если из каждого элемента c_{ij} матрицы весов C вычесть достаточно большое число L , то получится новая матрица весов $C' = [c'_{ij}]$, все элементы c'_{ij} которой отрицательны. Тогда кратчайший путь от s к t — с исключением отрицательных циклов — необходимо будет гамильтоновым, т. е. проходящим через все другие вершины. Так как вес любого гамильтонова пути с матрицей весов C' равен весу этого пути с матрицей C , но уменьшенному на постоянную величину $(n - 1) \cdot L$, то кратчайший путь (простая цепь) от s к t с матрицей C' будет кратчайшим гамильтоновым путем от s к t при первоначальной матрице C . Задача нахождения кратчайшего гамильтонова пути намного сложнее, чем задача о кратчайшем пути; она обсуждается отдельно. Поэтому мы будем предполагать, что все циклы в G имеют неотрицательный суммарный вес. Отсюда также вытекает, что неориентированные дуги (ребра) графа G не могут иметь отрицательные веса.

Следующие задачи являются непосредственными обобщениями сформулированной выше задачи о кратчайшем пути.

(i) Для заданной начальной вершины s найти кратчайшие пути между s и *всеми* другими вершинами $x_i \in X$.

(ii) Найти кратчайшие пути между *всеми парами* вершин.

В последующих разделах будет показано, что почти все методы, позволяющие решить задачу о кратчайшем (s - t)-пути, дают также (в процессе решения) и все кратчайшие пути от s к x_i ($Ax_i \in X$). Таким образом, они позволяют решить задачу с небольшими дополнительными вычислительными затратами. С другой стороны, задача (i) может быть решена либо n -кратным применением алгоритма задачи (ii), причем на каждом шаге в качестве начальной вершины s берутся различные вершины, либо однократным применением специального алгоритма.

В настоящей главе мы дадим общие алгоритмы решения сформулированных выше задач и частные алгоритмы для случаев, когда все c_{ij} неотрицательны. Эти частные случаи встречаются на практике довольно часто (например, когда c_{ij} являются расстояниями), так что рассмотрение этих специальных алгоритмов оправдано. Мы будем предполагать, что матрица не удовлетворяет, вообще говоря, условию треугольника, т. е. не обязательно $c_{ij} \leq c_{ik} + c_{kj}$ для всех i, j и k . В противном случае кратчайший путь между x_i и x_j состоит из одной единственной дуги (если такая дуга существует) (x_i, x_j) и задача становится тривиальной. Если в графе G дуга (x_i, x_j) отсутствует, то ее вес полагается равным ∞ .

На практике часто требуется найти не только кратчайший путь, но также второй, третий и т. д. кратчайшие пути в графе. Располагая этими результатами, можно решить, какой путь выбрать в качестве наилучшего (указанный подход полезен при использовании таких критериев, которые являются субъективными по своей природе или не могут быть непосредственно включены в алгоритм). Кроме того, второй, третий и т. д. кратчайшие пути можно использовать при анализе «чувствительности» задачи о кратчайшем пути. В этой главе мы опишем алгоритм вычисления K кратчайших простых цепей между двумя заданными вершинами общего графа.

В настоящей главе обсуждаются также задачи нахождения в графах путей с максимальной надежностью и с максимальной пропускной способностью. Эти задачи связаны с задачей о кратчайшем пути, хотя в них характеристика пути (скажем, вес) является не суммой, а некоторой другой функцией характеристик (весов) дуг, образующих путь. Такие задачи можно переформулировать как задачи о кратчайшем пути. Однако можно поступить иначе: непосредственно приспособить для их решения те методы, которые применяются в задачах о кратчайшем пути.

Обсуждается также случай, когда учитываются и пропускные способности, и надежности дуг. Это приводит к задаче о пути с наибольшей ожидаемой пропускной способностью. И хотя такая частная задача не может быть решена при помощи техники отыскания кратчайшего пути, но итерационный алгоритм, использующий эту технику в качестве основного шага, является эффективным методом получения оптимального ответа.

Задачи о кратчайшем пути, в которых на пути накладываются некоторые ограничения, в настоящей главе не рассматриваются, так как к ним непосредственно не применимы те методы расстановки пометок, на которых базируются все описанные здесь алгоритмы нахождения кратчайшего пути. Эти задачи с ограничениями часто

настолько сложны, что соответствующие алгоритмы позволяют находить оптимальные решения лишь таких задач, размеры которых (например, число вершин) на несколько порядков меньше, чем у аналогичных задач без ограничений. Ряд важных задач с ограничениями, например, задача о кратчайшем гамильтоновом пути, рассматривается в отдельных разделах.

5.1. Кратчайший путь между двумя заданными вершинами s и t

Сначала мы приведем очень простой и эффективный алгоритм решения этой задачи для случая $c_{ij} \geq 0$ ($\forall i, j$), а затем распространим описанный метод на общий случай $c_{ij} > 0$ с оговоркой, что циклы с отрицательными весами отсутствуют.

5.1.1. Случай неотрицательной матрицы весов

Наиболее эффективный алгоритм решения задачи о кратчайшем ($s-t$)-пути первоначально дал Дейкстра. В общем случае этот метод основан на приписывании вершинам временных пометок, причем пометка вершины дает верхнюю границу длины пути от s к этой вершине. Эти пометки (их величины) постепенно уменьшаются с помощью некоторой итерационной процедуры, и на каждом шаге итерации точно одна из временных пометок становится постоянной. Последнее указывает на то, что пометка уже не является верхней границей, а дает точную длину кратчайшего пути от s к рассматриваемой вершине. Опишем этот метод подробно.

Алгоритм Дейкстры ($c_{ij} \geq 0$)

Пусть $l(x_i)$ — пометка вершины x_i .

Присвоение начальных значений

Шаг 1. Положить $l(s) = 0$ и считать эту пометку постоянной. Положить $l(x_i) = \infty$ для всех $x_i \neq s$ считать эти пометки временными. Положить $p = s$.

Обновление пометок

Шаг 2. Для всех $x_i \in \Gamma(p)$, пометки которых временные, изменить пометки в соответствии со следующим выражением:

$$l(x_i) \leftarrow \min [l(x_i), l(p) + c(p, x_i)]. \quad (1)$$

Превращение пометки в постоянную

Шаг 3. Среди всех вершин с временными пометками найти такую, для которой $l(x^*_i) = \min [l(x_i)]$.

Шаг 4. Считать пометку вершины x^* постоянной и положить $p = x^*$.

Шаг 5. (i) (Если надо найти лишь путь от s к t .) Если $p = t$, то $l(p)$ является длиной кратчайшего пути.

Останов.

Если $p \neq t$, перейти к шагу 2.

(ii) (Если требуется найти пути от s ко всем остальным вершинам.)

Если все вершины отмечены как постоянные, то эти пометки дают длины кратчайших путей.

Останов.

Если некоторые пометки являются временными, перейти к шагу 2. Доказательство того, что вышеприведенный алгоритм действительно дает кратчайшие пути, чрезвычайно простое, дадим набросок этого доказательства.

Допустим, что на некотором этапе постоянные пометки дают длины кратчайших путей. Пусть S_1 — множество вершин с этими пометками, а S_2 — множество вершин с временными пометками. В конце шага 2 каждой итерации временная пометка $l(x_i)$ дает кратчайший путь от s к x_i , проходящий полностью по вершинам множества S_1 . (Так как при каждой итерации в множество S_1 включается только одна вершина, то обновление пометки $l(x_i)$ требует только одного сравнения на шаге 2.)

Пусть кратчайший путь от s к x^*_i не проходит целиком по S_1 и содержит по крайней мере одну вершину из S_2 , и пусть $x_j \in S_2$ — первая такая вершина в этом пути. Так как по предположению c_{ij} неотрицательны, то часть пути от x_j к x^*_i должна иметь неотрицательный вес Δ и $l(x_j) < l(x^*_i) - \Delta < \Delta(x^*_i)$. Это, однако, противоречит утверждению, что $l(x^*_i)$ — наименьшая временная пометка, и, следовательно, кратчайший путь к x^*_i проходит полностью по вершинам множества S_1 , и поэтому $l(x^*_i)$ является его длиной.

Так как вначале множество S_1 равно $\{s\}$ и при каждой итерации к S_1 добавляется x^*_i , то предположение, что $l(x_i)$ равно длине кратчайшего пути $\forall x_i \in S_1$, выполняется при каждой итерации. Отсюда по индукции следует, что алгоритм дает оптимальный ответ.

Если требуется найти кратчайшие пути между s и всеми другими вершинами полного связного графа с n вершинами, то в процессе работы алгоритма выполняются $n(n - 1)/2$ операций сложения и сравнения на шаге 2 и еще $n(n - 1)/2$ операций сравнения на шаге 3. Кроме того, при осуществлении шагов 2 и 3 необходимо определить, какие вершины являются временными, а для этого нужно еще $n(n - 1)/2$ операций сравнения. Эти величины являются верхними границами для числа операций, необходимых при отыскании кратчайшего пути между заданными вершинами s и t . Они действительно достигаются, если окажется, что вершина t будет

последней вершиной, получившей постоянную пометку. (Джонсон предложил так называемый метод сортировки, позволяющий уменьшить число операций на шаге 3.)

Как только длины кратчайших путей от s будут найдены (они будут заключительными значениями пометок вершин), сами пути можно получить при помощи рекурсивной процедуры с использованием соотношения (2). Так как вершина x'_i непосредственно предшествует вершине x_i в кратчайшем пути от s к x_i , то для любой вершины x_i соответствующую вершину x'_i можно найти как одну из оставшихся вершин, для которой

$$l(x'_i) + c(x'_i, x_i) = l(x_i). \quad (2)$$

Если кратчайший путь от s до любой вершины x_i является «единственным, то дуги (x'_i, x_i) этого кратчайшего пути образуют ориентированное дерево с корнем s . Если существует несколько «кратчайших» путей от s к какой-либо другой вершине, то при некоторой фиксированной вершине x'_i соотношение (2) будет выполняться для более чем одной вершины x_i . В этом случае выбор может быть либо произвольным (если нужен какой-то один кратчайший путь между s и x_i), либо таким, что рассматриваются все дуги (x'_i, x_i) , входящие в какой-либо из кратчайших путей, и при этом совокупность всех таких дуг образует не ориентированное дерево, а общий граф, называемый *базой относительно s* или кратко — *s -базой*

Пример. Рассмотрим граф, изображенный на рис. 1, где каждое неориентированное ребро рассматривается как пара противоположно ориентированных дуг равного веса.

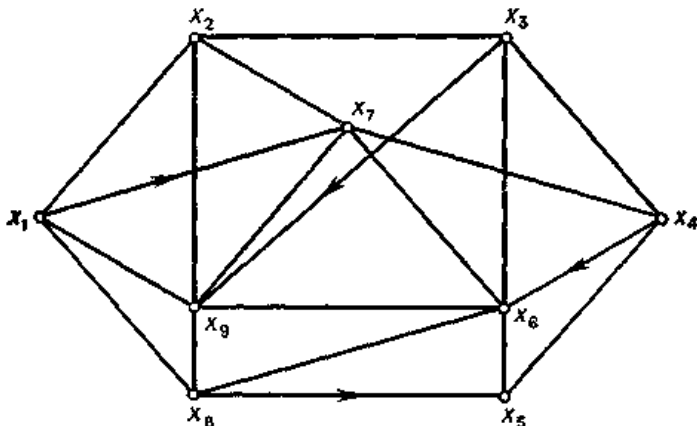


Рис. 1. Граф из примера.

Матрица весов C_1 приведена ниже.

	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9
x_1		10					3	6	12
x_2	10		18				2		13
x_3		18		25		20			7
x_4			25		5	16	4		
$C_1 = x_5$				5		10			
x_6			20		10		14	15	9
x_7		2		4		14			24
x_8	6				23	15			5
x_9	12	13				9	24	5	

Требуется найти все кратчайшие пути от вершины x_1 ко всем остальным вершинам. Мы воспользуемся алгоритмом Дейкстры. Постоянные пометки будем снабжать анаком +, остальные пометки рассматриваются как временные.

Алгоритм работает так:

Шаг 1. $l(x_1) = 0^+$, $l(x_i) = \infty \forall x_i \neq x_1, p = x_1$.

Первая итерация

Шаг 2. $\Gamma(p) = \Gamma(x_1) = \{x_2, x_7, x_8, x_9\}$ — все пометки временные. Возьмем сначала x_2 . Из (1) получаем

$$l(x_2) = \min[\infty, 0^+ + 10] = 10,$$

аналогично $l(x_7) = 3, l(x_8) = 6, l(x_9) = 12$.

Шаг 3. $\min(\underbrace{10}_{x_2}, \underbrace{3}_{x_7}, \underbrace{6}_{x_8}, \underbrace{12}_{x_9}, \underbrace{\infty}_{x_3, x_4, x_5, x_6}) = 3$ соответствует x_7 .

Шаг 4. x_2 получает постоянную пометку $l(x_7) = 3^+$, $p = x_7$.

Шаг 5. Не все вершины имеют постоянные пометки, поэтому переходим к шагу 2. Пометки в начале следующей итерации показаны на рис. 2 (а).

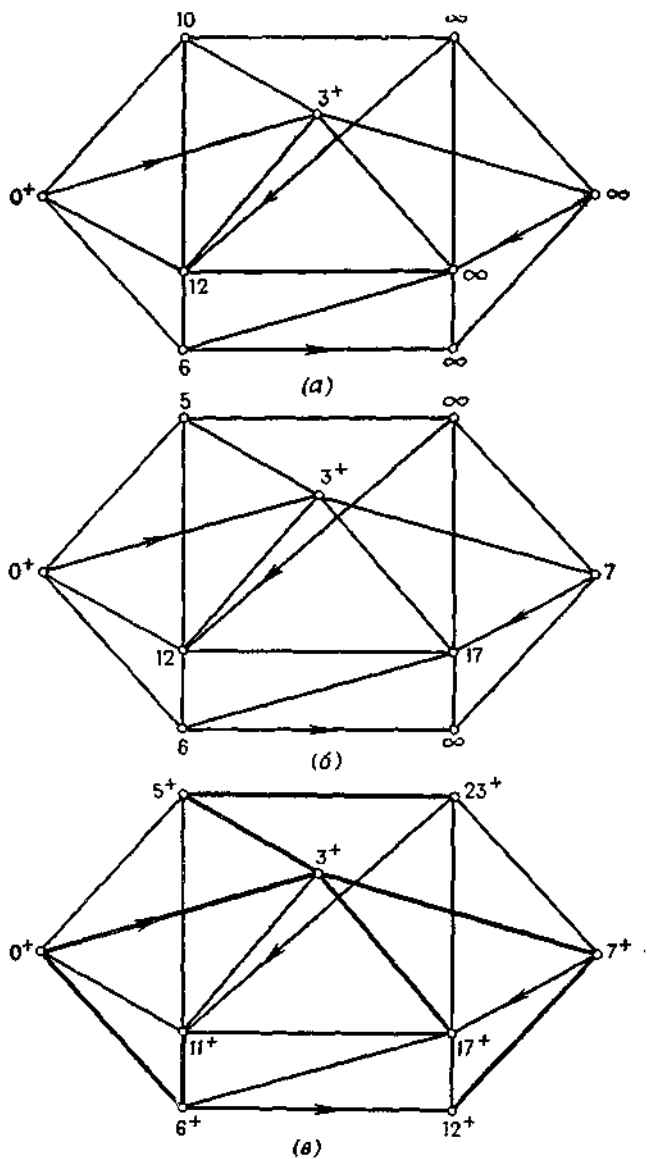


Рис. 2 (а) Пометки в конце 1-й итерации. (б) Пометки в конце шага 2 на 2-й итерации, (в) Окончательные пометки вершин и x_j -база.

Вторая итерация

Шаг 2. $\Gamma(p) = \Gamma(x_7) = \{x_2, x_4, x_6, x_8\}$ — все пометки временные.

Из соотношения (1) имеем

$$l(x_2) = \min [10, 3^+ + 2] = 5,$$

аналогично $l(x_4) = 7, l(x_6) = 17, l(x_8) = 12$. Пометки изображены на рис. 2 (б).

Шаг 3. $\min \left[\underbrace{5}_{x_2}, \underbrace{7}_{x_4}, \underbrace{17}_{x_6}, \underbrace{6}_{x_8}, \underbrace{12}_{x_9}, \underbrace{\infty}_{x_3, x_5} \right] = 5$ соответствует x_2 .

Шаг 4. x_2 получает постоянную пометку

$$\boxed{l(x_2) = 5^+}, p = x_2.$$

Шаг 5. Перейти к шагу 2.

Третья итерация

Шаг 2. $\Gamma(p) = \Gamma(x_2) = \{x_1, x_3, x_7, x_9\}$ — только вершины x_3 и x_9 имеют временные пометки; из соотношения (1) получаем

$$l(x_3) = \min [\infty, 5^+ + 18] = 23$$

и аналогично

$$l(x_9) = 12.$$

Шаг 3. $\min \left[\underbrace{23}_{x_3}, \underbrace{7}_{x_4}, \underbrace{17}_{x_6}, \underbrace{6}_{x_8}, \underbrace{12}_{x_9}, \underbrace{\infty}_{x_5} \right] = 6$ соответствует x_8 .

Шаг 4. x_8 получает постоянную пометку $\boxed{l(x_8) = 6^+}$, $p = x_8$

Шаг 5. Перейти к шагу 2.

Продолжая этот процесс, получим окончательную картину расстановки пометок, изображенную на рис. 2 (в). Для нахождения кратчайшего пути между вершиной x_2 (например) и начальной вершиной x_1 мы последовательно используем соотношение (2). Таким образом, полагая $x_i = x_2$, находим вершину x'_2 , непосредственно предшествующую x_2 в кратчайшем пути от x_1 к x_2 : вершина x'_2 должна удовлетворять соотношению

$$l(x'_2) + c(x'_2, x_2) = l(x_2) = 5.$$

Единственной такой вершиной является x_7 . Далее, применяем второй раз соотношение (2), беря $x_i = x_7$; получаем вершину x'_7 , непосредственно предшествующую x_7 в кратчайшем пути от x_1 к x_2 . Вершина x'_7 удовлетворяет соотношению

$$l(x'_7) + c(x'_7, x_7) = l(x_7) = 3.$$

Единственной такой вершиной является x_1 , и поэтому кратчайший путь от x_1 к x_2 есть (x_1, x_7, x_2) . x_1 -база, дающая все кратчайшие пути от x_1 ,

представляет собой дерево, изображенное жирными линиями на рис. 2 (в).

5.1.2. Случай общей матрицы весов

Только что описанный алгоритм Дейкстры применим лишь в том случае, когда $c_{ij} \geq 0$ для всех i и j . Однако если матрица C является матрицей стоимостей, то дуги, приносящие доход, должны иметь отрицательные «стоимости». В этом случае для нахождения кратчайших путей между вершиной s и всеми другими вершинами можно воспользоваться описанной ниже процедурой. Этот метод также является итерационным и основан на пометках вершин, причем в конце k -й итерации пометки равны длинам тех кратчайших путей (от s ко всем остальным вершинам), которые содержат не более $k + 1$ дуг. В отличие от алгоритма Дейкстры никакая из пометок во время этого процесса не рассматривается как окончательная. Описываемый метод был первоначально предложен в середине 50-х годов Фордом, Муром и Беллманом. Перейдем к его описанию.

Алгоритм для общей матрицы весов

Пусть $l^k(x_i)$ — пометка вершины x_i в конце $(k + 1)$ -й итерации.

Присвоение начальных знаний

Шаг 1 Положим $S = \Gamma(s)$, $k = 1$, $l^1(s) = 0$, $l^1(x_i) = c(s, x_i)$ для всех $x_i \in \Gamma(s)$ и $l^1(x_i) = \infty$ для всех остальных x_i .

Обновление пометок

Шаг 2. Для каждой вершины $x_i \in \Gamma(s)$ ($x_i \neq s$) изменить ее пометку следующим образом:

$$l^{k+1}(x_i) = \min \{ l^k(x_i), \min_{x_j \in T_i} \{ l^k(x_j) + c(x_j, x_i) \} \}, \quad (3)$$

где $T_i = \Gamma^{-1}(x_i) \cap S$. (Множество S содержит теперь все вершины, для которых кратчайшие пути из s состоят из k дуг.)

Множество T_i содержит те вершины, для которых текущие кратчайшие пути из s состоят из k дуг (т. е. те, вершины которых лежат в S) и для которых существуют дуги к вершине x_i . Отметим, что если $x_i \notin \Gamma(S)$, то кратчайший путь от s к x_i не может состоять из $k + 1$ дуг и поэтому изменять пометку вершины x_i не нужно. Для вершин $x_i \in \Gamma(S)$ положим $l^{k+1}(x_i) = l^k(x_i)$.

Проверка на окончание

Шаг 3. (а) Если $k \leq n - 1$ и $l^{k+1}(x_i) = l^k(x_i)$ для всех x_i , то получен оптимальный ответ и пометки равны длинам кратчайших путей. Останов.

(б) Если $k < n - 1$ и $l^{k+1}(x_i) \neq l^k(x_i)$ для некоторой вершины x_i , то перейти к шагу 4.

(в) Если $k = n - 1$ и $l^{k+1}(x_i) \neq l^k(x_i)$ для некоторой вершины x_i , то в графе существует цикл отрицательного веса и задача не имеет решения. Останов.

Подготовка к следующей итерации

Шаг 4. Обновить множество следующим образом:

$$S = \{x_i \mid l^{k+1}(x_i) \neq l^k(x_i)\}. \quad (4)$$

(Новое множество S содержит теперь все вершины, кратчайшие пути до которых из s имеют длину $k + 1$.)

Шаг 5. Положить $k = k + 1$ и перейти к шагу 2.

Как только получены длины кратчайших путей от s к каждой другой вершине, то сами пути опять находятся просто, с помощью соотношения (2). Пути могут быть получены и другим способом, если в дополнение к пометке $l^k(x_i)$ для каждой вершины хранить во время вычислений другую пометку $\theta^k(x_i)$. Пометка $\theta^k(x_i)$ указывает вершину, непосредственно предшествующую вершине x_i в кратчайшем пути от s к x_i во время k -й итерации. Можно начать с $\theta^1(x_i) = s \forall x_i \in \Gamma(s)$ и для всех остальных вершин x_i выбрать $\theta^1(x_i)$ произвольно (скажем, равной 0). Пометки $\theta^k(x_i)$ можно тогда изменять в соответствии с соотношением (3). Таким образом, $\theta^{k+1}(x_i) = \theta^k(x_i)$, если в квадратных скобках в выражении (3) будет наименьшим первый член, или $\theta^{k+1}(x_i) = x_j$, если наименьшим является второй член. Если $\theta(x_i)$ — вектор, составленный из пометок θ при завершении работы алгоритма, то кратчайший путь от s к x_i получается в обратном порядке, а именно $s, \dots, \theta^3(x_i), \theta^2(x_i), \theta(x_i), x_i$, где $\theta^2(x_i)$ является сокращением для $\theta(\theta(x_i))$ и т. д.

Доказательство того, что приведенный алгоритм дает оптимальное решение, весьма простое. Здесь мы его не приводим. Однако заметим, что оно базируется на принципе динамического программирования и том факте, что если не существует никакого оптимального пути из k дуг, то не может также существовать и никакого оптимального пути, содержащего $k + 1$ дуг. Описанный алгоритм можно применять и в случае неотрицательной матрицы весов, хотя это, вообще говоря, намного хуже, чем использование алгоритма Дейкстры. В случае полного связного графа с n вершинами этот алгоритм требует порядка n^3 операций сложения и сравнения, в то время как в алгоритме Дейкстры требуется n^2 операций. Некоторые улучшения описанного алгоритма, предложенные Йеном, позволяют уменьшить число операций в четыре раза, но порядок роста остается равным трем.

Пример. Рассмотрим граф, изображенный на рис. 3, где опять неориентированные ребра рассматриваются как пары противоположно ориентированных дуг с равными весами.

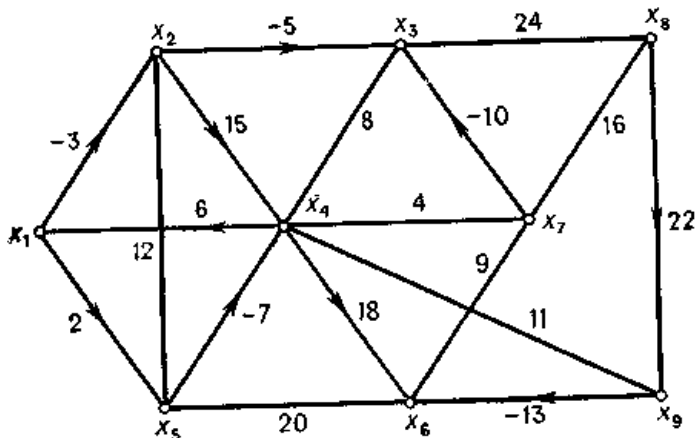


Рис. 3. Граф из примера.

Веса указаны у соответствующих дуг и могут быть как положительными, так и отрицательными числами. Требуется найти кратчайшие пути от x_1 ко всем остальным вершинам, при условии, что граф не содержит циклов отрицательного веса, или найти такой цикл, если он существует.

Алгоритм работает следующим образом

Присвоение начальных значений

Шаг 1. $s = x_1$, $S = \{x_2, x_5\}$, $l^1(x_1) = 0$, $l^1(x_2) = -3$, $l^1(x_5) = 2$, $l^1(x_i) = \infty$ для всех других x_i . Положить $k = 1$.

Первая итерация

Шаг 2. $\Gamma(S) = \{x_2, x_3, x_4, x_5, x_6\}$. Поэтому для x_2 : $T_2 = \{x_1, x_5\} \cap \{x_2, x_5\} = \{x_5\}$ и из соотношения (3) получаем

$$\begin{aligned} l^2(x_2) &= \min [-3, \underbrace{\{l^1(x_5) + c(x_5, x_2)\}}_{x_j=x_5}] = \\ &= \min [-3, (2 + 12)] = \\ &= -3; \end{aligned}$$

для x_3 : $T_3 = \{x_2, x_4, x_7, x_8\} \cap \{x_2, x_5\} = \{x_2\}$,

$$l^2(x_3) = \min [\infty, \underbrace{(-3 - 5)}_{x_j=x_2}] = -8;$$

для x_4 : $T_4 = \{x_2, x_3, x_5, x_7, x_9\} \cap \{x_2, x_5\} = \{x_2, x_5\}$,
 $l^2(x_4) = \min[\infty, \min\{\underbrace{(-3+15)}_{x_j=x_2}, \underbrace{(2-7)}_{x_j=x_5}\}] = -5$;

для x_5 : $T_5 = \{x_1, x_2, x_6\} \cap \{x_2, x_5\} = \{x_2\}$,
 $l^2(x_5) = \min[2, \underbrace{(-3+12)}_{x_j=x_2}] = 2$;

для x_6 : $T_6 = \{x_4, x_5, x_7, x_9\} \cap \{x_2, x_5\} = \{x_5\}$,
 $l^2(x_6) = \min[\infty, \underbrace{(2+20)}_{x_j=x_5}] = 22$.

Пометки $l^2(x_i)$ таковы: $[0 \ -3, -8, -5, 2, 22, \infty, \infty, \infty]$ для $x = x_1, x_2, \dots, x_9$ соответственно.

Шаг 3 (б). Перейти к шагу 4.

Шаг 4. $S = \{x_3, x_4, x_6\}$.

Шаг 5. $k = 2$, перейти к шагу 2.

Вторая итерация

Шаг 2. $\Gamma(S) = \{x_1, x_3, x_4, x_5, x_6, x_7, x_8, x_9\}$;

для x_3 : $T_3 = \{x_2, x_4, x_7, x_8\} \cap \{x_3, x_4, x_6\} = \{x_4\}$,
 $l^3(x_3) = \min[-8, \underbrace{(-5+8)}_{x_j=x_4}] = -8$;

для x_4 : $T_4 = \{x_2, x_3, x_5, x_7, x_9\} \cap \{x_3, x_4, x_6\} = \{x_3\}$, ~
 $l^3(x_4) = \min[-5, \underbrace{(-8+8)}_{x_j=x_3}] = -5$;

для x_5 : $T_5 = \{x_1, x_2, x_6\} \cap \{x_3, x_4, x_6\} = \{x_6\}$,
 $l^3(x_5) = \min[2, \underbrace{(22+20)}_{x_j=x_6}] = 2$;

для x_6 : $T_6 = \{x_4, x_5, x_7, x_9\} \cap \{x_3, x_4, x_6\} = \{x_4\}$,
 $l^3(x_6) = \min[22, \underbrace{(-5+18)}_{x_j=x_4}] = 13$;

для x_7 : $T_7 = \{x_4, x_6, x_8\} \cap \{x_3, x_4, x_6\} = \{x_4, x_6\}$,
 $l^3(x_7) = \min[\infty, \min\{\underbrace{(-5+4)}_{x_j=x_4}, \underbrace{(22+9)}_{x_j=x_6}\}] = -1$;

для x_8 :

$$T_8 = \{x_3, x_7\} \cap \{x_3, x_4, x_6\} = \{x_3\},$$

$$l^3(x_8) = \min[\infty, \underbrace{(-8 + 24)}_{x_j + x_8}] = 16;$$

для x_9 :

$$T_9 = \{x_4, x_6\} \cap \{x_3, x_4, x_6\} = \{x_4\},$$

$$l^3(x_9) = \min[\infty, \underbrace{(-5 + 11)}_{x_j - x_4}] = 6.$$

Пометки $l^3(x_i)$ равны $[0, -3, -8, -5, 2, 13, -1, 16, 6]$ для $x_i = x_1, x_2, \dots, x_9$ соответственно.

Шаг 3 (б). Перейти к шагу 4.

Шаг 4. $S = \{x_6, x_7, x_8, x_9\}$.

Шаг 5. $k = 3$, перейти к шагу 2.

И т. д.

Продолжая этот процесс, получаем результаты, приведенные ниже в краткой форме.

Третья итерация

Шаг 2.

$$\Gamma(S) = \{x_3, x_4, x_5, x_6, x_7, x_8, x_9\},$$

$$T_3 = \{x_7, x_8\}, l^k(x_9) = -11,$$

$$T_4 = \{x_7, x_9\}, l^k(x_4) = -5,$$

$$T_5 = \{x_6\}, l^k(x_5) = 2,$$

$$T_6 = \{x_7, x_9\}, l^k(x_6) = -7,$$

$$T_7 = \{x_6, x_8\}, l^k(x_7) = -1,$$

$$T_8 = \{x_7\}, l^k(x_8) = 15,$$

$$T_9 = \{x_8\}, l^k(x_9) = 6.$$

Вектор пометок $l^4(x_i)$ равен $[0, -3, -11, -5, 2, -7, -1, 15, 6]$

Шаг 4. $S = \{x_3, x_4, x_8\}$.

Четвертая итерация

Шаг 2. $\Gamma(S) = \{x_3, x_4, x_5, x_7, x_8, x_9\}$,

$$T_3 = \{x_8\}, l^5(x_9) = -11,$$

$$T_4 = \{x_3\}, l^5(x_4) = -5,$$

$$T_5 = \{x_6\}, l^5(x_5) = 2,$$

$$T_7 = \{x_6, x_8\}, l^5(x_7) = -1,$$

$$T_8 = \{x_3\}, l^5(x_8) = 13,$$

$$T_9 = \{x_8\}, l^5(x_9) = 6.$$

Вектор пометок $l^5(x_i)$ равен $[0, -3, -11, -5, 2, -1, 13, 6]$.

Шаг 4. $S = \{x_8\}$.

Пятая итерация

$$\begin{aligned} \text{Шаг 2. } \Gamma(S) &= \{x_3, x_7, x_9\}, \\ T_8 &= \{x_8\}, \quad l^6(x_3) = -11, \\ T_7 &= \{x_8\}, \quad l^6(x_7) = -1, \\ T_9 &= \{x_9\}, \quad l^6(x_9) = 6. \end{aligned}$$

Шаг 3 (а). Останов.

Вектор пометок $l^6(x_i)$ совпадает с $l^5(x_i)$, и, следовательно, эти пометки равны длинам кратчайших путей. Сами пути строятся

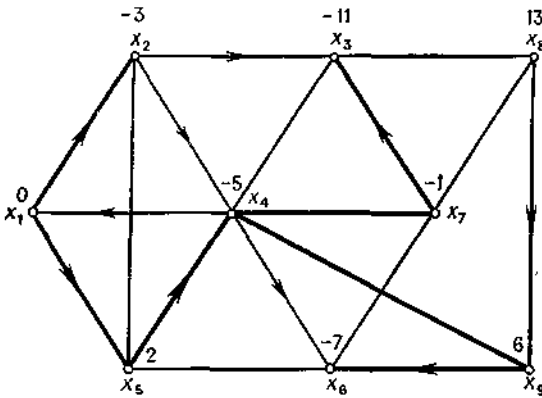


Рис.4. Окончательные пометки вершин и x_1 -база.

последовательно, как и в том случае, когда использовалось соотношение (2). Соответствующая x_1 -база показана жирными линиями на рис. 4.

5.2. Кратчайшие пути между всеми парами вершин

Пусть требуется найти кратчайшие пути между всеми парами вершин графа. Очевидный способ получить ответ на этот вопрос заключается в n -кратном применении алгоритма предыдущего раздела, причем каждый раз в качестве начальной вершины s берутся различные вершины. В случае полного графа с неотрицательной матрицей весов C время, необходимое для вычислений, пропорционально n^3 , а для произвольной матрицы весов оно пропорционально n^4 . Поэтому если задача о кратчайшем пути имеет большую размерность, то ее невозможно решить с помощью последовательного применения алгоритма из разд. 5.1.2.

В настоящем разделе мы опишем иной подход к задаче нахождения кратчайших путей между всеми парами вершин. Этот метод применим как к неотрицательным, так и к произвольным матрицам весов и время, необходимое для вычислений, пропорционально n^3 . Если этот метод применить к графам с неотрицательной матрицей весов, то он экономит почти 50% времени по сравнению с n -кратным применением алгоритма Дейкстры. Метод был предложен первоначально Флойдом и развит Мерчлэндом. Он базируется на использовании последовательности из n преобразований (итераций) начальной матрицы весов C . При этом на k -й итерации матрица представляет длины кратчайших путей между каждой парой вершин с тем ограничением, что путь между x_i и x_j (для любых x_i и x_j) содержит в качестве промежуточных только вершины из множества $\{x_1, x_2, \dots, x_k\}$.

5.2.1. Алгоритм Флойда (для произвольной матрицы весов)

Предположим, что в начальной матрице весов $c_{ii} = 0$ для всех $i = 1, 2, \dots, n$ и $c_{ij} = \infty$, если в графе отсутствует дуга (x_i, x_j) .

Присвоение начальных значений

Шаг 1. Положить $k = 0$.

Итерация

Шаг 2. $k = k + 1$.

Шаг 3. Для всех $i \neq k$, таких, что $c_{ik} \neq \infty$, и для всех $j \neq k$, таких, что $c_{kj} \neq \infty$, введем операцию

$$c_{ij} = \min[c_{ij}, (c_{ik} + c_{kj})]. \quad (5)$$

Проверка на окончание

Шаг 4. (а) Если $c_{ii} < 0$, то в графе G существует цикл отрицательного веса, содержащий вершину x_i , и решения нет. Останов.

(б) Если все $c_{ii} \geq 0$ и $k = n$, то получено решение. Матрица $[c_{ij}]$ дает длины всех кратчайших путей. Останов.

(в) Если все $c_{ii} \geq 0$, но $k < n$, то вернуться к шагу 2.

Доказательство оптимальности ответа, полученного с помощью этого алгоритма, чрезвычайно простое, и мы предоставляем его читателю. Основная операция алгоритма, определяемая соотношением (5), называется *трехместной операцией*. Она имеет разнообразные применения в задачах той же природы, что и задача о кратчайшем пути. Такие задачи обсуждаются в последующих разделах.

Сами кратчайшие пути можно найти по их длинам с помощью рекурсивной процедуры, подобной той, которая выше определялась

соотношением (2). С другой стороны, можно использовать технику, предложенную Ху, для записи информации о самих путях (наряду с информацией о длинах путей). Этот последний метод аналогичен особенно полезен в тех случаях, когда требуется найти в графе цикл отрицательного веса (если такой существует). В этом методе в дополнение к матрице весов C хранится и обновляется вторая $(n \times n)$ -матрица $\Theta = [\theta_{ij}]$. Элемент θ_{ij} указывает вершину, непосредственно предшествующую вершине x_j в кратчайшем пути от x_i к x_j . Матрице Θ присваиваются начальные значения $\theta_{ij} = x_i$ для всех x_i и x_j . В соответствии с (5) на шаге 3 алгоритма обновление матрицы происходит так:

$$\theta_{ij} = \begin{cases} \theta_{ik}, j, & \text{если } (c_{ik} + c_{kj}) < c_{ij} \text{ в квадратных скобках} \\ & \text{в выражении (5),} \\ \text{не изменяется,} & \text{если } c_{ij} \leq (c_{ik} + c_{kj}). \end{cases}$$

В конце алгоритма кратчайшие пути получаются непосредственно из заключительной матрицы Θ . Таким образом, кратчайший путь между двумя вершинами x_i и x_j дается следующей последовательностью вершин:

$$x_i, x_\alpha, \dots, x_\gamma, x_\beta, x_\alpha, x_j,$$

где $x_\alpha = \theta_{ij}$, $x_\beta = \theta_{i\alpha}$, $x_\gamma = \theta_{i\beta}$ и т. д. до $x_i = \theta_{i_j}$.

Здесь следует отметить, что если всем c_{ii} придать начальные значения ∞ (а не 0), то конечное значение величины c_{ii} будет равно весу цепи, проходящей через вершину x_i . Легко также видеть, что, исходя из структуры матрицы Θ , полученной в процессе той итерации, когда элемент c_{ii} становится отрицательным, легко найти цикл отрицательного веса, соответствующий этому элементу.

Это обстоятельство используется дальше при реализации основного шага в алгоритме поиска в графе потока минимальной стоимости. Оно оказывается также полезным и во многих других приложениях (об этом говорится в следующем разделе).

5.3. Обнаружение циклов отрицательного веса

Задача выявления циклов отрицательного веса в произвольном графе важна как сама по себе, так и в качестве основного шага в более сложных алгоритмах. В настоящем разделе эта задача обсуждается несколько более подробно, чем в предыдущем.

В разд. 5.2 отмечалось, что алгоритм Флойда нахождения кратчайших путей между всеми парами вершин может быть использован и для обнаружения в графе циклов отрицательного веса. Кроме того, в тех случаях, когда в графе имеется вершина s , из которой достижимы все остальные вершины этого графа, для нахождения циклов отрицательного веса можно также использовать алгоритм из разд. 5.1.2 (как это указывается на шаге 3(в)). Если не все вершины графа G достижимы из s (например, когда G является неориентированным графом, составленным не менее чем из двух связанных компонент), то алгоритм разд. 5.1.2 завершит работу (как и должно быть) при следующем распределении пометок: у вершин из компоненты, содержащей s , пометки конечные, а пометки вершин в других компонентах равны ∞ . В этом случае циклы отрицательного веса, лежащие в других компонентах, не будут обнаружены. Тем не менее во многих важных приложениях алгоритма обнаружения цикла отрицательного веса всегда имеется в распоряжении вершина s , из которой достижимы все остальные вершины. В таких случаях алгоритм из разд. 5.1.2 при поиске циклов отрицательного веса предпочтительней с вычислительной точки зрения, чем алгоритм Флойда.

Правила остановки алгоритма из разд. 5.1.2 даются в форме, позволяющей минимизировать объем вычислений при поиске кратчайших путей из вершины s , если только в графе G отсутствуют циклы отрицательного веса. Эти правила могут быть видоизменены так, чтобы циклы отрицательного веса обнаруживались по возможности быстрее.

После каждого изменения пометки вершины x_i и нахождения подходящей вершины x_{j^*} , удовлетворяющей соотношению (3), можно проверить, действительно ли x_i принадлежит текущему кратчайшему пути от s к x_{j^*} (этот путь можно найти, используя текущие пометки θ). Если это так, то вершина x_{j^*} была помечена через x_i и из неравенства $\theta(x_{j^*}) + c(x_{j^*}, x_i) < \theta(x_i)$ следует, что часть текущего кратчайшего пути от x_i к x_{j^*} вместе с дугой (x_{j^*}, x_i) должна образовывать цикл отрицательного веса и работа алгоритма может закончиться. Если же пометка вершины x_i либо не изменяется (в соответствии с (3)), либо изменяется, но x_i не будет принадлежать текущему кратчайшему пути от s к x_{j^*} , то работа алгоритма может продолжаться до шага 3(а). Следует заметить, что описанная модификация делает излишним шаг 3(в) алгоритма из разд. 5.1.2, так как теперь цикл отрицательного веса обнаруживается сразу же после его появления, а не в конце всей процедуры.

5.3.1. Оптимальные циклы в графах с двойными весами

Во многих ситуациях возникает следующая задача. Дан граф, дугам которого приписаны два числа — вес и некоторое другое число b_{ij} . Нужно найти такой цикл Φ , для которого целевая функция

$$z(\Phi) = \frac{\sum_{(x_i, x_j) \in \Phi} c_{ij}}{\sum_{(x_i, x_j) \in \Phi} b_{ij}}$$

минимальна (или максимальна).

Рассмотрим, например, обслуживание судном или самолетом некоторой сети маршрутов и предположим, что c_{ij} — «выгода», а b_{ij} — «время», необходимое для прохождения по дуге (x_i, x_j) . Задача нахождения замкнутого обслуживаемого маршрута максимизирующего выгоду, относится к задачам описываемого типа. Другими задачами, которые можно сформулировать как задачи нахождения оптимальных циклов в графах с двойными весами, являются задачи планирования параллельных вычислений и управления технологическими процессами.

Задачу нахождения в графе с двойными весами цикла Φ , для которого отношение $z^*(\Phi)$ минимально, можно решить, используя алгоритм обнаружения в графе цикла отрицательного веса. Это делается так. Допустим, что веса c_{ij} и b_{ij} — произвольные действительные числа (положительные, отрицательные или нуль) с единственным ограничением, что для всех циклов Φ из G

$$\sum_{(x_i, x_j) \in \Phi} b_{ij} > 0.$$

(В большинстве практических ситуаций, например, упомянутых выше, $c_{ij} \geq 0$, но $b_{ij} > 0$ для всех i и j .)

Возьмем некоторое пробное значение z^k целевой функции $z(\Phi)$ и рассмотрим граф G с модифицированными весами

$$c_{ij}^k = c_{ij} - z^k b_{ij}.$$

Попытка найти в G цикл отрицательного веса с матрицей $[c_{ij}^k]$ может привести к следующим трем возможностям.

А. Существует цикл Φ^- отрицательного веса, для которого

$$\sum_{(x_i, x_j) \in \Phi^-} c_{ij}^k < 0.$$

Б. Не существует никакого цикла с отрицательным весом и

$$\sum_{(x_i, x_j) \in \Phi} c_{ij}^k > 0 \text{ для всех циклов } \Phi.$$

В. Существует цикл с нулевым (но не отрицательным) весом, т. е.

$$\sum_{(x_i, x_j) \in \Phi^0} c_{ij}^k = 0 \text{ для некоторого цикла } \Phi^0.$$

В случае А z^* (минимальное значение z) меньше чем z^k , так как неравенство

$$\sum_{(x_i, x_j) \in \Phi^-} c_{ij}^k \equiv \sum_{(x_i, x_j) \in \Phi^-} c_{ij} - z^k \sum_{(x_i, x_j) \in \Phi^-} b_{ij} < 0$$

может выполняться только при

$$\frac{\sum_{(x_i, x_j) \in \Phi^-} c_{ij}}{\sum_{(x_i, x_j) \in \Phi^-} b_{ij}} < z_k,$$

а это, очевидно, означает, что z^* должно быть меньше чем z_k .

Аналогично в случае Б $z^* > z^k$, а в случае В $z^* = z^k$. Все сказанное приводит к следующей процедуре поиска. Начнем с некоторого значения z^1 . Если это значение слишком велико (т. е. имеет место случай А), возьмем $z^2 < z^1$; если же оно слишком мало (т. е. имеет место случай Б), возьмем $z^2 > z^1$. Как только будут найдены верхняя и нижняя границы (соответственно z_u и z_l) для z^* , берем $z^k = (z_u + z_l)/2$ и заменяем в случае А z_u на z^k , а в случае Б — z_l на z^k . Так как число итераций пропорционально числу значащих разрядов требуемой точности, т. е. пропорционально $\log 1/\eta$, где η — величина погрешности, и так как каждая итерация (нахождение отрицательного цикла или вычисление полной матрицы расстояний) требует порядка n^3 операций, то решение сформулированной выше задачи с двойными весами требует $O[n^3 \log 1/\eta]$ операций.

5.4. Нахождение K кратчайших путей между двумя заданными вершинами

В разд. 5.1 были даны методы нахождения в произвольном графе кратчайшего пути от s к t . Но во многих практических приложениях требуется еще, чтобы кратчайший путь обладал некоторыми дополнительными свойствами. Эту задачу можно, конечно, рассматривать как задачу кратчайшего пути с некоторыми дополни-

тельными ограничениями или как многоцелевую задачу, в которой учитываются не только длина пути, но и другие его свойства. Но эти усложнения будут, вообще говоря, сильно увеличивать вычислительную работу, и с практической точки зрения более просто найти K кратчайших путей от s к t и выбрать среди них тот, который обладает нужными свойствами. Хотя такой метод и не эквивалентен прямому рассмотрению дополнительных свойств пути, но он применим даже в тех случаях, когда эти свойства сформулированы не строго или когда они по своей природе субъективны. В этом методе предполагается, что K кратчайших путей между s и t могут быть найдены достаточно эффективно. Именно это будет предметом изучения настоящей главы.

Мы будем здесь предполагать, что рассматриваются только простые цепи. И хотя кратчайший путь необходимо должен быть простой цепью (при условии, что граф не содержит циклов отрицательного веса), но второй, третий и т. д. кратчайшие пути не обязаны быть такими даже в случае, когда все c_{ij} положительны. Задача нахождения K кратчайших путей без требования, что они являются простыми цепями, намного проще, и для ее решения Хоффман и Пэвли, Сакарович, Беллман и Калаба предложили итерационные методы, аналогичные описанным выше. Однако модифицирование этих методов с целью получения простых цепей осуществить не совсем легко, а так как почти во всех практических приложениях алгоритма нахождения K кратчайших путей требуются именно простые цепи, то мы опишем метод, предложенный Йеном и позволяющий находить K кратчайших *простых* цепей.

Пусть $P^k = s, x_2^k, x_3^k, \dots, x_{q_k}^k; t$ — k -й кратчайший путь от S к t , где $x_2^k, x_3^k, \dots, x_{q_k}^k$ — соответственно 2-я, 3-я, ..., q_k -я вершины k -го кратчайшего пути. Пусть P_i^k — «отклонение от пути P^{k-1} в точке i ». Под этим понимается следующее: P_i^k — кратчайший из путей, совпадающих с P^{k-1} от s до i -й вершины, а затем идущий к вершине, отличной от $(i + 1)$ -х вершин тех (ранее уже построенных) кратчайших путей P^j ($j = 1, 2, \dots, k - 1$), которые имеют те же самые начальные подпути от s к i -й вершине, что и P^{k-1} . P_i^k приходит в вершину t по кратчайшему подпути, не проходящему ни через одну из вершин $s, x_2^{k-1}, x_3^{k-1}, \dots, x_i^{k-1}$, участвующих в формировании первой части пути P_i^k . Отсюда следует, что путь P_i^k необходимо должен быть простой цепью.

Первый подпуть $s, x_2^k, x_3^k, \dots, x_i^k$ (совпадающий с $s, x_2^{k-1}, x_3^{k-1}, \dots, x_i^{k-1}$) пути P_i^k называется его корнем и обозна-

чается R^k_i , а второй подпуть x^k_i, \dots, t пути P^k_i называется ответвлением и обозначается через S^k_i .

Алгоритм начинает работу с нахождения P^1 с помощью алгоритма кратчайшего пути от s к t , описанного в разд. 5.1. Этот путь помещается в список L_0 (который должен содержать k -е кратчайшие пути). В общем случае для нахождения P^k нужно уже иметь кратчайшие пути P^1, P^2, \dots, P^{k-1} . Ниже приводится описание алгоритма.

Описание алгоритма

Присвоение начальных значений

Шаг 1. Найти P^1 . Положить $k = 2$. Если существует только один кратчайший путь P^1 , включить его в список L_0 и перейти к шагу 2. Если таких путей несколько, но меньше, чем K , включить один из них в список L_0 , а остальные в список L_1 . Перейти к шагу 2. Если существует K или более кратчайших путей P^1 , то задача решена. Останов.

Нахождение всех отклонений

Шаг 2. Найти все отклонения P^k_i ($k - 1$)-го кратчайшего пути p^{k-1} для всех $i = 1, 2, \dots, q_{k-1}$, выполняя для каждого i шаги с 3-го по 6-й.

Шаг 3. Проверить, совпадает ли подпуть, образованный первыми i вершинами пути P^{k-1} , с подпутем, образованным первыми i вершинами любого из путей P^j ($j = 1, 2, \dots, k - 1$). Если да, положить $c(x_i^{k-1}, x_{i+1}^j) = \infty$; в противном случае ничего не изменять. (При выполнении алгоритма вершина x_l обозначается s). Перейти к шагу 4.

Шаг 4. Используя алгоритм кратчайшего пути, найти кратчайшие пути S^k_i от x_i^{k-1} к t , исключая из рассмотрения вершины s ,

$x_i^{k-1}, x_3^{k-1}, \dots, x_i^{k-1}$. Если существует несколько кратчайших путей, взять в качестве S^k_i один из них.

Шаг 5. Построить P^k_i , соединяя R^k_i ($\equiv s, x^{-1}, x^{-1}, \dots, \dots, x^{-1}$ с S и поместить P в список L_1 .

Шаг 6. Заменить элементы матрицы весов, измененные на шаге, их первоначальными значениями и возвратиться к шагу 3.

Выбор кратчайших отклонений

Шаг 7. Найти кратчайший путь в списке L_1 . Обозначить этот путь P^k и переместить его из L_1 в L_0 . Если $k = K$, то алгоритм заканчивает работу и L_0 дает требуемый список K кратчайших путей. Если $k < K$, положить $k = k + 1$ и вернуться к шагу 2.

Если в L_1 имеется более чем один кратчайший путь (скажем, h путей), то поместить в L_0 любой из них и продолжать, как и выше, до тех пор, пока увеличенное на h число путей, уже находящихся в L_1 , не совпадет с K или не превысит его. Тогда алгоритм завершает работу.

Обоснование вышеприведенного алгоритма основано на очевидном факте: путь P^k должен быть отклонением на i -м этапе (при некотором $i \geq 1$) от одного из более коротких путей P^1, P^2, \dots, P^{k-1} . Поэтому необходимо просто построить все кратчайшие отклонения от каждого из P^j и просмотреть их для нахождения самого короткого отклонения. Это и будет путь p^k . Следует заметить, что при k -й итерации все кратчайшие отклонения для $P^j, j = 1, 2, \dots, k - 2$, уже включены в L_j и нужно найти только одно отклонение от P^{k-1} и пополнить им имеющийся список.

На шаге 3 мы полагаем $c(x_i^{k-1}, x_{k+1}^j) = \infty$ для тех P^j , начальные подпути которых, содержащие i вершин, совпадают с начальным i -вершинным подпутем пути P^{k-1} . Это делается для того, чтобы не получить снова P^j в качестве отклонения (в точке i) от пути P^{k-1} , что вполне могло случиться, если бы мы действовали иначе, так как при $j < k - 1$ вес пути P^j не больше веса пути P^{k-1} .

Хотя на шаге 5 алгоритма каждый порожденный путь P^k_j и помещается в список L_i , совершенно очевидно, что на k -й итерации этот список не должен содержать более $K - k + 1$ кратчайших отклонений P^k_j . С вычислительной точки зрения наиболее сложным является шаг 4, где требуется $O(n^2)$ или $O(n^3)$ операций в зависимости от того, будут ли все $c_{ij} \wedge 0$ или $c_{ij} \cong 0$. Так как этот шаг при k -й итерации выполняется q_k раз и так как $q_k \geq n$, а число итераций равно K , то рассматриваемый алгоритм при нахождении кратчайших путей требует порядка Kn^3 или Kn^4 операций. Первая величина относится к графам с неотрицательными матрицами весов, а вторая — к графам с произвольными матрицами.

Легко видеть, что кратчайшие пути от s к t будут теми же самыми один относительно другого, если все веса c_{ij} заменить на $c'_{ij} = c_{ij} + h_i - h_j$, где величины h — произвольные числа, приписанные вершинам графа. Пользуясь этим фактом, Д. Кнут (в частном сообщении) показал, что, взяв $h_i = d(s, x_i)$, можно всегда получить $c'_{ij} \leq 0$. Так как расстояния $d(s, x_i)$, могут быть получены в общем случае с помощью $O(n^3)$ операций, то на самом деле (преобразуя предварительно веса) и в общем случае можно найти K кратчайших путей, используя $O(Kn^3)$ операций.

5.5. Кратчайший путь между двумя заданными вершинами в ориентированном ациклическом графе

Методы, развитые в предыдущих разделах этой главы, применимы к совершенно произвольным графам. Но на практике задачу кратчайшего пути часто требуется решать для класса ориентированных ациклических графов. Такие графы возникают в методах ПЕРТ и МКП (PERT (Project Evaluation Research Task) и CPM Critical Path Method). Для первого метода в отечественной литературе принят термин СПУ (сетевое планирование и управление). Второй метод называют методом критического пути).

Допустим, что нужно реализовать некий большой проект, и этот проект состоит из большого числа этапов. Мы можем изобразить каждый этап вершиной некоторого графа и построить дугу от вершины x_i к вершине x_j , чтобы показать, что этап i должен предшествовать этапу j . Каждой дуге приписывается некоторый вес c_{ij} , равный минимальной задержке во времени между началом этапа i и началом этапа j . Пусть, например, проект представляет собой процесс строительства здания. Этап i может состоять в строительстве стен, этап j — вставка окон, этап k — прокладка проводов в стенах и т. д. Очевидно, что в этом случае нужно провести дуги от x_i к x_j , от x_i к x_k . Но минимальный срок между началом строительства стен и вставкой окон может быть отличным от срока между строительством стен и прокладкой проводов. Если, например, оконные рамы деревянные и стены должны сначала высохнуть, а для прокладки проводов это не важно, то $c_{ij} > c_{ik}$.

Совершенно очевидно, что рассматриваемый граф будет ориентированным и ациклическим. Действительно, предположение о существовании некоторого цикла, содержащего вершину x_i , приводит к (нелепому для нашей задачи) выводу о возможности повторения этапа i .

В задаче требуется найти минимальное время, необходимое для реализации проекта. Иными словами, нужно найти в графе *самый длинный путь* между вершиной s , изображающей начало, и вершиной t , изображающей завершение всех необходимых для реализации проекта работ. Самый длинный путь называется *критическим* путем, так как этапы, относящиеся к этому пути, определяют полное время реализации проекта и всякая задержка с началом выполнения любого из этих этапов приведет к задержке выполнения проекта в целом.

Сходство этой задачи с задачей о кратчайшем пути между s и t совершенно очевидно и ее можно решить, используя алгоритм из разд. 5.1.1 (так как все $c_{ij} \geq 0$), заменив все операции \min на \max . Но этот алгоритм был бы слишком неэкономичным, так как он не учитывает специальной структуры графа. Можно применить другой алгоритм нахождения самого длинного пути. Этот новый алгоритм описывается ниже именно в терминах самого длинного пути, так как обычно требуется найти как раз такой путь. Задача о кратчайшем пути в ориентированном ациклическом графе также может быть решена с помощью приводимого алгоритма, нужно лишь все операции \max заменить на \min .

5.5.1. Алгоритм нахождения самого длинного (критического) пути в ориентированном ациклическом графе

Пусть вершины пронумерованы так, что дуга (x_i, x_j) всегда ориентирована от вершины x_i к вершине x_j , имеющей больший номер. Для ациклического графа такая нумерация всегда возможна и производится очень легко. При этом начальная вершина получает номер 1, а конечная — номер n .

Присвоим вершине x_j пометку $l(x_j)$, — равную длине самого длинного пути от 1 до x_j , используя для этого соотношения

$$l(x_j) = \max_{x_i \in \Gamma^-(x_j)} [l(x_i) + c_{ij}]; \quad (7)$$

затем присвоим пометку вершине $(x_j + 1)$, применив снова формулу (7), и так далее до тех пор, пока последняя вершина n не получит пометку $l(n)$. В соотношении (7) $l(1)$ полагается равным нулю. Если вершина x_j помечена, то пометки $l(x_i)$ известны для всех вершин $x_i \in \Gamma^-(x_j)$, так как в соответствии со способом нумерации это означает, что $x_i < x_j$ и, следовательно, вершины x_i уже помечены в процессе применения алгоритма.

Пометка $l(n)$ равна длине самого длинного пути от 1 до n . Сами дуги, образующие путь, могут быть найдены обычным способом последовательного возвращения. А именно дуга (x_i, x_j) принадлежит пути тогда и только тогда, когда $l(x_j) = l(x_i) + c_{ij}$. Начиная с вершины x_j равной n , полагаем на каждом шаге x_j равной такой вершине x_i (скажем, x_i^*), для которой выполняется это последнее равенство, и так продолжаем до тех пор, пока не будет достигнута начальная вершина (т. е. пока не будет $x_i^* = 1$).

Совершенно очевидно, что пометка $l(x_j)$ вершины x_j дает длиннейший путь от 1 до x_j , т. е. самое раннее возможное начало выполнения этапа, изображаемого вершиной x_j . Таким образом, если (x_i, x_j) — дуга в графе, то $l(x_j) - l(x_i) - c_{ij}$ (неотрицательная величина, как это видно из (7)) дает самое большое возможное время задержки начала этапа i , не оказывающее влияния на время начала этапа j . Из вышесказанного видно, что если l и j — два последовательные этапа в самом длинном пути, то $j(x_i) - j(x_i) - c_{ij} = 0$.

Пример. Рассмотрим диаграмму ПЕРТ, изображенную на рис. 5 (а), где числа, стоящие у дуг, указывают продолжительность задержек в днях.

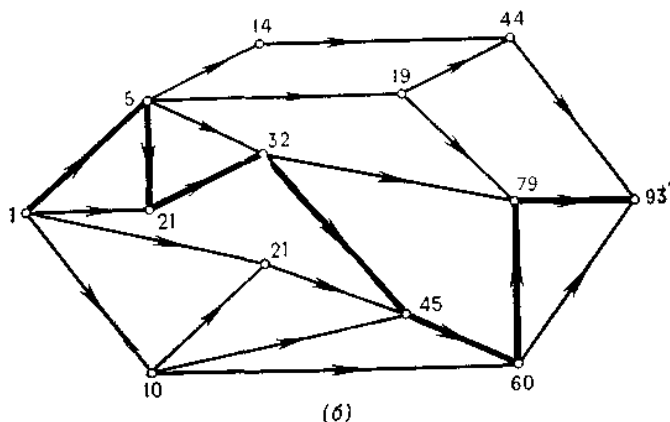
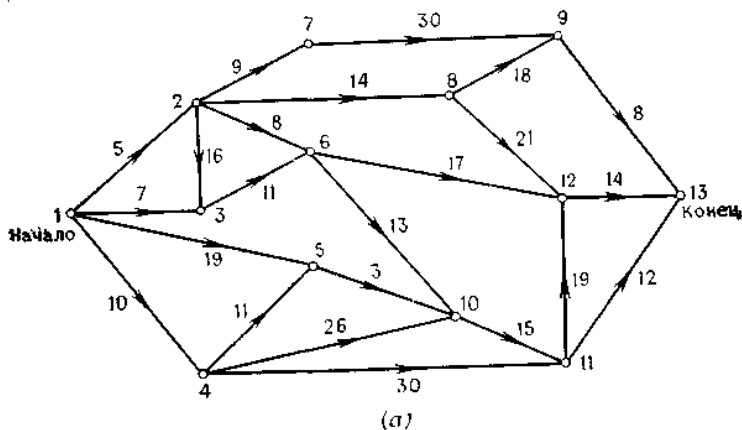


Рис. 5. (а) Граф из примера. (б) Окончательные пометки вершина и критический путь.

Каков наименьший возможный срок c_{ij} выполнения этого проекта? (Предполагается, что вершина 13 является конечной и ей соответствует нулевая продолжительность.) Вершины графа уже пронумерованы так, что дуга (x_i, x_j) существует лишь при $x_j > x_i$. Применение выражения (7) к графу, изображенному на рис. 5 (а), приводит к расстановке пометок, показанной на рис. 5 (б); критический путь изображен жирной линией. Длина этого пути и, следовательно, наименьший возможный срок выполнения проекта равен 93 дням.

Если один из этапов, принадлежащих критическому пути, задерживается на D дней, то и весь проект задержится на тот же срок. С другой стороны, задержка, скажем, этапа 8 (не принадлежащего критическому пути) возможна на $79 - 19 - 21 = 39$ дней, причем это не отразится на общем сроке реализации проекта. Лишь задержка на срок $D > 39$ дней повлияет на пометку вершины 12 (принадлежащую критическому пути).

Метод ПЕРТ описан в этой главе в терминах самого длинного пути в графе, вершины которого отвечают этапам выполнения проекта, а дуги — отношениям предшествования, причем вес c_{ij} Дуги (x_i, x_j) равен интервалу времени между началом этапа i и j . На практике диаграммы ПЕРТ истолковываются несколько иначе, а именно дуги представляют этапы, а вершины изображают абстрактные «события», указывающие начала или окончания этапов. Хотя такое представление само по себе является неполным (так как при этом отношения предшествования точно не описываются), указанный недостаток можно устранить, добавляя фиктивные этапы для точного описания отношений предшествования. С другой стороны, такое представление имеет важное преимущество перед первоначальным, поскольку приводит на практике к более простым диаграммам. Это происходит в силу того, что временная задержка c_{ij} между началами этапов i и j является, вообще говоря, постоянной для данного i и совершенно не зависит от следующего этапа. Редкие исключения всегда можно учесть, вводя некоторые фиктивные этапы. В таких случаях подобное представление приводит на практике к более простым графам (даже с учетом фиктивных вершин), в то время как представление, рассмотренное ранее, остается неизменным. Другой особенностью практического использования метода ПЕРТ является то, что интервалы времени (т. е. временные задержки) между этапами считаются случайными величинами, а не детерминированными, как считалось здесь.

Задачи планирования, решаемые с помощью метода ПЕРТ, не учитывают наличие ресурсов и не рассматривают потребности этапов в

ресурсах. Вместо этого производится (после нахождения, критического пути) такое распределение ресурсов по этапам проекта, при котором задерживается выполнение этапов, не принадлежащих критическому пути. Если же ограничения на ресурсы учитывать на стадии планирования, то задача из очень легкой превращается в очень трудную и решить ее удастся лишь для размерностей на 3—4 порядка меньших, чем в случае соответствующей задачи без ограничений.

5.6. Задачи, близкие к задаче о кратчайшем пути

В алгоритме Флойда нахождения кратчайшего пути между всеми парами вершин была использована трехместная операция (5). Применение этой операции n раз к матрице весов $[c_{ij}]$ дает «заключительную» матрицу, элементы которой равны длинам кратчайших путей. Эта операция является частным случаем более общей трехместной операции

$$z_{ij} = \text{OPT} [z_{ij}, z_{ik} \otimes z_{kj}], \quad (8)$$

где z — функция пути, подлежащая оптимизации, а \otimes — некоторая общая операция. Операция \otimes налагает единственное ограничение в случае ее применения к различным задачам нахождения кратчайших путей в графах. А именно, если путь от x_i к x_j проходит через промежуточную вершину x_k и z — некоторая характеристика пути, на которой базируется оптимизация, то для \otimes должно выполняться условие

$$z_{ij} = z_{ik} \otimes z_{kj}. \quad (9)$$

Ниже приводится несколько примеров использования этой операции.

5.6.1. Наиболее надежный путь

В задаче о кратчайшем пути в качестве «длины» пути бралась сумма весов дуг, образующих этот путь. Рассмотрим теперь случай, когда «вес» дуги представляет ее надежность. Надежность, пути от s к t , составленного из дуг, взятых из множества P , дается формулой

$$\rho(P) = \prod_{(x_i, x_j) \in P} \rho_{ij}, \quad (10).$$

где ρ_{ij} — надежность дуги (x_i, x_j) , т. е. вероятность ее существования в графе (или — в случае физических систем — вероятность того, что она находится в работоспособном состоянии).

Задачу нахождения наиболее надежного пути от s к t можно свести к задаче о кратчайшем пути от s к t , взяв в качестве «веса» c_{ij} дуги (x_i, x_j) величину $c_{ij} = -\log \rho_{ij}$. Прологарифмировав обе части соотношения (10), получим

$$\log \rho(P) = \sum_{(x_i, x_j) \in P} \log \rho_{ij} = - \sum_{(x_i, x_j) \in P} c_{ij}.$$

Отсюда видно, что кратчайший путь от s к t с матрицей весов $[c_{ij}]$ будет в то же время и наиболее надежным путем с матрицей $[\rho_{ij}]$, а надежность этого пути равна антилогарифму его длины. При другой формулировке задачи о наиболее надежном пути возможно непосредственное использование трехместной операции; а именно, операция вводится с помощью соотношения

$$\rho_{ij} = \max [\rho_{ik}, \rho_{kj}].$$

В качестве начальной $[\rho_{ij}]$ -матрицы берется матрица надежностей дуг, причем нулевые элементы указывают на отсутствие соответствующих дуг. Такая формулировка позволяет, очевидно, найти наиболее надежные пути между всеми парами вершин.

5.6.2. Путь с наибольшей пропускной способностью

В этой задаче каждая дуга (x_i, x_j) графа имеет пропускную способность q_{ij} и требуется найти путь от s к t с наибольшей пропускной способностью. Пропускная способность пути P определяется, конечно, дугой из P с наименьшей пропускной способностью, т. е.

$$Q(P) = \min_{(x_i, x_j) \in P} [q_{ij}]. \quad (11)$$

Теорема 1. Пропускная способность пути с наибольшей пропускной способностью от s к t равна

$$\min_K \{ \max_{(x_i, x_j) \in K} [q_{ij}] \},$$

где K — любой $(s-t)$ -разрез (в множестве дуг).

Доказательство. Пусть \hat{Q} — пропускная способность пути с наибольшей пропускной способностью. Так как каждый путь от s к t должен содержать по крайней мере одну дугу из каждого $(s-t)$ -разреза, то для каждого разреза K будет выполняться неравенство

$$\max_{(x_i, x_j) \in K'} [q_{ij}] \leq \hat{Q}. \quad (12)$$

Поскольку в каждом $(s-t)$ -пути должна присутствовать по крайней мере одна дуга с пропускной способностью, не превосходящей \hat{Q} , и поскольку, взяв по одной такой дуге из каждого $(s-t)$ -пути, мы получаем некоторый $(s-t)$ -разрез (по определению), то существует хотя бы один разрез, скажем K' , для которого

$$\max_{(x_i, x_j) \in K'} [q_{ij}] \leq \hat{Q}. \quad (13)$$

Таким образом, разрез \bar{K} , доставляющий минимум выражению

$$\min_K \{ \max_{(x_i, x_j) \in K} [q_{ij}] \}, \quad (14)$$

должен одновременно удовлетворять неравенствам (12) и (13), т. е. для него должно выполняться равенство. Отсюда следует справедливость теоремы.

Очевидный алгоритм нахождения пути с наибольшей пропускной способностью, основанный на теореме 1, состоит в следующем.

Шаг 1. Начать с $(s-t)$ -разреза $\bar{K} \equiv (\{s\}, X - \{s\})$ и найти наибольшую пропускную способность \bar{Q} дуг из \bar{K} .

Шаг 2. Построить остовный подграф $G' = (X, A')$, где

$$A' = \{(x_i, x_j) / q_{ij} \geq \bar{Q}\}.$$

Шаг 3. Найти множество $R'(s)$ вершин, достижимых из s в графе A' .

Шаг 4. Если $t \in R'(s)$, то $\hat{Q} = \bar{Q}$ и любой $(s-t)$ -путь в остовном подграфе G' будет иметь наибольшую пропускную способность, равную \hat{Q} . Если $t \notin R'(s)$, перейти к шагу 5.

Шаг 5. Взять в качестве \bar{K} разрез $(R'(s), X - R'(s))$ и найти наибольшую пропускную способность \bar{Q} дуг в этом новом разрезе.

(Текущее значение величины \bar{Q} меньше, чем предыдущее, в силу определения множества $R'(s)$ на шаге 3 и множества дуг A' на шаге 2.) Возвратиться к шагу 2.

Для неориентированных графов G Франк и Фриш предложили еще более простую процедуру нахождения пути с наибольшей пропускной способностью, состоящую в следующем.

Пусть K_l — некоторый $(s-t)$ -разрез, а \bar{Q} — наибольшая пропускная способность ребер из K_l . Если теперь «закоротить» любое ребро (x_i, x_j) с пропускной способностью $q_{ij} \geq \bar{Q}$, т. е. вершины x_i и x_j заменить одной вершиной x , положив

$$\Gamma(x) = \Gamma(x_i) \cup \Gamma(x_j), \quad \Gamma^{-1}(x) = \Gamma^{-1}(x_i) \cup \Gamma^{-1}(x_j),$$

и ребро (x_i, x_j) удалить, то получившийся граф G_I будет иметь тот же самый путь с наибольшей пропускной способностью, что и первоначальный граф G . Справедливость этого утверждения вытекает из того факта, что \bar{Q} является верхней границей для \hat{Q} (в соответствии с (12)), и, следовательно, ребра с $q_{ij} \geq \bar{Q}$ не могут повлиять на оптимальное решение, т. е. не могут войти в разрез \bar{K} . Поскольку «закорачивание» ребра (x_i, x_j) может повлиять только на разрезы, содержащие (x_i, x_j) (эти разрезы удаляются), то разрез \hat{K} графа G (или все разрезы, доставляющие минимум выражению (14), если их несколько) будет также и разрезом преобразованного графа G_I . Процедуру, примененную к первоначальному графу G , можно повторить и для графа G_I , выбирая другой $(s-t)$ -разрез K_2 , закорачивая все ребра из G_I , пропускная способность которых не меньше наибольшей пропускной способности любого ребра из K_2 . Это дает граф G_2 и т. д. Процесс закончится, когда будут закорочены s и t . Теперь каждый $(s-t)$ -путь в графе \hat{G} , образованный вершинами из G и теми ребрами, которые оказались закороченными, будет иметь максимально возможную пропускную способность \hat{Q} . Ограничение на применимость данного алгоритма только к неориентированным графам вызвано именно процессом «закорачивания», так как при этом неявно предполагается, что путь с пропускной способностью, не меньшей, чем \bar{Q} , существует между любыми вершинами (двумя или большим числом), заменяемыми в описанном процессе единственной вершиной, но такое предположение может не выполняться, если закорачиваемые ребра имеют ориентацию.

Пример. Задача состоит в нахождении $(s-t)$ -пути (s) иибольпей пропускной способности в неориентированном графе, изображенном на рис. ба, где пропускные способности ребер указаны числами, стоящими у этих ребер.

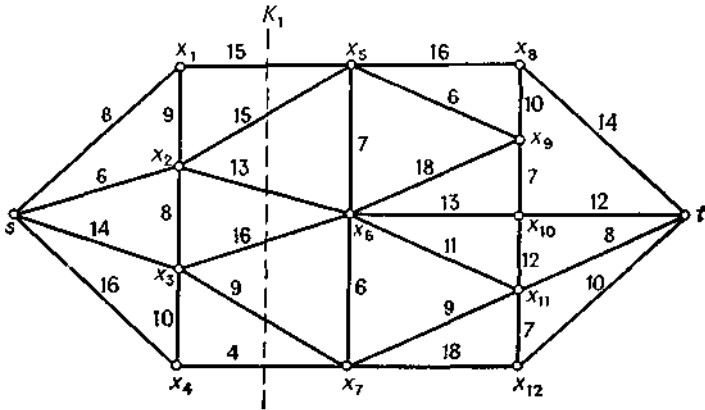


Рис. 6а. Граф из примера.

Выбрав произвольно $(s-t)$ -разрез K_1 , показанный на рис. 6а пунктиром, найдем, что ребром с максимальной пропускной способностью в K будет (x_3, x_6) с $q_{3,6} = 16$. Закорачивая все ребра с пропускной способностью ≥ 16 , получим граф G_1 , изображенный на рис. 6б.

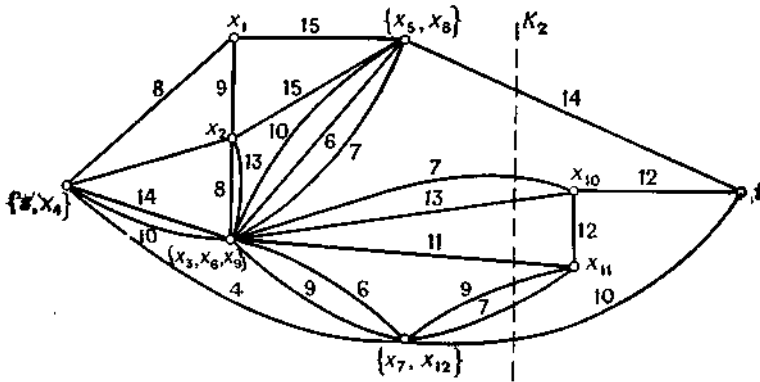


Рис. 6б. Граф после первого стягивания.

Выберем теперь $(s-t)$ -разрез K_2 графа G_1 так, как показано пунктиром на рис. 6б. Наибольшая пропускная способность ребер в K_2 равна 14. Поэтому закорачиваем все ребра с пропускной способностью ≥ 14 и получаем граф G_2 , изображенный на рис. 6в

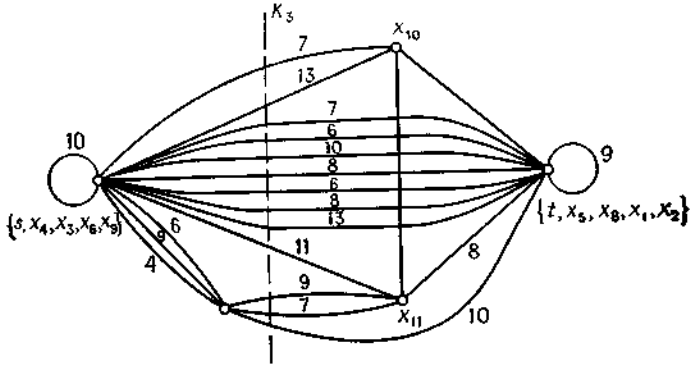


Рис. 6в. Граф после второго стягивания.

Беря в качестве следующего $(s-t)$ -разреза K_3 — он показан пунктиром на рис. 6в,— получаем $Q = 13$, а в построенном графе (полученном из G_2 после закорачивания всех ребер с пропускной способностью ≥ 13) вершины s и t будут закорочены.

Окончательным графом G будет остовный подграф графа G , содержащий лишь те ребра, которые закорачивались во время описанной процедуры (т. е. только ребра (x_i, x_j) с $q_{ij} \geq 13$). Этот последний граф изображен на рис. 6г и видно, что существует только один $(s-t)$ -путь с наибольшей пропускной способностью.

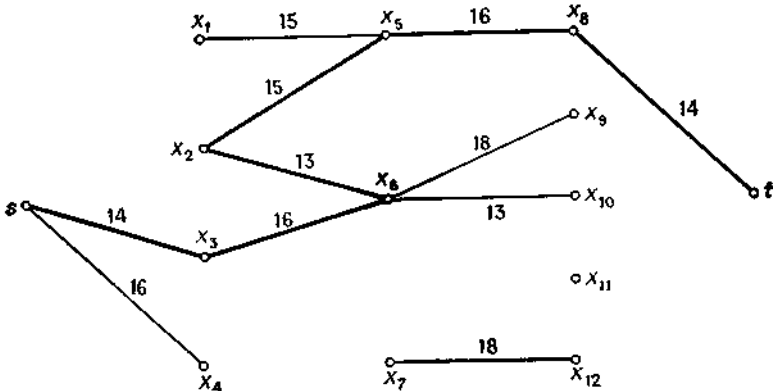


Рис. 6г. Граф G и путь с наибольшей пропускной способностью.

Этот путь показан жирными линиями, причем критическим ребром этого пути будет (x_2, x_6) с пропускной способностью 13.

Если требуется найти пути с наибольшей пропускной способностью между каждой парой вершин графа, то трехместную операцию из выражения (8) можно использовать в форме

$$q_{ij} = \max [q_{ij}, \min \{q_{ik}, q_{kj}\}], \quad (15)$$

поскольку так введенная операция удовлетворяет требованиям, налагаемым соотношением (9). Начальной матрицей $[q_{ij}]$ является исходная матрица пропускных способностей дуг (с нулевыми элементами на местах отсутствующих дуг); конечная матрица $[q_{ij}]$ дает пропускные способности путей между всеми парами вершин.

5.6.3. Путь с наибольшей приведенной пропускной способностью

Рассмотрим граф G , в котором каждой дуге (x_i, x_j) приписаны два числа ρ_{ij} и q_{ij} , представляющие соответственно надежность и пропускную способность дуги. Задача нахождения пути от s к t наибольшей приведенной пропускной способностью является комбинацией двух последних задач о путях, обсуждавшихся выше в разд. 5.6.1 и 5.6.2. Если приведенную пропускную способность пути P обозначить через $e(P)$, то задача состоит в нахождении пути, минимизирующего выражение

$$e(P) = \prod_{(x_i, x_j) \in P} \rho_{ij} \cdot \left\{ \min_{(x_i, x_j) \in P} [q_{ij}] \right\}. \quad (16)$$

В этом случае нельзя воспользоваться трехместной операцией, так как не удастся найти оператора \otimes , который (для пути от x_a к x_b через вершину x_c) удовлетворяет условию $e_{ab} = e_{ac} \otimes e_{cb}$.

Несуществование такого оператора можно обосновать следующим образом:

$$\begin{aligned} e_{ab} &= \prod_{(x_i, x_j) \in a \rightarrow b} \rho_{ij} \cdot \min_{(x_i, x_j) \in a \rightarrow b} [q_{ij}] = \\ &= \prod_{(x_i, x_j) \in a \rightarrow c} \rho_{ij} \cdot \prod_{(x_i, x_j) \in c \rightarrow b} \rho_{ij} \cdot \min \left\{ \min_{(x_i, x_j) \in a \rightarrow c} [q_{ij}], \min_{(x_i, x_j) \in c \rightarrow b} [q_{ij}] \right\}, \end{aligned}$$

где запись $a \rightarrow b$ и т. д. применяется как для пути от x_a к x_b , так и для множества дуг этого пути. Вводя обозначения

$$\begin{aligned} \bar{\rho}_{ac} &= \prod_{(x_i, x_j) \in a \rightarrow c} \rho_{ij}, \quad \bar{\rho}_{cb} = \prod_{(x_i, x_j) \in c \rightarrow b} \rho_{ij}, \\ \bar{q}_{ac} &= \min_{(x_i, x_j) \in a \rightarrow c} [q_{ij}], \quad \bar{q}_{cb} = \min_{(x_i, x_j) \in c \rightarrow b} [q_{ij}], \end{aligned}$$

можем записать

$$e_{ac} = \bar{\rho}_{ac} \cdot \bar{q}_{ac}, \quad e_{cb} = \bar{\rho}_{cb} \cdot \bar{q}_{cb}.$$

Следовательно,

$$e_{ab} = \min [\bar{\rho}_{ac} e_{cb}, \bar{\rho}_{cb} e_{ac}]. \quad (17)$$

Из этого выражения видно, что так как надежности $\bar{\rho}_{ac}$ и $\bar{\rho}_{cb}$ не могут быть выражены «на языке» величин e_{ac} , e_{cb} , ρ_{ij} и q_{ij} , то невозможно найти оператор, удовлетворяющий общим условиям (9) и, следовательно, невозможно непосредственно использовать трехместную операцию.

На самом деле нетрудно показать, что оптимальный путь от x_a к x_b через x_c не зависит от:

- (i) оптимальных путей от x_a к x_c и от x_c к x_b ;
- (ii) путей наибольшей пропускной способности от x_a к x_c и от x_c к x_b ;
- (iii) путей наибольшей надежности от x_a к x_c и от x_c к x_b или любой их комбинации.

В этом можно убедиться, рассматривая рис. 7, на котором изображены пути всех трех вышеуказанных типов от x_a к x_c и от x_c к x_b , причем оптимальное значение величины e , которое можно получить с помощью некоторой комбинации этих путей, равно 4,20.

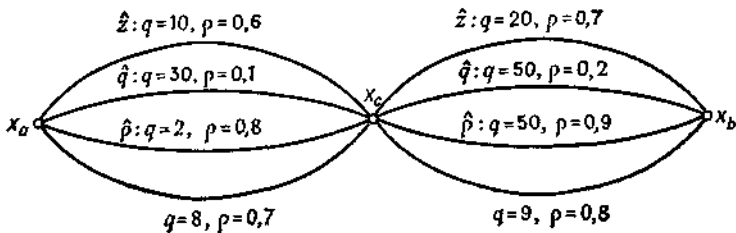


Рис. 7.

На рисунке изображены, кроме того, такие два пути от x_a к x_c и от x_c к x_b , ни один из которых не является оптимальным в любом из трех вышеприведенных смыслах, между их соответствующими конечными вершинами. Однако эти два пути образуют оптимальный путь между вершинами x_a и x_b , которому отвечает значение величины e , равное 4,48.

5.6.4. Метод нахождения пути наибольшей приведенной пропускной способности

Здесь мы дадим итерационный метод нахождения пути с наибольшей приведенной пропускной способностью от некоторой указанной вершины s в другую вершину t графа $G = (X, A)$. Этот метод состоит в постепенном исключении тех дуг графа, которые не могут принадлежать оптимальному пути. Исключение осуществляется до тех пор, пока граф не станет несвязным.

Сначала находим путь $P_{\hat{\rho}}$ между s и t , имеющий наибольшую надежность. Пусть $Q_{\hat{\rho}}$ —пропускная способность этого пути, т. е.

$$Q_{\hat{\rho}} = \min_{(x_i, x_j) \in P_{\hat{\rho}}} [q_{ij}].$$

При этом через $P_{\hat{\rho}}$ будет обозначаться также и множество дуг, образующих рассматриваемый путь. Если из A удалить множество дуг $A_0 \equiv \{(x_i, x_j) \mid (x_i, x_j) \in A, q_{ij} \leq Q_{\hat{\rho}}\}$ и образовать множество $A' = A - A_0$, то либо граф $G' = (X, A')$, являющийся остовным подграфом графа G , содержит оптимальный путь из G , либо $P_{\hat{\rho}}$ будет оптимальным путем. Это можно обосновать так.

Оптимальный путь в G' должен иметь по определению надежность, не большую, чем надежность пути $P_{\hat{\rho}}$. Следовательно, значение приведенной пропускной способности должно быть не меньше, чем значение этой величины для $P_{\hat{\rho}}$. Если $P_{\hat{\rho}}$ не оптимальный путь, то пропускная способность оптимального пути в G больше, чем $Q_{\hat{\rho}}$, а значит, ему не принадлежит никакая дуга из A_0 .

Далее уже в графе G' ищется путь P'_ρ с наибольшей надежностью. Пропускная способность Q'_ρ этого пути больше, чем $Q_{\hat{\rho}}$, но его надежность не превосходит надежности пути $P_{\hat{\rho}}$. Если значение приведенной пропускной способности пути P'_ρ больше, чем значение этой величины для $P_{\hat{\rho}}$, то P'_ρ берется как лучший путь и хранится до тех пор, пока не будет найден путь с большей приведенной пропускной способностью. Затем из A удаляется соответствующее множество дуг

$$A_1 = \{(x_i, x_j) \mid (x_i, x_j) \in A', q_{ij} \leq Q'_\rho\}.$$

Получается множество $A'' = A' - A_I$ и остовный подграф $G'' = (X, A'')$. По тем же самым причинам, что и выше, либо граф G'' содержит оптимальный путь, либо оптимальным будет наилучший из найденных до данного момента путь. Этот процесс продолжается до тех пор, пока не получится остовный подграф G^l , удовлетворяющий одному из следующих условий: либо он является несвязным, причем в нем нет пути между s и t , либо приведенная пропускная способность наилучшего из найденных ранее путей будет больше, чем произведение надежности пути P^l на пропускную способность такого пути в графе G , который обладает наибольшей пропускной способностью (этот путь можно найти с помощью методов, описанных выше); тогда, очевидно, никакой остовный подграф графа G^l не приводит к лучшему решению.

Оптимальным путем будет текущий наилучший путь, полученный в конце описанной процедуры.

На эффективность вышеприведенного метода влияют два фактора. Первый — «скорость» удаления дуг из графа. В худшем случае, когда на каждом этапе наиболее надежный путь использует дугу с наименьшей пропускной способностью, понадобится $m - k$ этапов (для нахождения пути с наилучшей надежностью), где m — число дуг в G и k — число дуг в оптимальном пути. Самым хорошим случаем будет тот, когда уже первый остовный подграф G' является несвязным и вершины s и t находятся в разных компонентах. Тогда потребуется только один этап.

Второй фактор связан с тем методом, который используется для «перевычисления» (на каждом этапе) пути с наибольшей надежностью в остовном подграфе. При этом следует заметить, что хотя удаление дуг может привести, вообще говоря, к элиминации нескольких дуг из s -базы предшествующего графа (дающего наиболее надежные пути от s ко всем другим вершинам), но та часть s -базы, которая содержит s вместе с пометками вершин, остается неизменной и ее не нужно находить заново.

Пример. Найти путь от s к t с наибольшей приведенной пропускной способностью в графе G , изображенном на рис. 8, где каждая дуга имеет пометку (a, b) , причем a является пропускной способностью дуги, а b — ее надежностью.

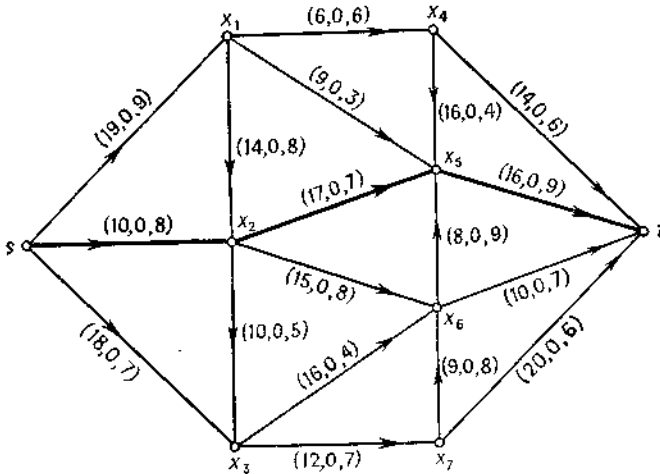


Рис. 8. Граф из примера.

Путь P_p с наибольшей надежностью в графе G показан на рис. 8 жирными линиями. Здесь $\beta = 0,504$, $Q_p = 10$ и, следовательно, приведенная пропускная способность этого пути равна $e = 5,04$. Удаляя из G все дуги с пропускными способностями ≤ 10 , получим граф G' , изображенный на рис. 9а.

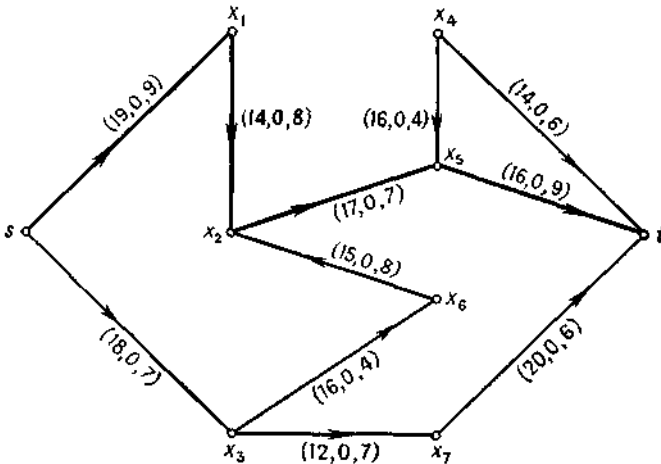


Рис. 9а. Граф G' из примера.

Путь P'_b с наибольшей надежностью в графе G' показан на рисунке жирными линиями. Для этого пути $\beta = 0,454$, $Q_b = 14$ и, значит, приведенная пропускная способность пути P'_b равна $e=14 \times 0,454=6,36$, что больше полученной ранее величины (5,04) и, следовательно, заменяет ее. Удаляя из G все дуги с пропускной способностью ≤ 14 , получаем граф G'' , изображенный на рис. 9б.

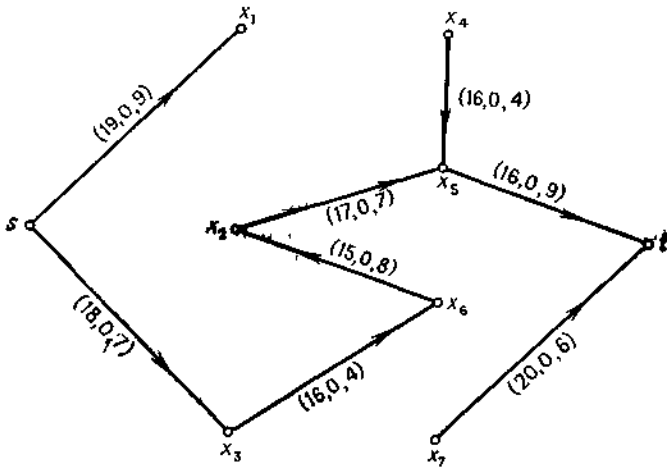


Рис. 9б. Граф G'' из примера.

В этом графе существует только один путь между s и t ; $\beta = 0,141$, $Q''_b = 15$ и, следовательно, приведенная пропускная способность равна 2,12, что хуже, чем лучшее предыдущее значение этой величины. Удаляя все дуги с пропускной способностью ≤ 15 из G'' , мы разъединим вершины s и t , так что лучший предыдущий ответ, т. е. путь $(s, 1, 2, 5, t)$, с приведенной пропускной способностью 6,36 будет оптимальным ответом.

Пусть T — ориентированный остов графа $G = (X, A)$ и корнем остова T является вершина s . Пусть $l(x_i)$ — расстояние от вершины s до вершины x_i , измеренное вдоль пути в этом дереве.

Теорема. T является s -базой тогда и только тогда, когда для каждой дуги $(x_i, x_j) \in A$ выполняется условие $l(x_j) \leq l(x_i) + c_{ij}$.

Вышеприведенная теорема является основой следующей итерационной процедуры нахождения кратчайшего пути от s ко всем другим

вершинам графа (здесь мы предполагаем, что все вершины $x_i \in X$ достижимы из s).

Шаг 1. Взять в качестве T любой ориентированный остов графа G и для каждой вершины x_i найти пометку $l(x_i)$, равную расстоянию между этой вершиной и корнем s вдоль пути в дереве T .

Шаг 2. Если каждая дуга $(x_i, x_j) \in A$ удовлетворяет условию

$$l(x_j) \leq l(x_i) + c_{ij},$$

то остановиться. Текущий остов T будет s -базой графа G . В противном случае перейти к шагу 3.

Шаг 3. Если для некоторой дуги (x'_i, x'_j)

$$l(x'_j) > l(x'_i) + c_{i'j'},$$

то добавить к дереву дугу (x'_i, x'_j) и удалить из него ту дугу (x_i, x'_j) , конечной вершиной которой является x'_i , после чего возникает новое дерево T .

Шаг 4. Обновить пометки $l(x_k)$ у всех вершин x_k , достижимых из x'_j вдоль путей дерева T . Возвратиться к шагу 2.

Вышеописанный алгоритм является конечным, и его сложность для полного n -вершинного графа растет как $O(n^3)$.

Далее займемся изучением циклов в графах и исследованием некоторых из их свойств и связей с другими понятиями (такими, как понятие дерева), введенными ранее. Два типа циклов в графах, эйлеровы и гамильтоновы циклы, часто встречаются в практических задачах и поэтому представляют особый интерес.

В настоящей главе мы будем заниматься циклами как в ориентированных, так и в неориентированных графах, а также в мультиграфах, являющихся слабым обобщением понятия графа. Последние являются графами, в которых может существовать несколько дуг

(x_i, x_j) между двумя данными вершинами x_i и x_j . Если наибольшее число «запараллеленных» дуг равно s , то граф называется s -графом. Так на рисунке 10 представлен 3-граф. Очевидно, что обычный граф можно рассматривать как 1-граф.

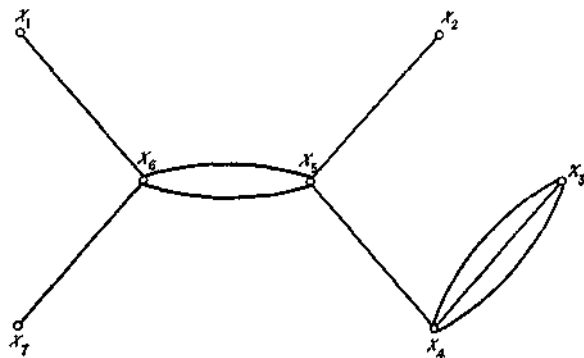


Рис. 10

s-Графы очень часто возникают на практике, когда граф используется для представления физической системы в естественных науках, вопросах управления, инженерном деле и т. д. Так, например, 3-граф на рис. 10 изображает молекулярную структуру органического химического соединения акрилонитрила. В электротехнике графы, изображающие электрические цепи, почти всегда являются s-графами, так как параллельно может быть включено несколько электрических компонентов. В проблемах надежности оборудования или сетей связи наиболее важные устройства или линии связи часто дублируются, утраиваются и т. д., а возникающая избыточность увеличивает надежность системы. Графы таких систем также являются s-графами.

5.7. Цикломатическое число и фундаментальные циклы

Пусть G — неориентированный s-граф с n вершинами, m ребрами и p связными компонентами.

Определим число $\rho(G)$ как

$$\rho(G) = n - p, \quad (18)$$

$\rho(G)$ дает тогда полное число ребер в остовах каждой из p связных компонент графа G .

Число $\nu(G)$ определяется как

$$\nu(G) = m - \rho(G) = m - n + p \quad (19)$$

и называется *цикломатическим числом* (иногда его называют также дефектом или *первым числом Бетти*), а число $\rho(G)$ называется *коцикломатическим числом*.

В теории электрических цепей (и на самом деле при представлении любой системы с сосредоточенными параметрами) числа $\rho(G)$ и $\nu(G)$ имеют прямой физический смысл. Так, цикломатическое число равно наибольшему числу независимых контуров в графе электрической цепи, т. е. наибольшему числу независимых круговых токов, которые могут протекать в цепи. Коцикломатическое число равно наибольшему числу независимых разностей потенциалов между узлами электрической цепи.

Рассмотрим, например, электрическую цепь, показанную на рис. 11а. Неориентированный граф G этой цепи состоит из единственной связной компоненты ($p = 1$) и изображен на рис. 11б, где остов T показан жирной линией.

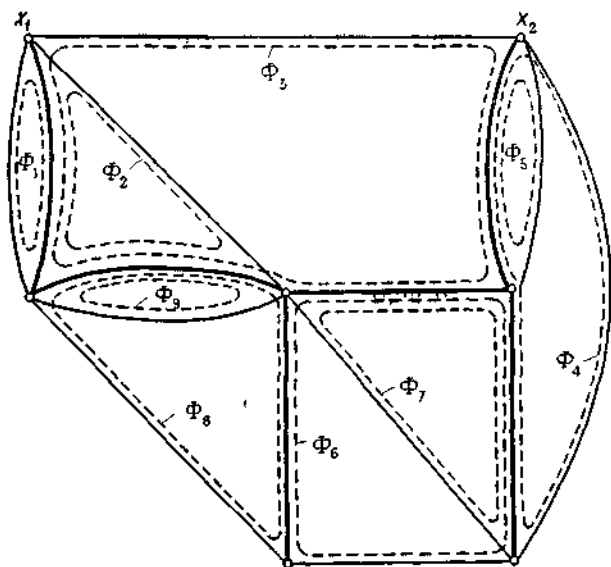
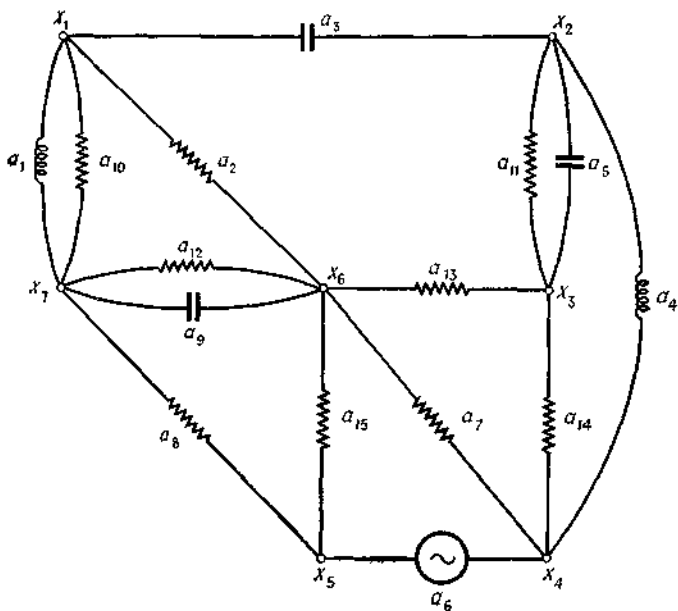


Рис. 11. (а) Электрическая цепь. (б) Фундаментальные циклы.

Добавление любого ребра (x_i, x_j) из G , не принадлежащего T , к ребрам дерева T приводит к образованию точно одного (простого) цикла, состоящего из ребер остова T , лежащих на (единственной) цепи из x_j в x_i и только что добавленного ребра. Например, добавляя ребро $a_3=(x_1, x_2)$, получаем цикл $(x_1, x_7, x_6, x_3, x_2, x_1)$. Так как в графе G имеется m ребер, $n - 1$ из которых лежат в T , то число всех циклов, построенных таким способом, равно $m - n + 1$, что совпадает с цикломатическим числом графа G . В примере на рис. 11 число таких циклов равно $15 - 7 + 1=9$, они изображены на рисунке пунктирными линиями и обозначены через $\Phi_1, \Phi_2, \dots, \Phi_9$. Все эти циклы, очевидно, независимы между собой, так как каждый из них имеет по крайней мере одно ребро, не принадлежащее никакому другому циклу. В общем случае $\nu(G)$ циклов, получаемых добавлением какого-либо ребра из G , не лежащего в T , к ребрам T , называются *фундаментальными циклами*. Если семейство этих циклов обозначить через Φ (в нашем примере $\Phi = \{\Phi_1, \Phi_2, \dots, \Phi_9\}$), то любой другой цикл в графе, не принадлежащий к Φ , может быть выражен в виде линейной комбинации циклов из Φ , если принять следующее соглашение.

Пусть каждый фундаментальный цикл $\Phi_i, i = 1, \dots, \nu(G)$, представлен m -мерным вектором, в котором j -я компонента равна 1 или 0 в зависимости от того, принадлежит ли j -е ребро данному циклу. Тогда, используя Φ как символ сложения по модулю 2, любой цикл Φ_k можно представить как сумму по модулю 2 фундаментальных циклов. Так, цикл $\Phi_{10} = (a_3, a_{11}, a_{14}, a_6, a_8, a_1)$ на рис. 11(б) может быть представлен в виде

$$\begin{aligned} \Phi_{10} &= \Phi_1 \oplus \Phi_3 \oplus \Phi_6 \oplus \Phi_8 \\ &= (100000000100000) \oplus (001000000111100) \\ &\quad \oplus (000001000000111) \oplus (000000010001001) \\ &= (101001010010010) \end{aligned}$$

Отметим, что обращение вышеприведенного утверждения неверно, а именно некоторая сумма по модулю 2 фундаментальных циклов не обязательно дает единственный цикл, но может представлять два и более циклов. Например, сумма $\Phi_2 \oplus \Phi_3 \oplus \Phi_6 \oplus \Phi_7$ соответствует двум циклам $(a_2, a_3, a_{11}, a_{13})$ и (a_6, a_7, a_{15}) . Таким образом, для порождения всех циклов графа G не надо брать все $2^{\nu(G)} - 1$ комбинаций фундаментальных циклов и складывать их по модулю 2: некоторые из этих сумм на самом деле не будут циклами. Более того,

если данная сумма, скажем $\Phi_i \oplus \Phi_j \oplus \Phi_k \oplus \dots$, не порождает цикла, то нельзя отбросить другие суммы, ее содержащие, так как, складывая по модулю 2 сумму $\Phi_i \oplus \Phi_j \oplus \Phi_k \oplus \dots$ с другой суммой, например $\Phi_\alpha \oplus \Phi_\beta \oplus \Phi_\gamma \oplus \dots$ (которая сама может не порождать цикла), мы можем получить цикл. Для преодоления этой трудности Уэлч предложил некоторый метод, позволяющий отбрасывать «некорректные» комбинации фундаментальных циклов, как только они появляются. Этот метод основан на упорядочении фундаментальных циклов в соответствии со специальным множеством правил.

Следует также заметить, что, хотя число фундаментальных циклов равно $\nu(G)$, сами эти циклы определены неоднозначно и зависят от первоначально выбранного остова T . Действительно, в графе G можно найти множество из $\nu(G)$ независимых циклов, которые нельзя получить добавлением ребер к дереву, как это делалось выше. О таком множестве не следует говорить, что оно «фундаментальное».

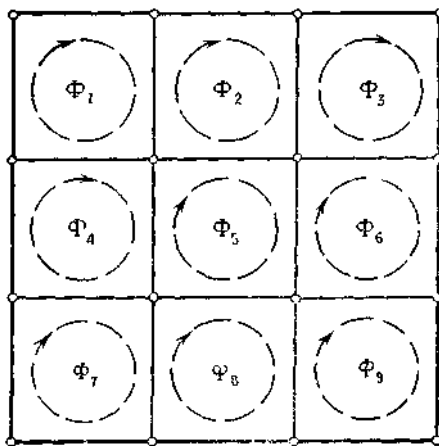


Рис. 12. Независимые, но нефундаментальные циклы.

На рис. 12 показано множество из $\nu(G) = 9$ независимых циклов графа G , которое не может быть получено добавлением ребер ни к какому остову графа G и которое поэтому не будет фундаментальным множеством.

5.8. Деревья графа

Будем называть *деревом связного графа* любое покрывающее дерево (каркас или остов), связывающее все его вершины и имеющее в качестве ветвей ребра этого графа. Два дерева считаются различными, если они отличаются хотя бы одним ребром.

Существует простой способ определения количества различных деревьев графа без петель или (или мультиграфа) с p вершинами. Для этого необходимо записать квадратную матрицу p -го порядка, по главной диагонали которой расположены степени вершин, а ij - и ji -элементы равны взятому со знаком минус числу ребер, связывающих вершины i и j .

Вычислив любой из главных миноров этой матрицы, получим искомое число деревьев графа. Например, для графа рис. 13 имеем:

$$D = \begin{bmatrix} 3 & -1 & -2 & 0 & 0 \\ -1 & 4 & -1 & -1 & -1 \\ -2 & -1 & 4 & -1 & 0 \\ 0 & -1 & -1 & 5 & -3 \\ 0 & -1 & 0 & -3 & 4 \end{bmatrix}; \quad \Delta_{22} = \begin{bmatrix} 3 & -2 & 0 & 0 \\ -2 & 4 & -1 & 0 \\ 0 & -1 & 5 & 3 \\ 0 & 0 & -3 & 4 \end{bmatrix} = 76.$$

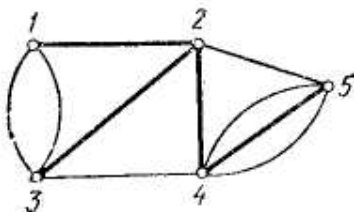


Рис.13. Связный граф и одно из его деревьев (выделено жирными линиями).

Одно из 76 деревьев графа изображено на рис. 13 жирными линиями. Приведенный способ определения числа деревьев графа известен как *теорема Трента*.

5.8.1 Формирована дерева графа

Для данного (p, q) графа процесс формирования дерева можно организовать поочередным рассмотрением ребер двумя способами:

- 1) очередное ребро графа относится к дереву, если оно не образует цикла с уже выбранной совокупностью ветвей, до тех пор пока не получится $p - 1$ ветвей, составляющих дерево;
- 2) очередное ребро удаляется из графа, если оно образует контур с оставшимися ребрами, до тех пор пока не будет удалено $q - p + 1$ хорд, составляющих дополнение (остальные $p - 1$ ребер служат ветвями дерева).

Если требуется сформировать дерево с преимущественным включением ребер (например, экстремальное дерево), то ребра рассматриваются в порядке их иерархии по весу или какому-либо признаку. При использовании вычислительных средств граф $G = (V, E)$ задается как совокупность ребер $e_i = (\alpha_i, \beta_i)$, где α_i и β_i — вершины, инцидентные ребру e_i , причем $\alpha_i, \beta_i \in V$ и $e_i \in E$. Ребра упорядочиваются в соответствии с принятой иерархией и дерево формируется, например, по первому способу.

В промежуточной ситуации совокупность ребер, отнесенных к дереву, образует некоторый (вообще, несвязный) подграф (рис. 14).

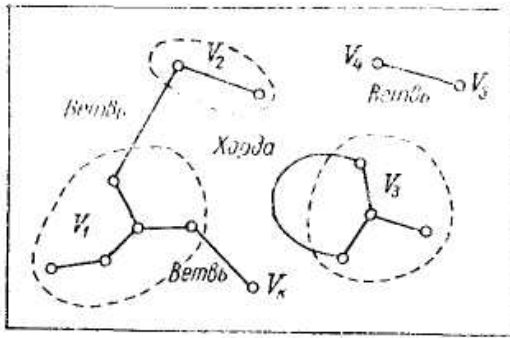


Рис. 14. Классы эквивалентности на множестве вершин графа.

Множества вершин каждой из компонент этого подграфа и одноэлементные множества, содержащие не вошедшие в этот подграф вершины, образуют совокупность классов эквивалентности V_i и определяют на множестве вершин V соответствующее разбиение. Ясно, что ребро e_i должно быть отнесено к дереву, если и только если инцидентные ему вершины α_i и β_i принадлежат различным классам эквивалентности. Пусть $V_r \neq V_s$ и $e_i \in N$ тогда $e_i \in T$ при условии $\alpha_i \in V_r$ и $\beta_i \in V_s$, при условии $V_r = V_s$ (через T обозначено дерево и через N — его дополнение). Включение ребра e_i в дерево означает объединение тех частей подграфа ветвей, к которым

принадлежат инцидентные вершины этого ребра, т. е. классы эквивалентности V_r и V_s объединяются в класс $V_r \cup V_s$. Включение ребра e_i в дополнение не изменяет разбиения множества вершин.

В исходном положении все p классов эквивалентности содержат по одной вершине, т. е. имеем полное разбиение множества вершин. Первое же рассматриваемое ребро относится к дереву и образует двухэлементный класс, объединяющий пару инцидентных этому ребру вершин.

В дальнейшем каждое включение ребра в дерево сопровождается объединением двух классов эквивалентности, так что при выборе для дерева k ветвей разбиение состоит из $p-k$ классов.

Процесс формирования дерева заканчивается после того, как будет отобрано $p-1$ ветвей, совокупность которых образует связный подграф без контуров. Ему соответствует полное отношение эквивалентности на множестве вершин графа, при котором все вершины объединяются в единственный класс. Если исходный граф несвязный, то конечное разбиение содержит столько же классов эквивалентности, сколько компонент имеет граф, причем каждый из этих классов объединяет вершины соответствующих компонент графа.

При машинной реализации процедуры формирования дерева (или леса) последовательные разбиения образуются путем идентификации вершин на одномерном массиве, содержащем p ячеек памяти, расположенных в той последовательности, которая принята при нумерации вершин. Так как любая вершина из данного класса может служить его представителем, то всем таким вершинам присваивается номер одной из них, который заносится в соответствующие ячейки. В исходном положении содержимое ячеек совпадает с их номерами, и в конечном — все ячейки каждой компоненты графа содержат одинаковые номера.

При рассмотрении очередного ребра $e_i = (\alpha_i, \beta_i)$ на массиве вершин учитывается содержимое r_i и s_i ячеек по адресам α_i и β_i . При $r_i = s_i$ ребро e_i относится к дополнению ($e_i \in N$), а при $r_i \neq s_i$ ребро e_i относится к дереву ($e_i \in T$) и одновременно осуществляется идентификация $r_i = s_i$ на множестве вершин V (во всех ячейках число r_i замещается числом s_i).

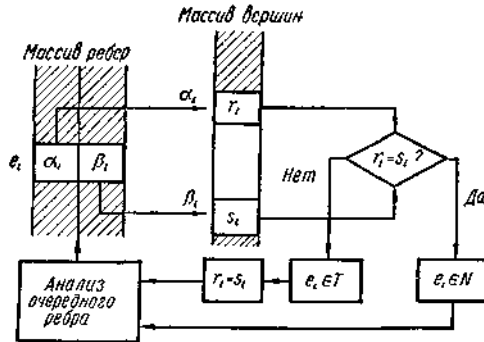


Рис. 15. Процедура формирования дерева графа.

Изложенная процедура изображена на рис.15 и иллюстрируется следующим примером:

	i	1	2	3	4	5	6	7	8	9
Массив ребер	α_i	1	2	2	1	3	1	1	2	4
	β_i	3	5	5	4	4	4	2	4	5

Идентификация на массиве вершин

Исходные разбитие	1	3	3		4			5		
	2	2	5		5			5		
	3	3	3		4			5		
	4	4	4		4			5		
	5	5	5		5			5		

Исходный граф и сформированное дерево показаны на рис. 16.

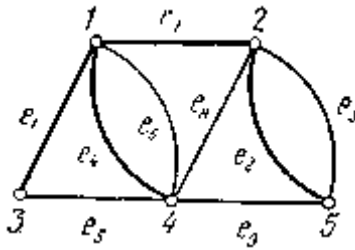


Рис. 3.13. Граф к примеру формирования дерева.

Благодаря невысоким требованиям к оперативной памяти изложенный алгоритм позволяет практически мгновенно формировать на вычислительной машине (компьютере) дерево графа, содержащее тысячи ребер.

5.8.2. Выявление всех деревьев графа

В ряде случаев, например при анализе цепей и систем, может возникнуть потребность получить все покрывающие деревья графа. Для решения этой задачи разработано много различных алгоритмов. Поясним один из них на примере графа рис. 17, ребра которого пронумерованы порядковыми числами.

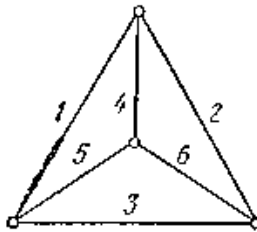


Рис. 17. Граф с порядковой нумерацией ребер.

Сначала записываются $p-1$ множество номеров ребер, инцидентных $p-1$ вершинам графа (кроме одной из p вершин):

$$\Omega_1 = \{1, 3, 5\}; \quad \Omega_2 = \{1, 2, 4\}; \quad \Omega_3 = \{2, 3, 6\}.$$

Рассматривая последовательно эти множества, образуем таблицы, столбцы которых представляют собой всевозможные сочетания различных ребер:

$$\begin{aligned} \begin{bmatrix} \Omega_1 \\ \Omega_2 \end{bmatrix} &= \begin{bmatrix} 1 & 1 & 3 & 3 & 3 & 5 & 5 & 5 \\ 2 & 4 & 1 & 2 & 4 & 1 & 2 & 4 \end{bmatrix}; \\ \begin{bmatrix} \Omega_1 \\ \Omega_2 \\ \Omega_3 \end{bmatrix} &= \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 3 & 3 & 3 & 3 & 3 & 5 & 5 & 5 & 5 & 5 & 5 \\ 2 & 2 & 4 & 4 & 4 & 1 & 1 & 2 & 4 & 4 & 1 & 1 & 1 & 2 & 2 & 4 & 4 & 4 \\ 3 & 6 & 2 & 3 & 6 & 2 & 6 & 6 & 2 & 6 & 2 & 3 & 6 & 3 & 6 & 2 & 3 & 6 \end{bmatrix}. \end{aligned}$$

Столбцы, содержащие одинаковые ребра (независимо от их порядка), попарно вычеркиваются, в результате чего получаем таблицу Ω , называемую *структурным числом графа*:

$$\Omega = \begin{bmatrix} 1 & 1 & 1 & 1 & 3 & 3 & 3 & 3 & 5 & 5 & 5 & 5 & 5 & 5 & 5 \\ 2 & 4 & 4 & 4 & 1 & 2 & 4 & 4 & 1 & 1 & 1 & 2 & 2 & 4 & 4 & 4 \\ 6 & 2 & 3 & 6 & 6 & 6 & 2 & 6 & 2 & 3 & 6 & 3 & 6 & 2 & 3 & 6 \end{bmatrix}.$$

Количество столбцов в этой таблице равно числу всех деревьев и каждый столбец соответствует одному из них. При этом числа в столбце указывают номера тех ребер, из которых состоит данное дерево. Ясно, что порядок следования столбцов и чисел в каждом столбце не меняет существа дела. Поэтому структурные числа считаются эквивалентными, если они содержат одинаковое количество столбцов, каждый из которых представлен различными множествами их элементов.

Несмотря на простоту изложенного алгоритма, его применение затруднено из-за необходимости выявлять и удалять столбцы, содержащие одинаковые множества ребер. Легко понять, что каждый такой столбец соответствует несвязному суграфу с одним или несколькими циклами, причем они обязательно появляются четное число раз. В рассматриваемом примере — это столбцы (1, 2, 3) и (3, 1, 2).

5.8.3. *k*-деревья

Так как дерево (p, q) -графа представляет собой минимальный связный подграф с p вершинами и $p-1$ ребрами, то удаление из дерева любой ветви разбивает его на две несвязные компоненты (компонентой может быть также изолированная вершина). Такой подграф с p вершинами и $p-2$ ребрами, не содержащий циклов, называют *2-деревом*. Вообще *k-дерево* можно определить как суграф без циклов, содержащий $p-k$ ребер графа. При этом понятие *1-дерева* совпадает с покрывающим деревом или просто деревом графа.

K-дерево можно получить путем исключения $k-1$ ветвей (при сохранении всех вершин) из покрывающего дерева, которое тем

самим разбивается на k компонент. Соответственно множество V вершин графа разбивается на k непересекающихся подмножеств V_1, V_2, \dots, V_k , каждое из которых содержит вершины соответствующей компоненты k -дерева. Примеры k -деревьев приведшим на рис. 18.

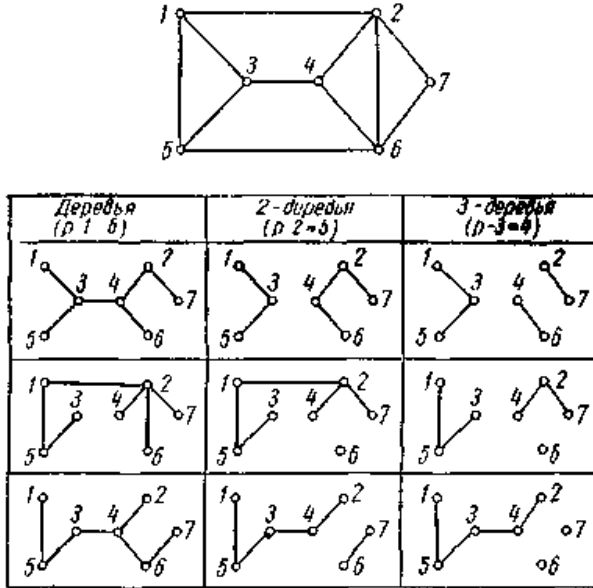


Рис.18. Примеры k -деревьев.

Если v_i и v_j — вершины различных компонент 2-дерева, т.е. $v_i \in V_1$ и $v_j \in V_2$, причем $V_1 \cap V_2 = \emptyset$, то оно становится деревом графа, полученного из исходного объединением вершин v_i и v_j . При этом говорят, что 2-дерево относится к типу (v_i, v_j) . Очевидно, число всех 2-деревьев (v_i, v_j) -типа равно числу деревьев графа, образованного из исходного объединением вершин v_i и v_j (ребра, соединяющие эти вершины, удаляются).

5.8.4. Деревья с корнем. Ветви

Пусть в дереве G отмечена вершина v_0 . В таком дереве можно естественным образом ориентировать ребра. Вершину v' ребра (v', v'') можно соединить единственной цепью L с корнем v_0 . Если эта цепь не содержит ребра (v', v'') , в последнее вводится ориентация от v' к v'' ; в противном случае — от v'' к v' . Такая ориентация согласована с

ориентацией того же ребра, определенной через вершину v'' . Действительно, если цепь L не содержит ребра (v', v'') , то, добавив это ребро, мы получим единственную цепь L' , связывающую v_0 и v'' и проходящую через рассматриваемое ребро. Если же цепь L содержит ребро (v', v'') , то, отбросив его, мы получим цепь L' , связывающую v_0 и v'' и не содержащую этого ребра.

Ориентированное таким образом дерево с корнем называется *ориентированным деревом*. В нем все ребра имеют направление *от корня* (рис. 19).

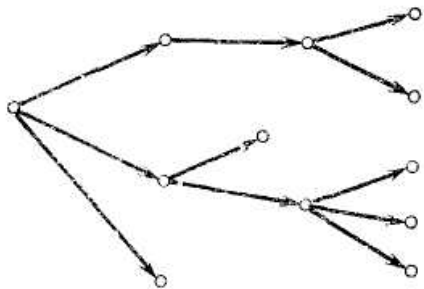


Рис.19.

Если изменить направления всех ребер ориентированного дерева на противоположные (*к корню*), получится ориентированный граф, который тоже называют ориентированным деревом, а иногда *сетью сборки*.

В каждую вершину ориентированного дерева (за исключением корня) входит только одно ребро, т. е. эта вершина является концом одного и только одного ребра. Действительно, такое ребро является последним в цепи, связывающей рассматриваемую вершину с корнем, а эта цепь единственна. В корень не входит ни одно ребро, все инцидентные корню ребра связывают его со своими вторыми концами, значит, корень является их началом. Любое дерево можно ориентировать, выбрав в качестве корня любую его вершину. Это доказывает, что в конечном дереве $ч$ и $сл$ о вершин на единицу больше числа $ребер$.

Пусть v — вершина дерева G с корнем v_0 , $B(v)$ — множество вершин, связанных с корнем цепями, проходящими через вершину v (т. е. имеющими ребра, инцидентные v).

Это множество порождает подграф $G(v)$, который называется *ветвью* вершины v в дереве с корнем v_0 . Как подграф дерева, т. е. графа без циклов, $G(v)$ тоже не имеет циклов; любые вершины $v, v'' \in B(v)$ связаны в этой ветви маршрутом, составленным из участков цепей $L(v')$ и $L(v'')$ от корня v_0 до этих вершин, начинающихся в вершине v .

Нужно только доказать, что эти участки целиком погружены в $G(v)$, но это справедливо, так как в каждую вершину такого участка ведет цепь из корня v_0 , являющаяся начальным участком соответствующей цепи $L(v')$ или $L(v'')$ и проходящая через вершину v . Таким образом, ветвь $G(v)$ связна и сама является деревом.

Рассмотрим $G(v)$ как дерево с корнем v . Для любой вершины $v' \in G(v)$ связывающая ее v и цепь $L(v, v')$ — это конец цепи $L(v_0, v')$, связывающей в дереве G эту вершину с корнем v_0 . Следовательно, последнее ребро этой цепи, входящее в вершину v' в исходном дереве G и в его ветви $G(v)$, ориентируется в обоих этих деревьях с корнем одинаково, так, что вершина v' является его концом.

5.8.5. Типы вершин и центры деревьев

Пусть дано конечное дерево G . Назовем его концевые вершины *вершинами типа 1*. Отметим, что если дерево имеет более двух вершин, то среди них есть неконцевые вершины. Действительно, пусть v_1 и v_2 — концевые вершины. Они должны быть связаны цепью. Если эта цепь состоит из одного ребра, то v_1 и v_2 не связаны ни с какими другими вершинами, что противоречит связности дерева; если же эта цепь состоит из двух или более ребер, то она проходит через вершины степеней большей или равной двум, т. е. неконцевые вершины.

Удалим из дерева G все вершины типа 1 и инцидентные им концевые ребра. Останется связный граф G' без циклов, т.е. дерево. Действительно, для любых вершин $v', v'' \in G'$ связывающая их в дереве G цепь $L(v', v'')$ не может проходить через концевые ребра и целиком лежит в G' . Дерево G' также имеет концевые вершины, которые будем называть *вершинами типа 2* в дереве G . Аналогично определяются вершины типов 3, 4 и т. д.

Легко видеть, что в конечном дереве G имеются вершины лишь конечного числа типов, причем число вершин максимального типа равно единице или двум, так как в соответствующем дереве каждая вершина является концевой. Каждая вершина дерева G , кроме концевых, имеет соседей на единицу меньшего типа. Действительно, в дереве $G^{(k-1)}$ определяющем предыдущий тип, она не была концевой, а в $G^{(k)}$ стала. Значит, при построении этого дерева были удалены некоторые инцидентные этой вершине концевые ребра дерева $G^{(k-1)}$ и другие их концы — соседние с рассматриваемой вершиной вершины типа $k-1$. Остальные инцидентные рассматриваемой вершине ребра, кроме, может быть, одного, были удалены раньше, и соответствующие им соседние вершины имеют меньший тип. Одно же инцидентное ребро остается, если только рассматриваемая вершина — не

единственная, имеющая максимальный тип. Другой конец этого ребра — вершина старшего типа, если тип рассматриваемой вершины не максимальный, и вершина того же типа — в противном случае (рис. 20, цифры у вершин означают их типы).

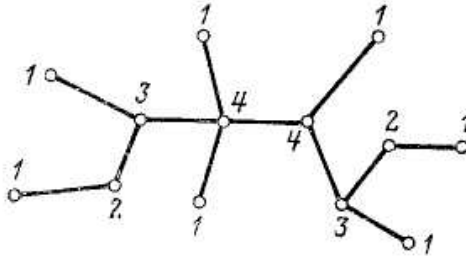


Рис.20

Теорема 1. *Центрами дерева являются вершины максимального типа k и только они.*

Любая цепь L с началом в вершине v_0 максимального типа идет последовательно по вершинам уменьшающихся типов. Ее первое ребро (v_0, v_1) ведет либо в вершину v_1 ранга k , либо в вершину меньшего типа. Следующие ребра (v_1, v_2) , (v_2, v_3) и т. д. уже обязательно ведут в вершины меньшего типа, так как единственная соседняя вершина большего или равного типа — это вершина, инцидентная уже использованному ребру. Таким образом, для длины l цепи L с началом в вершине максимального типа k имеем: $l(L) \leq k$, если есть еще одна вершина типа k ; $l(L) \leq k - 1$, если такая вершина единственна. Так как у каждой вершины дерева есть хотя бы одна соседняя, тип которой на 1 меньше, то из вершины максимального типа k можно построить путь длины $k - 1$. Следовательно, максимальное удаление от вершины v_0 максимального типа в дереве равно ее типу k или $k - 1$. Остается показать, что если вершина v имеет не максимальный тип, она является началом более длинной цепи. Построим сначала цепь $L(v, v_0)$, связывающую эту вершину с вершиной максимального типа k . Если v_0 — единственная вершина типа k , она связана по крайней мере с двумя вершинами типа $k - 1$, так как в графе $G^{(k-1)}$ не является концевой. Следовательно, можно построить цепь $L(v_0, v')$ длины $k - 1$, не проходящую через последнее ребро цепи $L(v, v_0)$ и не имеющее с этой цепью общих вершин (иначе в дереве G был бы цикл). Цепь

$$L = L(v, v_0) \cup L(v_0, v')$$

имеет длину большую, чем $k - 1$.

Если в дереве G есть две вершины v_0 и v_x максимального типа k , то тоже можно построить связывающую v с v_0 цепь $L(v, v_0)$. В случае,

когда эта цепь проходит через другую вершину v_l максимального типа k , ее длина не меньше двух. Тогда вершина v_0 является началом цепи $L(v, v')$ длиной $k - 1$ с первым ребром (v_0, v'') , ведущим в вершину v'' типа $k - 1$. Эта цепь не имеет с первоначально построенной цепью $L(v, v_0)$ общих вершин, иначе в дереве G был бы цикл. Следовательно,

$$L = L(v, v_0) \cup L(v_0, v')$$

— цепь длиной, не меньшей $k + 1$, с началом в рассматриваемой вершине v . Наконец, если цепь $L(v, v_0)$ не проходит через другую вершину v , максимального типа, длину $k + 1$ или больше имеет начинающаяся в вершине v цепь

$$L = L(v, v_0) \cup (v_0, v_1) \cup L(v_1, v''),$$

где цепь $L(v_1, v')$ длиной $k - 1$ строится так же, как в предыдущем случае цепь $L(v_0, v')$.

Итак, для любой вершины немаксимального типа ее максимальное удаление больше k .

Из теоремы непосредственно следует, что дерево имеет либо один, либо два центра.

Диаметральные цепи в деревьях проходят через его центр v_0 или через два центра v_0 и v_l , если их два. В первом случае длина диаметральной цепи равна $2k - 2$, а во втором $2k - 1$, где k —максимальный тип вершин дерева G (подразумевается, что оно конечное). Такая цепь строится из двух радиальных цепей $L(v_0, v')$ и $L(v_0, v'')$ или $L(v_0, v')$ и $L(v_l, v'')$ длиной $k - 1$, не имеющих общих вершин и существующих, как это показано ранее. Когда в дереве G два центра — v_0 и v_l эти цепи соединяются ребром (v_0, v_l) и получается цепь длиной $2k - 1$.

Докажем, что более длинных цепей в дереве G нет. Пусть L —какая-либо цепь, v_0, v_1, \dots, v_h — последовательность ее вершин, k_0, k_1, \dots, k_h — их типы. Вначале типы вершин могут расти. Пусть $k_0 < k_1 < \dots < k_g (g \leq h)$. Число вершин в этом участке последовательности вершин не превосходит максимального типа k , а число соединяющих их ребер пути L не больше $k - 1$. Соседних вершин одинакового ранга не больше двух, это центры дерева G , если их два. В последнем случае к участку возрастания типа вершин вдоль цепи L может добавиться одно ребро неубывания типа.

Если цепь L длиннее, возрастание типа вершин вдоль нее должно смениться убыванием. Пусть $v_{g'}$ — первая вершина цепи L , тип которой меньше типа предыдущей: $k_{g'-1} > k_{g'}$. Тогда тип следующей вершины еще меньше, и далее вдоль пути типы вершин убывают: $k_{g'-1} > k_{g'} > \dots > k_h$. Действительно, у вершины $v_{g'}$ только одна соседняя вершина имеет больший тип — это вершина $v_{g'-1}$, а все остальные соседние вершины, в том числе и вершина $v_{g'+1}$, имеют тип меньший, чем $k_{g'}$. Точно так

же у вершины $v_{g'+1}$ большого типа имеет только вершина v_g , а типы остальных соседей меньше $k_{g'+1}$ и т. д.

Таким образом, последовательность v_0, v_1, \dots, v_h можно разбить на участки возрастания типа v_0, v_1, \dots, v_g , его убывания $v_g, v_{g'+1}, \dots, v_h$ ($g' = g$ или $g + 1$) и, когда цепь L проходит через две вершины v_g и $v_{g'}$ максимального типа, участок постоянства типа. Длина участка убывания типа также не превосходит $k - 1$, а общая длина цепи L не больше $2k - 2$ или $2k - 1$ в случае, когда эта цепь проходит через два центра дерева G .

Остается доказать, что цепь L , не содержащая центра, имеет меньшую длину и не является диаметальной. Пусть $k' < k$ — максимальный ранг вершины v' , через которую проходит эта цепь. Тогда до этой вершины типы растут, а после нее падают. Следовательно, длина участков этой цепи до и после вершины v' не превышает $k' - 1$, а общая длина цепи L не больше $2k' - 2 < 2k - 2$. Если в дереве G два центра — v_0 и v_h , а цепь L проходит только через один из них, ее длина не больше $2k - 2$ и она также не является диаметальной.

5.8.6. Характеристики графов

Решение многих технических задач методами теории графов сводится к определению тех или иных характеристик графов. Знакомство с важнейшими характеристиками графов может оказаться полезным при изучении других дисциплин.

Цикломатическое число. Пусть G — конечный неориентированный граф. Его *цикломатическим числом* называется

$$\gamma(G) = v_c + v_e - v_v,$$

где v_c — число связных компонент графа; v_e — число его ребер, а v_v — число вершин. Цикломатическое число дерева равно нулю, цикломатическое число леса — сумме цикломатических чисел своих связных компонент-деревьев, т. е. также равно нулю. Цикломатические числа остальных конечных графов положительны. Так как цикломатическое число несвязного графа равно сумме цикломатических чисел связных компонент, достаточно рассмотреть связный граф G .

Если этот граф — дерево, его цикломатическое число равно нулю. В противном случае в графе G есть цикл $Z (e_1, e_2, \dots, e_h)$. Можно выбросить любое ребро этого цикла, и граф останется связным. Действительно, пусть $M (e'_1, e'_2, \dots, e'_i)$ — маршрут, связывающий вершины $v', v'' \in G$ и содержащий выброшенное ребро e_q цикла Z (если маршрут не содержит такого ребра, он связывает вершины v' и v'' и в новом графе G'). Это ребро можно заменить маршрутом

$$A' (e_q, e_{q+1}, \dots, e_h, e_1, \dots, e_{q-1}),$$

имеющим те же концы, что и само ребро.

Если в G' еще имеются циклы, можно выбросить следующее ребро, не нарушая связности, и т. д. В конце концов получится неориентированный связный граф G^γ без циклов, т. е. дерево (так как число вершин всех конструируемых графов G', G'', \dots, G^γ не меняется, если оно больше единицы, то все ребра выбросить нельзя). Число оставшихся ребер на единицу меньше числа вершин графа G^γ , а значит, и исходного графа G , но в последнем на γ ребер больше. Следовательно,

$$\gamma(G) = v_c(G^\gamma) + v_e(G^\gamma) - v_v(G^\gamma) + \gamma = \gamma > 0.$$

Хроматическое число. Пусть p — натуральное число. Граф G называется p -хроматическим, если его вершины можно раскрасить p различными цветами так, чтобы никакие две смежные вершины не были раскрашены одинаково. Наименьшее число p , при котором граф является p -хроматическим, называется хроматическим числом графа и обозначается $\gamma(G)$.

Если $\gamma(G)=2$, то граф называется бихроматическим. Необходимым и достаточным условием того, чтобы граф был бихроматическим, является отсутствие в нем циклов нечетной длины. Хроматическое число играет важную роль при решении задачи наиболее экономичного использования ячеек памяти при программировании. Однако его определение, за исключением случая бихроматического графа, представляет собой довольно трудную задачу.

Множество внутренней устойчивости. Множество $S \subseteq V$ графа $G=(V, E)$ называется внутренне устойчивым, если никакие две вершины из S не смежны, т.е. для любого $v \in S$ имеет место $E_v \cap S = \emptyset$.

Множество внутренней устойчивости, содержащее наибольшее число элементов, называется наибольшим внутренне устойчивым множеством, а число элементов этого множества называется числом внутренней устойчивости графа G . Наибольшее внутренне устойчивое множество играет важную роль в теории связи.

Множество внешней устойчивости. Множество $T \subset V$ графа $G=(V, E)$ называется внешне устойчивым, если любая вершина, не принадлежащая T , соединена дугами с вершинами из T , т. е. для любого $v \notin T$ имеет место $E_v \cap T \neq \emptyset$.

Множество внешней устойчивости, содержащее наименьшее число элементов, называется наименьшим внешне устойчивым

множеством, а число элементов этого множества называется числом внешней устойчивости графа G .

5.8.7. Максимальные графы исключения

Пусть дано семейство $\{F_i\}_1^s$ частей (называемых запрещенными) некоторого графа G^1 . Часть H этого графа начинается *графом исключения*, если она не содержит ни одной части из этого семейства, и *максимальным графом исключения*, если она не вложена в другой граф исключения. Обычно считается, что пустой граф \emptyset не принадлежит семейству запрещенных частей $\{F_i\}_1^s$. Иначе графов исключения нет. Если графы исключения существуют, то и максимальные графы исключения существуют. Когда рассматриваемый граф конечный, цепочка графов исключения, в которой каждый следующий содержит все предыдущие, должна оборваться, так как переход к следующему графу исключения в этой цепочке связан с добавлением новых элементов графа G , а их количество конечно. Следовательно, цепочка должна оборваться и последний входящий в нее граф — это максимальный граф исключения.

Отсюда следует, например, что любой граф G содержит максимальные части без циклов (семейство $\{Z_i\}$ состоит из конечных частей графа G). Можно также ограничить степени $\rho_H(v)$ вершин $v \in G$ в рассматриваемых графах исключения заданными значениями $k(v)$. Для этого семейство $\{F_i\}$ запрещенных частей должно включать все звездные графы $s(v)$ графа G с $(k(v) + 1)$ -м ребром.

5.8.8. Максимальные паросочетания и устойчивые множества вершин в графе

Пусть для всех вершин v графа G $k(v) = 1$. Тогда графы исключения $H \subset G$ состоят из изолированных вершин и пар вершин $v', v'' \in G$, связанных одним ребром $(v', v'') \in G$. Части H , содержащие только такие пары, называются *паросочетаниями* в графе G . Однако в отличие от приведенной ранее терминологии *максимальным паросочетанием* в конечном графе G называется не просто паросочетание, которое не вложено ни в какое другое, а содержащее максимально возможное число ребер.

Пусть теперь запрещенным множеством частей графа G является множество пар вершин $v', v'' \in G$, инцидентных какому-либо ребру этого графа. Тогда любой граф исключения H состоит из

изолированных вершин, не являющихся соседними в графе G . Такие множества вершин графа G называются *устойчивыми*. Как и паросочетание, устойчивое множество называется *максимальным*, если оно, содержит максимально возможное число вершин.

Рассмотрим две теоремы о максимальных графах исключения.

Теорема 1. *Если никакая конечная запрещенная часть F_i не имеет концевых ребер, то максимальный граф H покрывает все вершины графа G .*

Действительно, если некоторый граф исключения H' не содержит ребер, инцидентных некоторой вершине $v \in G$, эту вершину вместе с любым инцидентным этой вершине ребром можно добавить в часть H' . В полученной части H'' вершина v является концевой, а любая часть H'' либо не содержит v и тогда является частью H' , т. е. не принадлежит семейству $\{F_i\}$, либо содержит вершину v , являющуюся в этой части концевой (изолированные вершины тоже считаются концевыми), и не принадлежит семейству $\{F_i\}$, состоящему из частей без концевых вершин. Таким образом, часть H' не максимальна.

Теорема 2. *Пусть G — связный граф к $\{F_i\}_1^s$ — семейство его частей, каждая из которых по ребрам двусвязна. Тогда максимальные графы исключения связны и покрывают все вершины графа G .*

Граф G называется *двусвязным по ребрам*, если после удаления любого его ребра он остается связным. В двусвязном графе нет концевых вершин. Действительно, после удаления концевого ребра, единственного инцидентного такой вершине, она становится изолированной, а соответствующая часть G' графа G — несвязной. Следовательно, максимальный граф исключения H для семейства

двусвязных конечных частей $\{F_i\}_1^s$ покрывает все вершины графа G .

Пусть H — несвязный граф исключения, H_1 — его связная компонента. Последняя не покрывает всех вершин графа G . Значит, в любом маршруте $M(v', v'')$, соединяющем вершины $v' \in H_1$ и $v'' \in G \setminus H_1$, найдется ребро (v_1, v_2) с концами $v_1 \in H_1$ и $v_2 \in H_1$. Это ребро соединяет компоненту связности $H_1 \subset H$ с некоторой другой компонентой связности $H_2 \subset H$ (так как H покрывает все вершины графа G , $v_2 \in H$). Добавим ребро (v_1, v_2) к части H и получим объемлющую часть H' . Любая часть $H'' \subset H'$ либо не содержит ребра (v_1, v_2) , т. е. принадлежит графу исключения H и не является частью из семейства $\{F_i\}_1^s$, либо содержит это ребро и не является графом, двусвязным по ребрам. Действительно, после удаления этого ребра H'' становится несвязным графом, так как вершины v_1 и v_2 связаны только им. Значит

$H'' \notin \{F_i\}_1^s$ т. е. H' — тоже граф исключения и H — не максимален.

5.8.9. Максимальные деревья

Пусть G — связный граф, $\{Z_i\}$ — семейство всех его циклов. Тогда максимальный граф исключения T не содержит циклов. Он покрывает все вершины графа G , так как циклы Z_i не имеют концевых вершин, и связан, так как любой цикл двусвязан по ребрам. Следовательно, T — дерево. Будем называть его *максимальным деревом* в графе G .

Максимальное дерево в конечном связном графе G уже было раньше построено при определении цикломатического числа $\gamma(G)$ графа G . При этом было удалено $\gamma(G)$ ребер. Легко видеть, что все максимальные деревья T графа G имеют одно и то же число вершин, равное числу v_v вершин графа G , и число ребер, равное $v_v - 1$. Следовательно, при построении этих деревьев нужно удалить одинаковое число ребер, равное цикломатическому числу $\gamma(G)$ графа G . Пусть в графе G удалено $\gamma(G)$ ребер. Если остался связный граф H , его цикломатическое число уменьшилось на $\gamma(G)$ и стало равно нулю. Таким образом, при удалении произвольных $\gamma(G)$ ребер из конечного связного графа G получаем либо максимальное дерево, либо несвязный граф.

Задача о минимальном соединении. Пусть каждому ребру e конечного неориентированного связного графа G приписано некоторое действительное число $\mu(e)$ — *вес* этого ребра. *Минимальным соединением* называется максимальное дерево $T \subset G$ с минимальным общим весом

$$\mu(T) = \sum_{e \in T} \mu(e).$$

Задача о минимальном соединении внешне похожа на значительно более сложную *задачу о коммивояжере*, в которой требуется найти в полном неориентированном конечном графе K_n гамильтонов цикл Z минимального общего веса

$$\mu(Z) = \sum_{e \in Z} \mu(e).$$

В отличие от этой задачи минимальное соединение легко найти. Его построение начинается с выбора ребра e_1 минимального веса $\mu(e_1)$. На каждом следующем шаге к уже построенной части T_{i-1} добавляется ребро e_i такое, что часть $T_i = T_{i-1} \cup e_i$ не имеет циклов и из всех ребер, обладающих этими свойствами, e_i имеет минимальный вес $\mu(e_i)$. Если имеется несколько таких ребер, то можно выбрать любое из них.

Пусть лес T_{i-1} не покрывает всех вершин графа G . Тогда существует ребро e , инцидентное вершине $v \notin T_{i-1}$. Если добавление к T_{i-1} не приводит к появлению цикла (e — концевое ребро), и часть T_{i-1} можно

расширить. Если T_{i-1} покрывает все вершины G , но несвязен, его тоже можно расширить. Действительно, в маршруте $M(v', v'')$, соединяющем в графе G вершины v' и v'' из различных компонент T_{i-1} , найдется хотя бы одно ребро (v_1, v_2) , концы которого принадлежат разным компонентам леса T_{i-1} . В графе $\hat{T}_i = T_{i-1} \cup (v_1, v_2)$, это ребро является *разделяющим* (т. е. после удаления граф T_i становится несвязным) и через него не проходит никакой цикл. Нет в этом графе и циклов, не проходящих через ребро (v_1, v_2) , они принадлежали бы лесу T_{i-1} . Следовательно, пока не возникнет дерево T_{n-1} , покрывающее все вершины графа G (n — их количество), процесс расширения части T_i продолжается.

Отметим, что все ребра дерева T_{n-1} можно занумеровать в порядке их включения в эту часть графа G . Пусть T — дерево минимального общего веса, покрывающее вершины графа G . Оно существует, так как в конечном графе количество покрывающих деревьев конечно. Рассмотрим ребро $e_i \in T_{n-1} \setminus T$ с минимальным номером i . Граф

$T'' = T \cup e_i$ имеет цикломатическое число 1, т. е. не является деревом. В нем имеется некоторый цикл Z и другого цикла Z' быть не может. Действительно, в последнем имелось бы ребро e' , не принадлежащее Z , и, выбросив его из T' , мы получили бы связный граф T''' с цикломатическим числом 0 и циклом Z . Наконец, цикл Z проходит через ребро e_i : в дереве $T = T'' \setminus e_i$ циклов нет.

В дереве T_{n-1} нет циклов, значит Z содержит ребро

$$e' \notin T_{n-1}.$$

Тогда граф

$$T_1 = T' \setminus e' = T \cup e_i \setminus e'$$

покрывает все вершины G и не имеет циклов, т. е. является максимальным деревом в графе G . Так как дерево T имеет минимальный общий вес,

$$\mu(T_1) - \mu(T) = \sum_{e \in T'} \mu(e) - \sum_{e \in T} \mu(e) = \mu(e_i) - \mu(e') \geq 0.$$

Однако неравенство весов $\mu(e_i)$ и $\mu(e')$ невозможно, иначе при построении T_i вместо e_i было бы выбрано ребро e' . Действительно, все ребра T_{i-1} принадлежат дереву T , принадлежит ему и ребро e' . Значит,

$$T' = T_{i-1} \cup e' \subset \hat{T}$$

и в этой части графа G нет циклов.

Итак,

$$\mu(e_i) = \mu(e') \text{ и } \mu(T_1) = \mu(T),$$

т. е. T_1 — дерево минимального общего веса, максимальное в графе G . В нем больше общих ребер с деревом T_{n-1} , чем в дереве T . Повторив

при необходимости такую замену ребер, мы придем в конце концов к дереву T_{n-1} , и докажем, что

$$\mu(T_{n-1}) = \mu(T),$$

т. е. мы построили максимальное в графе G дерево T_{n-1} минимального общего веса.

5.8.10. Двудольные графы

Граф G называется двудольным, если множество его вершин V распадается на два непересекающихся подмножества V' и V'' , таких, что каждое ребро графа G имеет один конец из V' , а другой — из V'' . Двудольные графы рассматриваются во многих задачах дискретной математики. Одна из них — *задача о различных представителях подмножеств*. Пусть дано множество W и семейство его подмножеств $\{U_i\}$, требуется в каждом множестве этого семейства выбрать некоторый элемент u_i — представитель этого множества — так, чтобы разным множествам U_i и U_j соответствовали бы различные представители u_i и u_j .

Множество V' вершин двудольного графа G , соответствующего этой задаче, — это множество всех элементов W ; множество V'' состоит из элементов v''_i , соответствующих подмножествам u_i рассматриваемого семейства. Вершины $w \in W$ и $v''_i \in V''$ являются концами ребра (w, v''_i) в том и только том случае, когда $w \in U_i$. Требуется найти *максимальное паросочетание* (нерасширяемое множество ребер (w, v''_i) , в котором все вершины w и v''_i различны), и проверить, каждая ли вершина $v_i \in V''$ является концом некоторого ребра из этого паросочетания.

Другой пример. Пусть заданы два произвольных множества V' и V'' элементов разной природы (тем самым эти множества не имеют общих элементов) и отношение $v'Rv''$, являющееся истинным или ложным для всех пар $(v', v'') \in V' \times V''$. Тогда это отношение определяет двудольный граф G : ребро (v', v'') принадлежит G , если отношение $v'Rv''$ истинно.

Теорема 1. *Граф G является двудольным тогда и только тогда, когда вес его циклы имеют четную длину.*

Для цикла Z двудольного графа это условие выполняется: вершины, через которые этот цикл проходит, поочередно принадлежат множествам V' и V'' . Пусть теперь все циклы графа G имеют четную длину. Выберем вершину v_0 в некоторой связной компоненте G_0 графа G . Множество вершин G_0 распадается на множество V вершин с четным расстоянием от v_0 и множество V' вершин с нечетным расстоянием от v_0 . Если $v_1 \in V$, $v_2 \in V'$, то кратчайшие простые цепи $L_1(v_0, v_1)$ и $L_2(v_0, v_2)$ имеют четную длину. Предположим, что ребро

(v_1, v_2) существует. Пусть v' — последняя общая вершина для $L_1(v_0, v_1)$ и $L_2(v_0, v_2)$, т. е. $L_1(v_0, v_1) = L'(v_0, v') \cup L''(v', v_1)$, $L_2(v_0, v_2) = L'(v_0, v') \cup L'''(v', v_2)$ и начальные ребра L'' и L''' различны. Тогда цепь $L''(v', v_1) \cup (v_1, v_2) \cup L'''(v_2, v')$, где $L'''(v_2, v')$ обозначает цепь $L'''(v', v_2)$, пройденную в обратном порядке, является циклом нечетной длины, что противоречит предположению. Таким образом, никакие две вершины из V не соединены ребром; аналогичное утверждение доказывается и для V' . Поэтому всякое ребро из G_0 имеет один конец в V' , а другой — в V'' и G_0 — двудольный граф. Поскольку это верно для любой связной компоненты графа G , то и сам граф G — двудольный.

Максимальные двудольные части графа. Пусть запрещенное семейство частей неориентированного графа G состоит из простых циклов Z нечетной длины. Тогда графы исключения можно рассматривать как двудольные, а максимальные графы исключения — это максимальные двудольные части графа G .

Циклы нечетной длины не имеют концевых вершин и являются двусвязными по ребрам. В силу соответствующих общих теорем из этого следует, что максимальные двудольные части графа G покрывают все его вершины и в каждой связной компоненте этого графа лежит одна связная компонента максимальной двудольной части.

Для построения максимальной двудольной части графа G выберем в каждой его связной компоненте G_0 вершину v_0 . Для каждой вершины $v \in G_0$ можно определить ее ранг $r(v, v_0)$ относительно вершины v_0 , равный ее расстоянию от вершины v_0 . Удалим из G_0 все ребра (v', v'') , связывающие вершины v' и v'' рангов одинаковой четности. Так как все ребра кратчайшего пути $L(v_0, \dots, v)$, соединяющего вершину v_0 с вершиной $v \in G_0$, соединяют вершины соседних рангов i и $i + 1$, имеющих разную четность, они не будут удалены. Значит, часть G'_0 графа G_0 , состоящая из вершин $v \in G_0$ и неудаленных ребер, — связный граф. Легко видеть, что он двудольный: множество V' состоит из всех вершин четных рангов, множество V'' — из вершин нечетных рангов и каждое оставшееся ребро соединяет вершину из множества V' с вершиной из множества V'' . Значит, в нем нет циклов нечетной длины. Докажем, что, добавив любое удаленное ребро (v_1, v_2) , мы получим некоторый цикл нечетной длины. Пусть $r(v_1, v_0) = i$, $r(v_2, v_0) = j$. Числа i и j имеют одинаковую четность. Рассмотрим кратчайшие цепи $L_1(v_0, \dots, v_1)$ и $L_2(v_0, \dots, v_2)$, соединяющие вершину v_0 с вершинами v_1 и v_2 . Будем идти от вершины v_2 к v_0 по цепи L_2 , пока первый раз не попадем в вершину $v_h \in L_1$, в крайнем случае $v_h = v_0$. Тогда цикл, состоящий из отрезков $L'_1(v_h, \dots, v_i)$, $L'_2(v_h, \dots, v_j)$ путей L_1 и L_2 и

ребра (v_1, v_2) , является простым циклом нечетной длины. Действительно, все вершины этого цикла различны (а значит, различны и ребра), так как цепи L_1 и L_2 — кратчайшие, длина цепи L'_1 равна $i - h$, а длина цепи L'_2 равна $j - h$. Следовательно, длина цикла Z равна $(i - h) + (j - h) + 1$, т. е. нечетна, и G'_0 — максимальная двудольная часть графа G_0 . Из таких частей, построенных для каждой компоненты, максимальные двудольные части графа G можно составить 2^l способами, где l — число связных компонент графа G .

О максимальном паросочетании в двудольном графе. Пусть G — двудольный граф, $\{(v'_i, v''_i)\}$ ($i = 1, 2, \dots$) — паросочетание в нем, т. е. семейство ребер (v'_i, v''_i) , в котором все вершины $v'_i \in V'$ и $v''_i \in V''$ различны между собой. Если кроме вершин v''_i в множестве V'' имеется еще какая-либо вершина v'' , можно построить часть $S(v'')$ графа G следующим образом.

Каждая вершина множества $S(v'')$ принадлежит одному из множеств $V^{(i)}$ ($i = 0, 1, \dots$), определяемых условиями $V^{(0)} = \{v''\}$, для четного $i = 2j$:

$$V^{(2j)} = \{v' \mid (v', v'') \in G, v'' \in V^{(2j-1)} \setminus V^{(1)} \cup V^{(3)} \cup \dots \cup V^{(2j-1)}\};$$

для нечетного $i = 2j + 1$:

$$V^{(2j+1)} = \{v'' \mid (v', v'') \in \Pi, v' \in V^{(2j)} \setminus V^{(0)} \cup V^{(2)} \cup \dots \cup V^{(2j-2)}\},$$

где Π — рассматриваемое паросочетание. Будем говорить, что вершина $v \in S(v'')$ имеет индекс i , если $v \in V^{(i)}$. Множество ребер части $S(v'')$ состоит из всех ребер $(v', v'') \in G$, инцидентных вершинам v'' четных индексов. Легко видеть, что другой конец такого ребра принадлежит части $S(v'')$. Построение части $S(v'')$ изображено на рис. 21.

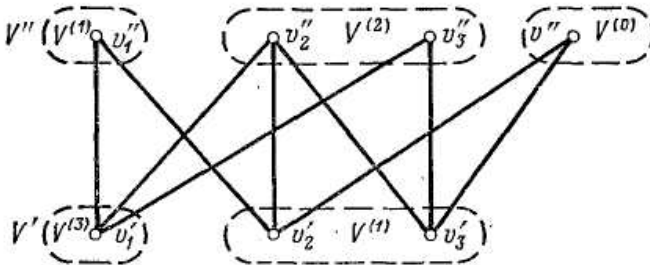


Рис. 21.

Таким образом, каждое ребро $(v_1, v_2) \in S(v'') \cap \Pi$ соединяет вершину v_1 нечетного индекса с вершиной v_2 следующего, а ребро (v_1, v_2) ,

соединяющее вершину v_2 четного индекса с вершиной v_1 следующего, не принадлежит паросочетанию Π . Это можно доказать по индукции.

Пусть $v' \in S(v'')$ — вершина нечетного индекса, не являющаяся концом никакого ребра из паросочетания Π . В эту вершину ведет *чередующаяся цепь* $L(e_1, e_2, \dots, e_{2j+1})$ с началом v'' , четные ребра которой принадлежат паросочетанию Π , а нечетные — не принадлежат. Тогда паросочетание Π можно расширить: выбросить из него j ребер e_2, e_4, \dots, e_{2j} и добавить $j+1$ ребер $e_1, e_3, \dots, e_{2j+1}$.

Если же паросочетание Π максимально, то все вершины $S(v'')$ нечетного индекса — это концы $v'_i \in V'$ некоторых ребер $(v'_i, v''_i) \in \Pi$, а множество вершин четного индекса состоит из других концов, v''_i этих ребер и вершины v'' . Так как все ребра, инцидентные вершинам v четного индекса из $S(v'')$, принадлежат этой части графа G , то в этом случае число вершин, $v \in S(v'') \cap V'$, имеющих в $S(v'')$ нечетный индекс и соседних в графе G вершинам $v \in S(v'') \cap V''$, на единицу меньше числа последних. Таким образом доказана *теорема Холла*.

Теорема 2. *Если максимальное паросочетание Π в конечном двудольном графе G не инцидентно всем вершинам из множества V'' , то существует подмножество последнего, для которого множество соседних вершин (лежащих в множестве V'') имеет на единицу меньшую мощность.*

В задаче о различных представителях множеств каждой вершине $v'' \in V''$ соответствует подмножество $U(v'')$ элементов множества V' , связанных в двудольном графе G ребрами с вершиной v'' . Для любого подмножества $\tilde{V}'' \subset V''$ множество соседних вершин — это объединение всех подмножеств $U(v'')$ для $v'' \in \tilde{V}''$.

Если число элементов

$$v' \in \bigcup_{v'' \in \tilde{V}''} U(v'')$$

меньше числа различных множеств $U(v'')$, входящих в это объединение, задачу о различных представителях, очевидно, нельзя решить. Теорема Холла показывает, что в противном случае, когда для

любого подмножества $\tilde{V}'' \subset V''$ выполнено условие

$$|V'| \leq \left| \bigcup_{v'' \in \tilde{V}''} U(v'') \right|,$$

задача о максимальных представителях решается.

5.9. Анатомия графов

5.9.1. Вводные замечания

При изучении структурных свойств (анатомии) графов удобно пользоваться их матричными представлениями.

Исходное описание графа дает его матрица инцидентности. Единственная неопределенность имеет место для петель, которым в матрице инцидентности соответствуют нулевые столбцы, но отсутствуют сведения о том, какие именно вершины инцидентны петлям. Для устранения этой неопределенности будем рассматривать графы без петель (мультиграфы) или не будем уточнять положение петель.

Можно также ограничиться рассмотрением связных графов, так как основные свойства легко обобщаются на случаи, когда граф состоит из нескольких компонент связности. Каждая такая компонента представляется своей матрицей инцидентности A_i ($i = 1, 2, \dots, k$), а общая матрица инцидентности A несвязного графа (при соответствующей группировке его вершин и ребер) имеет квазидиагональную форму:

$$A = \begin{bmatrix} A_1 & 0 & \dots & 0 \\ 0 & A_2 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & A_k \end{bmatrix}.$$

Сначала рассматриваются структурные свойства неориентированных графов, а затем они обобщаются на орграфы. Как будет видно из дальнейшего, эти два случая мало чем отличаются между собой.

Элементами матриц инцидентности неориентированных графов могут быть только нули и единицы. Поэтому все операции над элементами выполняются по модулю 2. Так, при сложении по модулю 2 нескольких чисел достаточно их арифметическую сумму разделить на два и остаток записать как результат такого сложения. При рассмотрении орграфов используются обычные правила, так как их матрицы инцидентности содержат в качестве своих элементов числа 0, 1 и -1 .

5.9.2. Свойства матрицы инцидентности.

Прежде всего, отметим очевидную зависимость между строками матрицы инцидентности A графа $G = (V, E)$. Так как каждый ее столбец содержит только два единичных элемента или состоит только из

нулей, если столбец соответствует петле, то сумма всех строк (по модулю 2) равна нулю. Это значит, что без потери информации вместо матрицы A можно рассматривать сокращенную матрицу A_0 , которая получается из A вычеркиванием любой строки (чаще вычеркивается последняя строка).

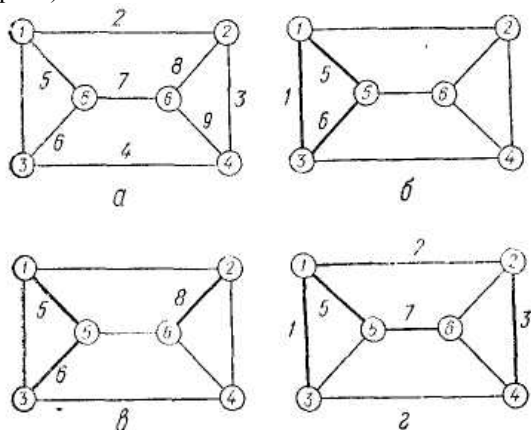


Рис. 22. Граф (а) и его суграфы (выделены жирными линиями): Циклический (б), ациклический (в) и покрывающее дерево (г).

Например, для графа на рис. 22, а имеем:

1 2 3 4 5 6 7 8 9

1	1			1					1
	1	1						1	2
1			1		1				3
		1	1						4
				1	1	1			5
						1	1	1	6

A

1 2 3 4 5 6 7 8 9

1	1			1					1
	1	1						1	2
1			1		1				3
		1	1						4
				1	1	1			5

A₀ =

Таким образом, из p строк матрицы A связного графа одна строка всегда линейно зависима, т. е. ранг матрицы A не может превышать $p-1$ (далее будет показано, что он в точности равен $p-1$).

Любое подмножество столбцов матрицы A можно рассматривать как матрицу инцидентности A' некоторого суграфа $G'=(V, E')$, содержащего все вершины V исходного графа и соответствующее выделенным столбцам подмножество $E' \subset E$ его ребер. При этом все столбцы A' линейно-независимы тогда и только тогда, когда суграф G не содержит циклов. Действительно, если совокупность ребер образует цикл (рис. 13, б), то каждая вершина инцидентна четному числу ребер этого цикла. Следовательно, сумма по модулю 2 соответствующих столбцов

(1, 5, 6) дает нулевой столбец, что означает их зависимость. Если же суграф не содержит циклов (рис. 13, в), то он имеет, по меньшей мере, пару (вообще, четное число) концевых вершин, каждая из которых инцидентна только одному (концевому) ребру. Поэтому сумма по модулю 2 соответствующих столбцов (5, 6, 8) будет содержать два (или четное число) единичных элементов и, следовательно, совокупность этих столбцов независима.

В связном графе с p вершинами всегда можно выделить $p - 1$ ребер так, чтобы они образовали суграф без циклов, представляющий собой дерево графа (рис. 13, з). Поэтому матрица инцидентности содержит не менее $p-1$ независимых столбцов. В то же время любой суграф, имеющий более $p-1$ ребер, обязательно содержит цикл, т. е. в матрице инцидентности не может быть больше $p - 1$ независимых столбцов. Отсюда следует, что матрица инцидентности связного графа содержит в точности $p-1$ независимых столбцов и значит ее ранг равен $p - 1$. Число $v = p - 1$ и определяет *ранг графа*.

5.9.3. Деревья и дополнения.

Из изложенного ясно, что совокупность $p - 1$ столбцов матрицы инцидентности линейно-независима, если соответствующие им ребра образуют дерево графа. И обратно, дерево графа соответствует совокупности $p - 1$ линейно независимых столбцов матрицы инцидентности. Остальные $q - p + 1$ столбцов соответствуют ребрам, которые образуют дополнение дерева

Рассматривая всевозможные сочетания по $p - 1$ из q столбцов матрицы A или A_0 и испытывая их на независимость, можно выявить все деревья графа. Однако такой путь нерационален и более удобным является алгоритм, изложенный в (5.8.2). Если же необходимо сформировать одно дерево, обладающее какими-либо свойствами (например, экстремальное дерево), то можно воспользоваться алгебраическим способом преобразования матрицы инцидентности.

Предварительно ребра графа нумеруются в том порядке, в каком их предпочтительно вводить в дерево (например, в порядке возрастающих весов). Затем, рассматривая поочередно столбцы матрицы инцидентности, необходимо выбрать $p - 1$ столбцов так, чтобы в совокупности они были линейно-независимы. Для этого используются операции над матрицей инцидентности, не нарушающие ее ранга — перестановка строк и столбцов, а также замена строки суммой по модулю 2 с другой строкой.

В результате в матрице A_n можно выделить единичную матрицу, столбцы которой будут соответствовать ветвям дерева. Если ис-

пользуется матрица A , то в конечном счете все элементы ее последней строки обращаются в нули и выделяется единичная матрица $(p-1)$ -го порядка. Эта процедура подобна алгоритму исключения Гаусса—Жордана с выбором опорных элементов по столбцам .

Например, преобразование матрицы A для графа рис. 22, a сводится к следующим операциям. Заменяем третью строку ее суммой с первой:

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

Заменяем первую и третью строки их суммами со второй строкой:

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

Далее заменяем первую, вторую и четвертую строки их суммами с третьей строкой:

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

В первых трех столбцах и строках образовались элементы единичной матрицы, однако в следующем столбце нельзя выбрать опорный элемент, так как в остальных строках этого столбца стоят нули. Поэтому четвертый столбец пропускается и рассматривается пятый. Заменяя вторую, третью и пятую строки их суммами с четвертой строкой, имеем:

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

Далее имеется возможность выбрать опорный элемент в седьмом столбце. Заменяя последнюю строку ее суммой с пятой строкой, получаем:

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

Как и следовало ожидать, последняя строка преобразовалась в нулевую (ее можно использовать для контроля и в дальнейшем отбросить). Пять столбцов, в которых имеется только по одному единичному элементу, составляют единичную субматрицу, а соответствующие им ребра (1, 2, 3, 5, 7) образуют дерево графа. Остальные столбцы (4, 6, 8, 9) соответствуют хордам, которые образуют дополнение дерева. Переставив столбцы так, чтобы слева выделилась единичная субматрица, результат преобразования матрицы инцидентности можно записать в виде:

II =

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

Суммы чисел справа у каждой строки приведенных матриц указывают на номера строк матрицы инцидентности, результатом суммирования которых является данная строка.

5.9.4. Разрезы.

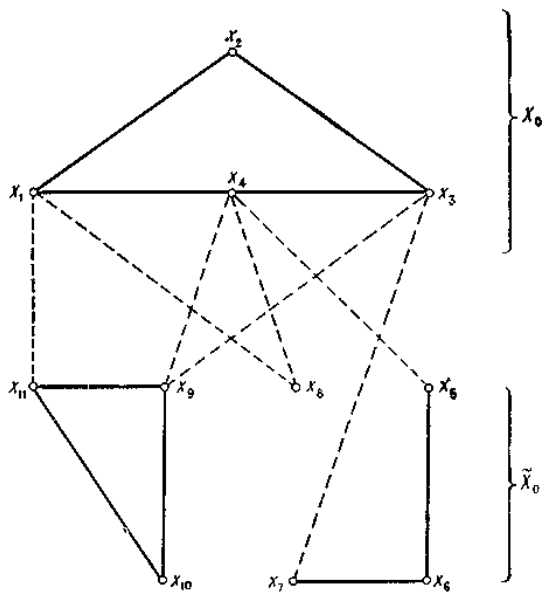
Во многих прикладных задачах требуется выделить в связном графе $G = (V, E)$ подмножество ребер $E' \subset E$, называемое *разрезом*, при удалении которых граф распадается на две или больше компонент.

Понятие разреза очень тесно связано с понятием цикла и определяется так.

Определение. Если вершины неориентированного графа $G = (V, E)$ разбиты на два множества V_0 и V_0^c (где $V_0 \subset V$ и V_0^c является дополнением V_0 относительно V), то множество ребер графа G , одни концевые вершины которых лежат в V_0 , а другие — в V_0^c , называется *разрезом* графа G .

Множество ребер разреза может быть представлено множеством вершин пары (V_0, V_0^c) . Таким образом, остовный подграф

$G_p = (V, E - (V_0, V_0^c))$, полученный из G удалением ребер, принадлежащих разрезу, является несвязным графом, состоящим по крайней мере из двух компонент. В графе на рис. 23 множество ребер, показанных пунктиром, образует разрез, определяемый множествами $V_0 = \{x_i / i = 1, \dots, 4\}$ и $V_0^c = \{x_i / i = 5, \dots, 11\}$.



----- разрез (V_0, V_0^0) .

Рис. 23

Получающийся остовный подграф состоит из четырех связных компонент:

$$\langle x_1, x_2, x_3, x_4 \rangle, \langle x_5, x_6, x_7 \rangle, \langle x_8 \rangle \text{ и } \langle x_9, x_{10}, x_{11} \rangle.$$

Множество ребер S , удаление которых из графа G дает несвязный остовный подграф $G_p = (V, E - S)$ и при этом не существует никакого собственного подмножества $S' \subset S$, такого, что граф $G'_p \equiv (V, E - S')$ также несвязен, называется *правильным разрезом*. Это множество можно также представить парой (V_0, V_0^0) , хотя теперь остовный подграф, полученный удалением правильного разреза из G , разбивает G *точно* на две связные компоненты, одна из которых имеет своими вершинами множество X_0 , а другая — множество V_0^0 . В общем случае каждый разрез является объединением некоторого числа правильных разрезов. В графе на рис. 23, например, разрез, показанный пунктиром, является объединением трех правильных разрезов:

$\{(x_1, x_{11}), (x_4, x_9), (x_3, x_9)\}$, $\{(x_1, x_8), (x_4, x_8)\}$ и $\{(x_3, x_7), (x_4, x_5)\}$. В остальной части этой главы, мы будем использовать слово разрез для обозначения правильного разреза, если только не оговорено противное.

Двойственность понятий остова и разреза графа G становится очевидной, если вспомнить, что остов определяется как минимальное множество ребер, связывающее все вершины графа G , в то время как разрез является минимальным множеством ребер, отделяющим одни вершины от других. Из этого замечания совершенно очевидно, что любой остов графа G должен иметь по крайней мере одно общее ребро с каждым правильным разрезом.

Разрез был определен выше для неориентированного графа. Если граф $G = (V, E)$ — ориентированный, то разрез графа G определяется как множество дуг, соответствующих ребрам неориентированного дубликата \bar{G} графа G . Некоторые из дуг разреза графа G будут ориентированы из V_0 в $V_0^{\%}$, и множество этих дуг будет записываться как $(V_0 \rightarrow V_0^{\%})$, в то время как множество дуг, ориентированных из вершин множества $V_0^{\%}$ в вершины множества V_0 , будет записываться как $(V_0^{\%} \rightarrow V_0)$. Поэтому разрез $(V_0, V_0^{\%})$, состоящий из дуг ориентированного графа, определяется как объединение

$$(V_0 \rightarrow V_0^{\%}) \cup (V_0^{\%} \rightarrow V_0).$$

Например для графа на рис. 24 множество дуг $\{(x_1, x_5), (x_2, x_8), (x_8, x_3), (x_4, x_8), (x_7, x_4)\}$ является разрезом, разделяющим множества вершин $V_0 = \{x_1, x_2, x_3, x_4\}$ и $V_0^{\%} = \{x_5, x_6, x_7, x_8\}$.

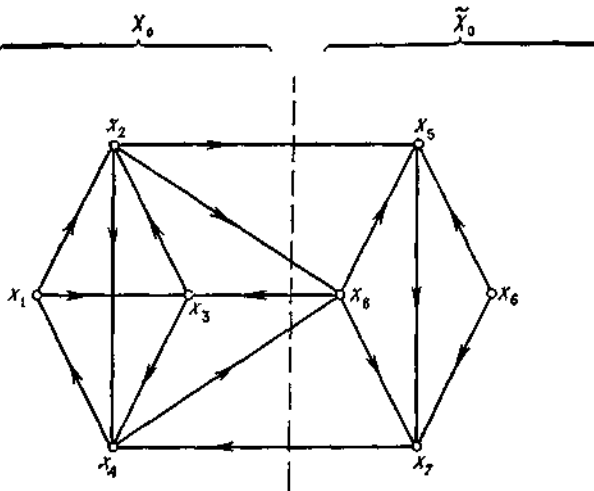


Рис. 24. Ориентированные разрезы.

Этот разрез образован дугами подмножеств

$$(V_0 \rightarrow \overset{\circ}{V}_0^k) = \{(x_2, x_5), (x_2, x_8), (x_4, x_8)\} \text{ и } (\overset{\circ}{V}_0^0 \rightarrow V_0) = \{(x_8, x_3), (x_7, x_4)\}.$$

Множество фундаментальных циклов неориентированного графа G было определено выше как множество $\nu(G)$ циклов, каждый из которых получается путем добавления какого-нибудь ребра, не принадлежащего остову T , к ребрам этого остова. Аналогичным образом, имея в виду отмеченную выше двойственность между разрезами и остовами, *фундаментальные разрезы* относительно остова T определяются как $n - 1$ разрезов, каждый из которых содержит одно, и только одно ребро, принадлежащее дереву T .

Следующая теорема устанавливает связь между фундаментальными разрезами и фундаментальными циклами и дает способ построения фундаментальных разрезов.

Теорема. Если T — остов неориентированного графа G , то фундаментальный разрез, определяемый ребром a_i из T , образован a_i и теми ребрами из G , не принадлежащими T , которые после добавления к T дают фундаментальные циклы, содержащие a_i .

Доказательство. Если из остова T удалить ребро a_i , то T распадается на два поддерева T_1 и T_2 (см. рис. 25).

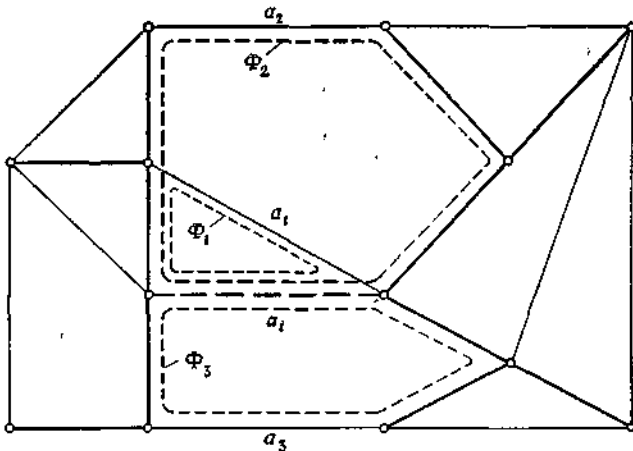


Рис. 25. Фундаментальные циклы, содержащие ребро a_i .

Любое ребро, одна концевая вершина которого лежит в T_1 , а другая в T_2 , должно принадлежать фундаментальному разрезу, так как добавление любого такого ребра к ребрам из T_1 и T_2 приводит к образованию другого остова графа G и, следовательно, любое множество, не содержащее таких ребер, не будет разрезом. Множество таких ребер вместе с ребром a_i является разрезом, так как их удаление разбивает

граф на два подграфа, один из которых имеет множеством своих вершин T_1 , а другой — T_2 . Значит, этот разрез является фундаментальным разрезом. Более того, так как ребро a_i является единственным ребром, через которое проходят цепи остова T , начинающиеся в вершинах из T_1 и оканчивающиеся в вершинах из T_2 , то единственными ребрами, замыкающими фундаментальные циклы, содержащие ребро a_i , будут те, одна концевая вершина которых лежит в T_1 , а другая в T_2 (таковы, например, ребра a_1 , a_2 и a_3 в графе на рис. 25). Это доказывает теорему.

Разрез называется *простым*, если никакое собственное подмножество его ребер не является разрезом данного графа. После удаления из графа ребер простого разреза образуется суграф, состоящий точно из ДВУХ компонент (компонентой такого суграфа может быть и изолированная вершина).

Графически разрез обычно выделяют замкнутой линией, которая пересекает принадлежащие ему ребра. При этом множество вершин V графа разбивается на два непустых подмножества V' и V''

($V' \cup V'' = V$, $V' \cap V'' = \emptyset$), связь между которыми осуществляется исключительно ребрами разреза. Так, простой разрез $E' = (2, 3, 7, 9)$, выделенный на рис. 26, *а* жирными линиями, разбивает множество вершин на подмножества $V' = \{1, 3, 4, 5\}$, и $V'' = \{2, 6\}$. Разрез

$E' = \{1, 2, 3, 4, 6, 7, 9\}$ на рис. 26, *б* не является простым, так как можно найти его подмножество, например $E'' = (1, 4, 6)$, которое также является разрезом.

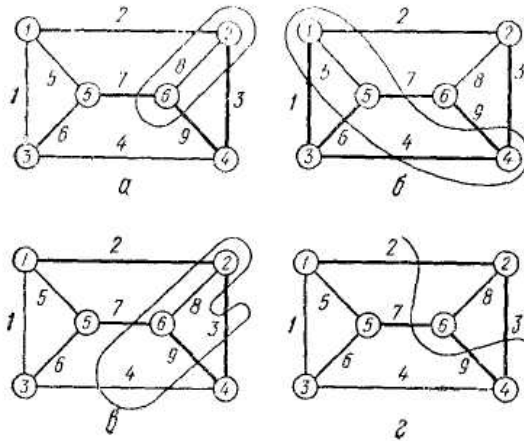


Рис.26. Разрезы графа (выделены жирными линиями): *а* — простым; *б* — непростым, *в*, *г* — варианты изображения простого разреза.

При выделении разреза замкнутой линией необходимо иметь в виду, что данному разрезу принадлежат только те ребра, которые пересекаются этой линией один (или нечетное число) раз. Ребра, которые имеют с выделяющей линией четное число пересечений, разрезу не принадлежат. Для упрощения разделяющую линию часто обрывают, считая условно, что она замыкается во внешней области графа. На рис. 26, *в* и *г* изображены варианты простого разреза (рис. 26, *а*), которые иллюстрируют эти положения

Совокупность ребер, инцидентных некоторой вершине графа, является разрезом с центром в этой вершине и называется *центральной*. В неразделимом графе каждый центральный разрез простой, удаление которого приводит к суграфу с изолированной вершиной. В разделимом графе совокупность ребер, инцидентных точке сочленения, образует разрез, который не является простым. При его удалении граф разбивается на три или больше компонент, одна из которых содержит только точку сочленения. Соответствующий пример показан на рис. 27.

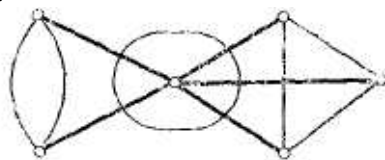


Рис. 27. Центральный разрез в точке сочленения разделимого графа.

В графе с p вершинами имеется p центральных разрезов, причем каждому из них соответствует строка инцидентности A . Единичные элементы строки указывают на совокупность ребер, образующих соответствующий разрез.

Аналогичное представление для любого разреза можно получить, суммируя по модулю 2 те строки матрицы инцидентности, которые соответствуют вершинам одного из подмножеств V' или V'' , на которые данный разрез разбивает множество вершин графа. Ясно, что безразлично, каким из этих двух подмножеств руководствоваться, поскольку в силу зависимости строк матрицы инцидентности (сумма их равна нулевой строке) в обоих случаях получим один и тот же результат. Однако удобно выбирать подмножество, которое содержит меньше вершин. Так, простой разрез (см. рис. 4.2,а) представляется суммой второй и шестой (или первой, третьей, четвертой и пятой) строк:

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

Таким образом, любой разрез можно рассматривать как объединение некоторой совокупности центральных разрезов.

5.9.5. Матрицы циклов и разрезов

Пусть T — остов неориентированного графа. Матрицей фундаментальных циклов графа G называется матрица $\Phi = [\varphi_{ij}]$, состоящая из $\nu(G)$ строк и m столбцов, в которой φ_{ij} равно 1, если ребро a_j принадлежит циклу Φ_i , и равно 0 в противном случае. Если ребра, не принадлежащие дереву T , пронумеровать последовательно от 1 до $\nu(G)$, а ребра дерева T пронумеровать от $\nu(G) + 1$ до m , то матрица циклов Φ будет иметь вид

$$\Phi = (\mathbf{I} \mid \Phi_{12}),$$

где \mathbf{I} — единичная матрица. Это объясняется тем, что каждый цикл Φ_i содержит одно, и только одно ребро, не принадлежащее остову T , и циклы всегда можно пронумеровать по числу принадлежащих им внеостовных ребер, вследствие чего все единицы в первой $(\nu(G) \times \nu(G))$ -подматрице матрицы Φ лежат на диагонали.

Матрица фундаментальных разрезов $\mathbf{K} = [k_{ij}]$ определяется аналогично — как матрица с $\nu - 1$ строками и m столбцами, где k_{ij} и есть 1, если ребро a_j принадлежит разрезу K_i , и 0 в противном случае. При той же самой нумерации ребер, что и выше, матрица \mathbf{K} имеет вид

$$\mathbf{K} = (\mathbf{K}_{11} \mid \mathbf{I}),$$

так как теперь каждый фундаментальный разрез содержит одно, и только одно ребро из ребер дерева T .

В графе на рис. 28 возьмем остов, изображенный жирными линиями.

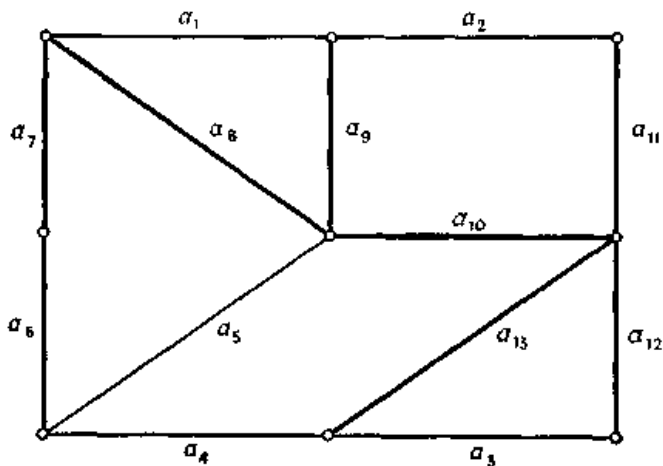


Рис. 28.

Тогда матрица фундаментальных циклов равна

	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}	a_{11}	a_{12}	a_{13}
Φ_1	1	0	0	0	0	0	0	1	1	0	0	0	0
Φ_2	0	1	0	0	0	0	0	0	1	1	1	0	0
Φ_3	0	0	1	0	0	0	0	0	0	0	0	1	1
Φ_4	0	0	0	1	0	1	1	1	0	1	0	0	1
Φ_5	0	0	0	0	1	1	1	1	0	0	0	0	0

9

матрица фундаментальных разрезов равна

	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}	a_{11}	a_{12}	a_{13}
K_1	0	0	0	1	1	1	0	0	0	0	0	0	0
K_2	0	0	0	1	1	0	1	0	0	0	0	0	0
K_3	1	0	0	1	1	0	0	1	0	0	0	0	0
K_4	1	1	0	0	0	0	0	0	1	0	0	0	0
K_5	0	1	0	1	0	0	0	0	0	1	0	0	0
K_6	0	1	0	0	0	0	0	0	0	0	1	0	0
K_7	0	0	1	0	0	0	0	0	0	0	0	1	0
K_8	0	0	1	1	0	0	0	0	0	0	0	0	1

где разрезы K_i , соответствующие ребрам остова, имеют номера $v(G) + i = 5 + i, i = 1, 2, \dots, 8$.

Существует несколько соотношений между матрицами циклов, разрезов и инцидентностей/ Ниже все арифметические операции являются операциями по модулю 2 Здесь мы будем рассматривать неориентированные графы без петель.

Теорема 1. *Матрица инцидентностей \mathbf{B} и транспонированная матрица фундаментальных циклов $\mathbf{\Phi}^t$ ортогональны, т. е. $\mathbf{B} \cdot \mathbf{\Phi}^t = 0$.*

Теорема 2. *Матрица фундаментальных циклов $\mathbf{\Phi}$ и транспонированная матрица фундаментальных разрезов \mathbf{K}^t ортогональны, т. е. $\mathbf{\Phi} \cdot \mathbf{K}^t = 0$.*

Эти две теоремы являются немедленным следствием двух очевидных фактов:

- (1) Каждая вершина в цикле инцидентна четному числу ребер (а в случае простого цикла — двум ребрам) этого цикла
- (2) Каждый разрез цикла, индуцированный некоторым разрезом, имеет четное число ребер, общих с этим разрезом

Теорема 1 следует из теоремы раздела (5.9.4), а теорема 2 — из (1), если вспомнить, что все операции рассматриваются по модулю 2, так что, например, $2 = 0 \pmod{2}$. Кроме того, существуют доказательства этих теорем, не использующие предположения о фундаментальности циклов и разрезов. Следовательно, сформулированные теоремы справедливы для любых матриц циклов и разрезов. Последние определяются аналогичным образом.

По теореме 2 можно написать

$$\mathbf{\Phi} \mathbf{K}^t = (\mathbf{I} \mid \mathbf{\Phi}_{12}) \left(\begin{array}{c} \mathbf{K}_{11}^t \\ \mathbf{I} \end{array} \right) = \mathbf{K}_{11}^t + \mathbf{\Phi}_{12} = 0.$$

Поэтому

$$\mathbf{K}_{11}^t = -\mathbf{\Phi}_{12} = \mathbf{\Phi}_{12}, \tag{20}$$

так как $-1 = 1 \pmod{2}$. Вследствие этого матрица фундаментальных разрезов может быть немедленно получена, как только известна матрица фундаментальных циклов, и наоборот. Справедливость соотношения (20) можно проверить на приведенных выше матрицах $\mathbf{\Phi}$ и \mathbf{K} для графа, изображенного на рис. 28.

5.9.6. Матрица сечений.

Так как из p строк матрицы инцидентности связного графа только $p-1$ линейно-независимы, то число независимых центральных разрезов также равно $p-1$. При замене любой строки матрицы инцидентности суммой ее с другими строками ранг этой матрицы не изменяется.

Поэтому после такой операции $p-1$ ее строк также будут соответствовать некоторой совокупности независимых разрезов. Например, на рис. 29 показаны независимые разрезы, соответствующие преобразованиям матрицы инцидентности графа (см. рис. 22, а).

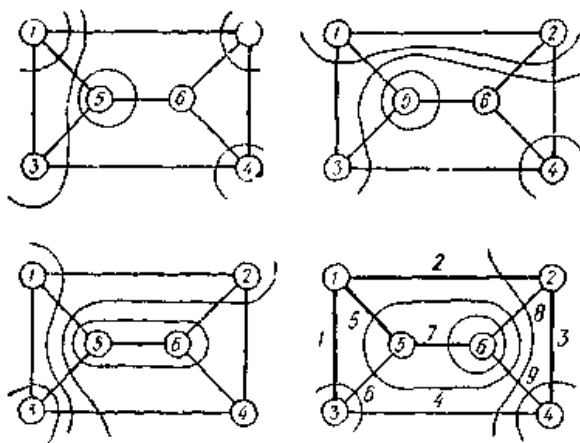


Рис. 29. Независимые разрезы, соответствующие преобразованиям матрицы инцидентности графа рис. 22, а.

На последнем шаге получаем совокупность независимых разрезов, определяемых деревом графа. Каждый из них определяется строкой результирующей матрицы Π и, как видно из этой матрицы, содержит одну и только одну ветвь дерева, не входящую в остальные разрезы. Такие разрезы называются *главными разрезами*, а в приложениях их называют *сечениями*.

Итак, дерево графа однозначно определяет множество сечений, которое представляется в аналитической форме матрицей Π , называемой *матрицей сечений*. Между ветвями дерева и сечениями имеет место взаимно-однозначное соответствие: каждая ветвь содержится в соответствующем ей сечении и каждое сечение содержит соответствующую ему ветвь. Поэтому строки матрицы Π часто обозначают номерами тех ветвей, которые определяют сечения (их можно также нумеровать порядковыми числами). Так, матрицу сечений, приведенных на рис. 29, запишем следующим образом:

$$\Pi =$$

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

Если столбцы, соответствующие ветвям дерева, объединены в единичную матрицу, как в приведенной выше записи, то матрица сечений представляется в канонической форме:

$$\Pi = [1, \pi],$$

где π - матрица сечений для хорд размера $(p - 1) \times (q - p + 1)$.

Следует заметить, что каноническая форма является удобной, но не обязательной при записи матрицы сечений. При расположении столбцов в порядке следования их номеров приведенная выше матрица запишется в виде:

$$\Pi =$$

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

Ребра и сечения связаны отношением инцидентности: содержащиеся в некотором сечении ребра инцидентны данному сечению, а содержащие некоторое ребро сечения инцидентны данному ребру. Ветвь дерева

инцидентна только своему сечению и называется *определяющей ветвью*.

Так как множество сечений представляет собой независимую совокупность $p-1$ разрезов, то любой разрез можно получить объединением соответствующих сечений. Например, разрез на рис.

26, а образуется объединением сечений 2,3 и 7. Различные деревья порождают и различные совокупности независимых сечений, число которых определяется так же, как и число различных деревьев графа.

5.9.7. Матрица контуров.

Любая из $q - p + 1$ хорд образует совместно с некоторой совокупностью ветвей дерева простой цикл (рис. 30, а). Так как каждый из этих циклов содержит, по крайней мере, одно ребро (хорду), которое не входит в другие циклы, то образованная таким способом совокупность $q - p + 1$ циклов является линейно-независимой. Число $\sigma = q - p + 1$ называется *цикломатическим числом* графа.

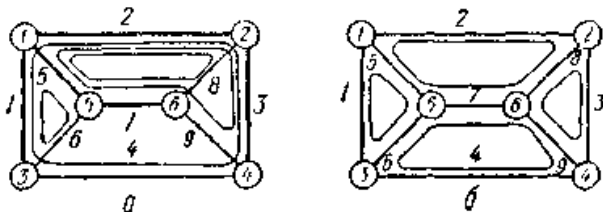


Рис. 30. Независимые циклы графа:
а — главные циклы (контуры); б — ячейки

Можно показать, что любой цикл графа представляется как объединение таких простых циклов, называемых *главными циклами*. В приложениях их обычно называют *контурами*, но не следует смешивать этот термин с контуром орграфа, в котором обход совершается по направлениям дуг.

Ребра и контуры связаны отношением инцидентности: содержащиеся в некотором контуре ребра инцидентны данному контуру, а содержащие некоторое ребро контуры инцидентны данному ребру. Хорда инцидентна только своему контуру и называется *определяющей хордой*.

Аналитическое представление совокупности контуров (p, q) -графа дает *матрица контуров* размера $(q - p + 1) \times q$, строки которой соответствуют контурам, а столбцы — ребрам. Обычно контурам присваиваются номера определяющих их хорд, либо они нумеруются

порядковыми числами. Инцидентность i -го контура и j -го ребра отмечается в матрице контуров единицей в ij -клетке, а нулевые элементы означают, что соответствующие контуры и ребра не инцидентны. Например, матрица контуров, приведенных на рис. 29, а, имеет вид:

$$P = \begin{array}{c|cccccccc|c} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & \\ \hline 4 & 1 & 1 & 1 & 1 & & & & & & \\ \hline 6 & 1 & & & & 1 & 1 & & & & \\ \hline 8 & & 1 & & & 1 & & 1 & 1 & & \\ \hline 9 & & 1 & 1 & & 1 & & 1 & & 1 & \\ \hline \end{array}$$

Столбцы матрицы контуров, соответствующие хордам, содержат по одному единичному элементу в различных строках. Поэтому их можно объединить в единичную матрицу. Так, для рассматриваемого примера:

$$P = \begin{array}{c|cccc|cc|c} & 1 & 2 & 3 & 5 & 7 & 4 & 6 & 8 & 9 & \\ \hline 4 & 1 & 1 & 1 & & & 1 & & & & \\ \hline 6 & 1 & & & 1 & & & 1 & & & \\ \hline 8 & & 1 & & 1 & 1 & & & 1 & & \\ \hline 9 & & 1 & 1 & 1 & 1 & & & & 1 & \\ \hline \end{array}$$

Таким образом, в канонической форме матрица контуров имеет вид:

$$P = [\rho, I],$$

где ρ — матрица контуров для ветвей дерева размера $(q-p+1) \times (p-1)$. Совокупность $q-p+1$ независимых циклов образуют также ячейки плоского графа. Каждый из таких циклов содержит ребра, охватывающие области, на которые граф разбивает плоскость (рис. 29, б).

5.9.8. Связь между топологическими матрицами.

Матрицы сечений и контуров отражают структурные свойства (топологию) графа, в связи с чем их можно назвать *топологическими матрицами*. Для данного графа вид этих матриц определяется выбранной совокупностью независимых разрезов и циклов. Если главные разрезы и циклы (сечения и контуры) порождаются одним и тем же деревом, то оно называется *фундаментальным деревом* графа. При этом между топологическими матрицами имеется взаимная связь, позволяющая по одной из них определить другую.

В самом деле, столбец матрицы π для некоторой хорды содержит единичные элементы в строках тех сечений, определяющие ветви которых образуют с одной хордой контур (рис. 30, а).

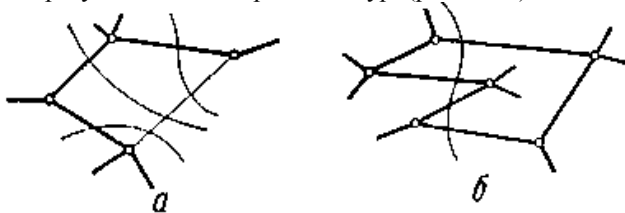


Рис. 30. Инцидентность ребер и разрезов:

а — совокупность сечений, инцидентных хорде;

б — четность числа ребер цикла, инцидентных разрезу.

Иначе говоря, столбцы матрицы π отображают инцидентность ветвей контурам, которые определяются соответствующими этим столбцам хордами. Поэтому столбцы матрицы π играют роль строк матрицы ρ , т. е.

$$\rho = \pi^t; \quad \pi = \rho^t,$$

а также

$$P = [\pi^t, 1]; \quad \Pi = [1, \rho^t].$$

Отсюда, в частности, следует (при операциях по модулю 2):

$$\Pi P^t = [1, \pi] \begin{bmatrix} \rho^t \\ 1 \end{bmatrix} = [1, \pi] \begin{bmatrix} \pi \\ 1 \end{bmatrix} = \pi + \pi = 0 \pmod{2}.$$

Соотношение $\Pi P^t = 0$ (или $P \Pi^t = 0$) справедливо и в общем случае, если под Π и P понимать соответственно матрицы произвольных разрезов и циклов, но при условии, что для обеих матриц принят один и тот же порядок следования ребер. Его доказательство основано на том факте,

что в связном графе каждый цикл имеет четкое число (возможно равное нулю) общих ребер с каждым разрезом (рис. 30, б). При умножении каждой строки матрицы Π на каждый столбец матрицы P (представляющий собой строку матрицы P) получим четное число единиц, сумма которых по модулю 2 дает нуль. Таким образом, матрица произвольных разрезов (или циклов) ортогональна транспонированной матрице произвольных циклов (или разрезов).

5.9.9. Пространство суграфов.

Структурные свойства графа удобно интерпретировать в терминах линейного векторного пространства над числовым полем $\{0,1\}$, операции сложения и умножения в котором определены по модулю 2. Пусть L — множество суграфов графа $G = (V, E)$ с естественной нумерацией вершин и ребер, т. е. $V = \{1, 2, \dots, p\}$ и $E = \{1, 2, \dots, q\}$. Любой суграф содержит все вершины и некоторое (возможно, пустое) подмножество ребер графа, т. е. $G' = (V, E')$, где $E' \subset E$. Ему можно поставить в соответствие q -мерный вектор $x = \{x_1, x_2, \dots, x_q\}$, в котором x_i принимает значение 1 или 0 в соответствии с тем, принадлежит или не принадлежит данному суграфу i -е ребро графа, т. е.

$$x_i = \begin{cases} 1, & \text{если } i \in E' \\ 0, & \text{если } i \notin E' \end{cases} \quad (i = 1, 2, \dots, q).$$

Например, суграфу, выделенному на рис. 22, жирными линиями, соответствует вектор $(0, 0, 0, 0, 1, 1, 0, 1, 0)$. Деревья и дополнения, циклы и разрезы, контуры и сечения можно рассматривать как суграфы и использовать для них векторное представление. Так, разрез на рис. 26, а представляется вектором $(0, 1, 1, 0, 0, 0, 1, 0, 1)$, а разрез на рис. 26, г — вектором $(0, 1, 1, 0, 0, 0, 1, 0, 1)$.

Суммой суграфов

$$G' + G'' = (V, E') + (V, E'')$$

будем называть суграф, которому соответствует вектор, равный сумме по модулю 2 векторов этих суграфов. Это значит, что сумма суграфов содержит те ребра, которые содержатся в E' , но не содержатся в E'' и которые содержатся в E'' , но не содержатся в E' . Иначе говоря, множество ребер суммы двух суграфов равно дизъюнктивной сумме ребер, принадлежащих каждому из них, т. е.

$$G' + G'' = (V, E') + (V, E'') = (V, E' \dot{+} E'').$$

Множество суграфов с определенной на нем внутренней операцией суммы образует абелеву группу. При этом нейтральным элементом является пустой суграф (не содержащий ребер), а обратным элементом

к каждому суграфу служит сам этот суграф. Совместно с внешней операцией — умножение суграфа на скаляр из числового поля $(0, 1)$ — абелеву группу можно рассматривать как линейное *пространство суграфов* данного графа $G = (V, E)$. Размерность этого пространства равна числу ребер q графа G и в качестве его базиса можно принять множество однореберных суграфов, которым соответствуют векторы $g_1 = (1, 0, \dots, 0)$, $g_2 = (0, 1, \dots, 0)$, ..., $g_q = (0, 0, \dots, 1)$. Любой суграф представляется вектором q -мерного пространства через его базис, т. е.

$$g = \alpha_1 g_1 + \alpha_2 g_2 + \dots + \alpha_q g_q = \sum_{i=1}^q \alpha_i g_i,$$

где α_i — элемент из числового поля, принимающий значение 0 или 1, Совокупность $\nu = p - 1$ разрезов образует базис ν -мерного *пространства разрезов*, а совокупность $\sigma = q - p + 1$ независимых циклов образует базис σ -мерного *пространства циклов*. Оба эти пространства являются подпространствами пространства суграфов.

5.9.10. Несвязные графы.

При рассмотрении несвязных графов совокупности независимых сечений и контуров можно определять для каждой его компоненты отдельно (рис. 31).

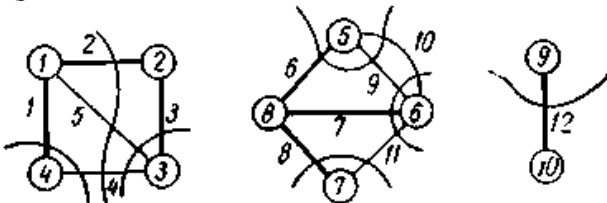


Рис. 31. Совокупность независимых сечений несвязного графа, состоящего из трех компонентов.

Ранг и цикломатическое число графа, содержащего k компонент, соответственно равны $\nu = p - k$ и $\sigma = q - p + k$. Матрица инцидентности A содержит $p - k$ независимых строк, и для перехода к сокращенной матрице A_0 необходимо удалить из каждой совокупности строк, соответствующих компонентам графа, по одной строке. Так, для графа на рис. 31 матрица инцидентности имеет блочную структуру:

$$A =$$

	1	2	3	4	5	6	7	8	9	10	11	12	
1	1	1			1								1
2		1	1										2
3			1	1	1								3
4	1			1									4
5						1			1	1			5
6							1		1	1	1		6
7								1				1	7
8						1	1	1					8
9													1
10													1

Сокращенную матрицу инцидентности можно записать в следующем виде:

$$A_0 =$$

1	1	1		1										1
2		1	1											2
3			1	1		1								3
5						1			1	1				5
6							1		1	1	1			6
7								1				1		7
9														1

Преобразуя матрицу A или A_0 с помощью операций над строками по модулю 2, можно получить матрицы сечений и контуров теми же способами, что и для связного графа.

5.9.11. Топология ориентированных графов.

Элементы матрицы инцидентности для орграфа принимают значения 0, 1 и -1 в зависимости от инцидентности и направления дуги относительно вершины. Например, для орграфа на рис. 32, а имеем:

	1	2	3	4	5	6	7	8	9		
$A =$	1	1			-1					1	
		-1	-1						1	2	
	-1			-1		-1				3	
			1	1						1	4
					1	1	-1				5
							1	-1	-1		6

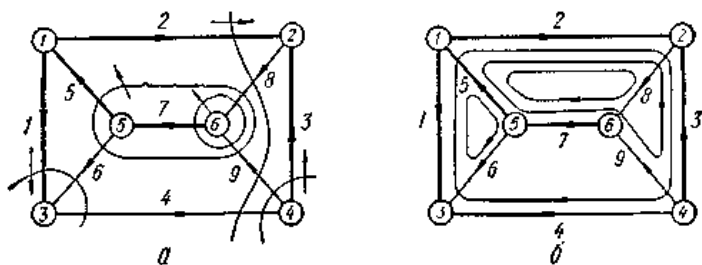


Рис. 32. Сечения (а) и контуры (б) ориентированного графа.

При преобразовании матрицы A операции над ее элементами выполняются как над обычными числами. Можно показать, что минор любого порядка матрицы A , как и ее элементы, может равняться только 0, 1 или -1 . Матрицы, обладающие таким свойством, называют *унимодулярными*.

С помощью операций перестановки строк и столбцов, а также замены строки ее алгебраической суммой с другой строкой можно преобразовать матрицу инцидентности A (или A_0) к такому виду,

чтобы элементы совокупности ее $p - 1$ столбцов и строк (последняя строка матрицы A преобразуется в нулевую и отбрасывается) представляли собой единичную матрицу. Тогда получим матрицу сечений в канонической форме $\Pi = [1, \pi]$, где π — матрица сечений хорд. В нашем примере:

	1	2	3	5	7	4	6	8	9	
$\Pi =$	1					1	1			1
		1				-1		-1	-1	2
			1			1				3
				1			1	-1	-1	5
					1			-1	-1	7

Каждому сечению приписывается направление, которое определяется направлением соответствующей ветви фундаментального дерева. Ненулевые элементы строки матрицы Π указывают на совокупность дуг, инцидентных данному сечению, причем знак плюс означает, что направления дуги и сечения совпадают, а знак минус означает, что эти направления противоположны. На рис. 32, а направления сечения указаны стрелками, но можно обойтись и без них, так как достаточно руководствоваться ориентацией ветвей дерева.

Фундаментальное дерево определяет и совокупность независимых контуров, каждый из которых образуется одной хордой и некоторой частью ветвей дерева. Направление контура обычно принимают совпадающим с направлением определяющей его хорды. Дуга, инцидентная данному контуру, может совпадать с направлением контура или быть противоположной ему. В первом случае соответствующий элемент матрицы контуров равен 1, а во втором случае — -1 . Так, в соответствии с рис. 32,б имеем:

$$P = \begin{array}{cccccccc} & 1 & 2 & 3 & 5 & 7 & 4 & 6 & 8 & 9 \\ \hline -1 & 1 & -1 & & & & 1 & & & \\ \hline -1 & & & & -1 & & & 1 & & \\ \hline & 1 & & & 1 & 1 & & & 1 & \\ \hline & 1 & -1 & & 1 & 1 & & & & 1 \end{array}$$

Между матрицами сечений и контуров орграфа, определяемых некоторым фундаментальным деревом, имеются такие же зависимости, как и для неориентированного графа:

$$P P^t = 0; \quad P \Pi^t = 0.$$

Представив топологические матрицы в канонической форме и выполнив соответствующие операции, имеем:

$$[1 \ \pi] \begin{bmatrix} \rho^t \\ 1 \end{bmatrix} = \rho^t + \pi = 0,$$

откуда получаем

$$\rho = -\pi^t; \quad \pi = -\rho^t.$$

Таким образом, для определения топологических матриц достаточно знать одну из них, а другая определяется полученными соотношениями.

5.10. Эйлеровы циклы и задача китайского почтальона

Определение. Если дан неориентированный s -граф G , то *эйлеров цикл* (эйлерова цепь) — это такой цикл (цепь), который проходит ровно один раз по каждому ребру.

Очевидно, не все графы имеют эйлеровы циклы (см., например, граф на рис. 33), но если эйлеров цикл существует, то это означает, что, следуя вдоль этого цикла, можно нарисовать граф на бумаге не отрывая от нее карандаша.

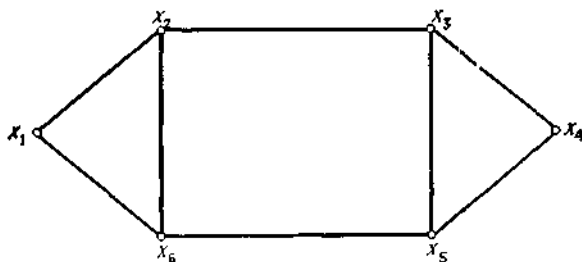


Рис. 33. Граф без эйлерова циклов.

3-граф на рис. 34 имеет следующий эйлеров цикл (начиная с вершины x_1):

$$a_1 a_2 a_3 a_4 a_{15} a_{14} a_{13} a_{12} a_{11} a_{16} a_{17} a_{10} a_9 a_8 a_5 a_7 a_6.$$

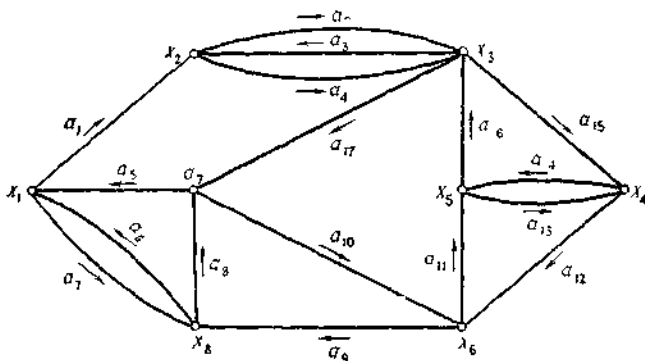


Рис 34.

Направление движения по каждому ребру показано на рисунке стрелками.

Эйлер первым в своей знаменитой задаче о Кенигсбергских мостах рассмотрел вопрос о существовании таких циклов в графах.

Кенигсберг (теперь Калининград) расположен на обоих берегах реки Преголя и на двух островах этой реки. Берега реки и два острова соединены семью мостами как показано на карте на рис. 35 (а).

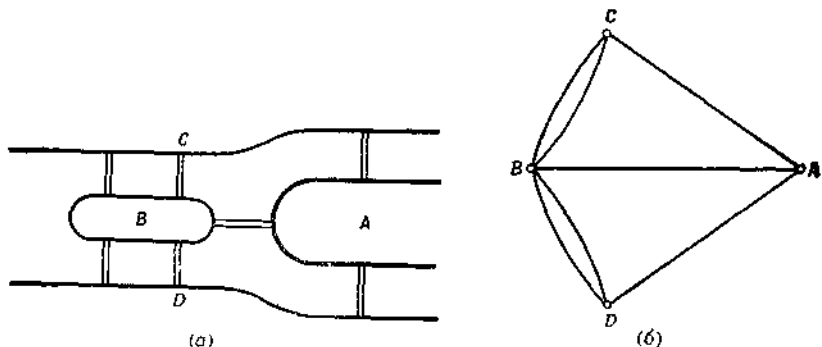


Рис. 35. (а) Карта Кенигсберга. (б) Эквивалентный граф.

Вопрос — поставленный в 1736 году — состоит в том, можно ли, начав с некоторой точки, совершить прогулку и вернуться в исходную точку, пройдя по каждому мосту ровно один раз. Если каждый берег реки и острова считать вершиной графа, а каждый мост — ребром, то карту рис. 35 (а) можно представить в виде 2-графа, изображенного на рис. 35 (б), и ответ на поставленный вопрос зависит теперь от существования эйлера цикла в этом графе. Эйлер установил, что указанный граф не содержит эйлера цикла, и этот результат ознаменовал рождение теории графов. Основная теорема о существовании эйлеровых циклов формулируется так.

Теорема (а). *Связный неориентированный s -граф G содержит эйлеров цикл (эйлерову цепь) тогда и только тогда, когда число вершин нечетной степени равно 0 (0 или 2).*

Доказательство. Мы докажем эту теорему для случая цикла; случай цепи рассматривается аналогично.

Необходимость. Любой эйлеров цикл должен прийти в вершину по одному ребру и покинуть ее по другому, так как любое ребро должно использоваться ровно один раз. Поэтому, если G содержит эйлеров цикл, то степени всех вершин должны быть четными.

Достаточность. Пусть G — связный неориентированный s -граф, все вершины которого имеют четную степень. Начнем путь из некоторой произвольной вершины x_0 и пойдем по некоторому ранее не использованному ребру к следующей вершине, и так до тех пор, пока не вернемся в вершину x_0 и не замкнем цикл. Если все ребра окажутся использованными, то нужный эйлеров цикл построен. Если же некоторые ребра не использованы, то пусть Φ — только что построенный цикл. Так как граф G связан, то цикл Φ должен проходить через некоторую вершину, скажем x_i , являющуюся

конечной вершиной какого-либо до сих пор неиспользованного ребра. Если удалить все ребра, принадлежащие Φ , то в оставшемся графе все вершины по-прежнему будут иметь четную степень, так как в цикле Φ должно быть четное число ребер (0 является четным числом), инцидентных каждой вершине.

Начиная теперь с x_i , получаем цикл Φ' , начинающийся и оканчивающийся в x_i . Если все оставшиеся ранее ребра использованы для цикла Φ' , то процесс окончен. Нужный эйлеров цикл будет образован частью цикла Φ от вершины x_0 до x_i , затем циклом Φ' и, наконец, частью цикла Φ от вершины x_i до x_0 . Если же все еще остались неиспользованные ребра, то объединение построенных выше циклов Φ и Φ' дает новый цикл Φ . Мы снова можем найти вершину x_i , принадлежащую циклу и являющуюся концевой вершиной некоторого неиспользованного ребра. Затем мы можем приступить к построению нового цикла Φ' , начинающегося в x_i , и так продолжать до тех пор, пока не будут использованы все ребра и не будет получен таким образом эйлеров цикл Φ . Это доказывает теорему.

Очевидно, что если граф G несвязен (за исключением изолированных вершин), то эйлеров цикл не существует, так как нет никакой цепи, ведущей из одной его компоненты в другую. Очевидно также, что в случае эйлеровой цепи две ее конечные вершины p и q являются вершинами нечетной степени.

Если G — ориентированный s -граф, то справедлива теорема, полностью аналогичная предыдущей, а именно

Теорема (б). *Связный ориентированный s -граф G содержит эйлеров цикл (эйлерову цепь) тогда и только тогда, когда все полустепени захода $d_i(x_i)$ и полустепени исхода $d_o(x_i)$ вершин удовлетворяют условиям:*

$$\text{для случая цикла } d_i(x_i) = d_o(x_i), \forall x_i \in X,$$

$$\text{для случая цепи } d_i(x_i) = d_o(x_i) \forall x_i \neq p \text{ или } q,$$

$$d_i(q) = d_o(q) + 1$$

$$\text{и } d_i(p) = d_o(p) - 1,$$

где p — начальная, а q — конечная вершины эйлеровой цепи.

Флэри дал алгоритм построения эйлерова цикла в неориентированном графе (если такой цикл существует). Этот алгоритм легко может быть распространен на ориентированные графы и состоит в следующем. *Алгоритм нахождения эйлерова цикла.* Начинать с некоторой вершины p и каждый раз вычеркивать пройденное ребро. Не проходить по ребру,

если удаление этого ребра приводит к разбиению графа на две связные компоненты (не считая изолированных вершин).

5.10.1. Некоторые родственные задачи

Сразу же вспоминается ряд вопросов, связанных с тем, имеется ли в неориентированном графе эйлеров цикл. Например:

(i) Каково наименьшее число цепей или циклов необходимо для того, чтобы каждое ребро графа G содержалось точно в одной цепи или в одном цикле? Очевидно, что если G имеет эйлерову цепь или эйлеров цикл, то ответом будет число один. (Этот вопрос относится к нахождению наименьшего числа реберно непересекающихся подграфов (в случае цепей или циклов), необходимых для «покрытия» ребер графа G . Задачи этого типа обсуждались ранее как варианты задачи о покрытии множества.)

(ii) Ребрам графа G приписаны положительные веса. Требуется найти цикл, проходящий через каждое ребро графа G по крайней мере один раз и такой, что для него общий вес (а именно сумма величин $n_j c(a_j)$, где число n_j показывает, сколько раз проходило ребро a_j , а $c(a_j)$ — вес ребра) минимален. Очевидно, что если G содержит эйлеров цикл, то любой такой цикл будет оптимальным, так как каждое ребро проходится только один раз и вес этого цикла равен тогда

$$\sum_{j=1}^m c(a_j).$$

Сформулированная выше задача (ii) называется задачей *китайского почтальона*, и ее решение имеет много потенциальных приложений, как например:

(А) **Сбор мусора.** Рассмотрим проблему сбора домашнего мусора. Допустим, что определенный район города обслуживается единственной машиной. Ребра графа G представляют дороги, а вершины — пересечения дорог. Величина $c(a_j)$ — вес ребра — будет соответствовать длине дороги. Тогда проблема сбора мусора в данном районе сводится к нахождению цикла в графе G , проходящего по каждому ребру G по крайней мере один раз. Требуется найти цикл с наименьшим километражем. В действительности емкость машины и продолжительность рабочего дня накладывают ограничения на число улиц, которые может обслужить одна машина за день. То, что действительно может потребоваться, — это не нахождение отдельного цикла, а некоторого числа Q циклов, обслуживаемых по одному в день, скажем, за период в Q дней; при этом циклы могут быть выбраны так, что не нарушаются вышеупомянутые ограничения.

(Б) **Доставка молока или почты.** Еще две задачи, когда требуется определить маршрут, проходящий хотя бы один раз по каждой из улиц, возник при доставке молока или почты. Здесь задача состоит в нахождении маршрута, минимизирующего общий километраж (или время, стоимость и т. д.).

(В) **Проверка электрических, телефонных или железнодорожных линий.** Проблема инспектирования распределенных систем (лишь некоторые из которых названы выше) связана с непременным требованием проверки всех «компонент». Поэтому она также является проблемой типа (ii) или близка к ней.

Другие приложения могут быть связаны с разбрасыванием смеси песка и соли на главных дорогах зимой для предотвращения обледенения, с наилучшим методом работы автоматических вентиляционных устройств, с уборкой помещений и коридоров в больших учреждениях и даже с наилучшим маршрутом осмотра музея!

5.10.2. Алгоритм для задачи китайского почтальона

В случае когда веса всех ребер равны единице, задачу китайского почтальона рассмотрели Беллман и Кук с использованием динамического программирования. Более общую задачу, когда веса ребер произвольны, сформулировали и решили как задачу о паросочетании (с частной задачей о кратчайшей цепи) Эдмондс, Эдмондс и Джонсон, Басакер и Саати и Кристофидес.

Рассмотрим неориентированный граф $G = (X, A)$. Среди вершин из X некоторые вершины (скажем из множества X^+) будут иметь четные степени, а остальные (из множества $X \equiv X - X^+$) — нечетные степени. Далее сумма степеней d_i всех вершин $x_i \in X$ равна удвоенному числу ребер в A (так как каждое ребро добавляет по единице к степеням двух его концевых вершин) и поэтому равна четному числу $2m$. Следовательно,

$$\sum_{x_i \in X} d_i = \sum_{x_i \in X^+} d_i + \sum_{x_i \in X^-} d_i = 2m,$$

и так как сумма

$$\sum_{x_i \in X^+} d_i$$

четна, то сумма

$$\sum_{x_i \in X^-} d_i$$

также четна. Но все d_i в последней сумме нечетны, значит число $|X|$ вершин нечетной степени четно.

Пусть M — множество таких цепей (скажем μ_{ij}) в G между концевыми вершинами x_i и x_j ($x_i, x_j \in X$), что никакие две цепи не имеют одинаковых конечных вершин, т. е. цепи соединяют различные пары вершин из X и покрывают все вершины множества X . (В дальнейшем такое множество цепей будем называть *цепным паросо четанием* (на

множестве X или для множества X). Число цепей μ_{ij} в M равно $\frac{1}{2} |X|$,

а как было показано выше, $|X|$ всегда четно, следовательно, это число всегда целое, если, конечно, оно определено. Предположим теперь, что все ребра, образующие цепь μ_{ij} , добавлены к G в качестве *искусственной* параллельной цепи, причем эти ребра остаются в графе G . Это означает, прежде всего, что все ребра из G , образующие цепь μ_{ij} , теперь удвоены. Так поступаем с каждой цепью $\mu_{ij} \in M$, и полученный s -граф обозначим через $G^*(M)$. Так как некоторые ребра из G могут входить более чем в одну цепь μ_{ij} , то некоторые ребра из $G^*(M)$ могут быть (после того как добавлены все «новые» цепи μ_{ij}) утроены, учетверены и т. д.

Теперь мы можем доказать следующую теорему.

Теорема. *Для любого цикла, проходящего по G , можно выбрать множество M , для которого граф $G^*(M)$ имеет эйлеров цикл, соответствующий первоначально взятому циклу в графе G . Это соответствие таково, что если цикл проходит по ребру (x_i, x_j) из G l раз, то в $G^*(M)$ существует l ребер (одно реальное и $l - 1$ искусственных) между x_i и x_j , каждое из которых проходится ровно один раз эйлеровым циклом из $G^*(M)$. Справедливо и обратное утверждение.*

Доказательство. Если цикл проходит по графу G , то по крайней мере одно ребро, инцидентное каждой вершине x_i нечетной степени, должно проходиться дважды. (Ребро, проходящее дважды, можно рассматривать как два параллельных ребра — одно реальное и одно искусственное — и каждое из них проходится один раз.) Пусть это ребро (x_i, x_k) . В случае нечетности степени d_k вершины x_k графа G добавление искусственного ребра прежде всего сделает d_k четным, и значит только ребро (x_i, x_k) нужно будет проходить дважды, если ограничиться рассмотрением лишь вершин x_i и x_k . В случае когда d_k четно, добавление искусственного ребра сделает d_k нечетным, а второе ребро выходящее из x_k , должно быть пройдено дважды (т. е. добавляется еще одно искусственное ребро). Такая ситуация

сохраняется до тех пор, пока не встретится вершина нечетной степени, о чем говорилось выше. Следовательно, чтобы удовлетворить условию возвращения в вершину x_i , нужно дважды пройти всю цепь из x_i в некоторую другую вершину нечетной степени x_r , принадлежащую множеству X . Это автоматически приводит к выполнению условия прохождения вершины x_r . Аналогично обстоит дело для всех других вершин x_i из X . Это значит, что все множество M цепей из G , определенное выше, должно проходиться дважды, и так как отсюда вытекает, что каждое ребро из $G(M)$ должно проходиться один раз, то теорема доказана.

Алгоритм решения задачи китайского почтальона немедленно следует из доказанной теоремы, так как все, что для этого необходимо, состоит в нахождении множества цепей M^* (цепного паросочетания для множества вершин нечетной степени), дающего наименьший дополнительный вес. Цикл наименьшего веса, проходящий по G , будет иметь вес, равный сумме весов всех ребер из G плюс сумма весов ребер в цепях из M^* . Это то же самое, что и сумма весов всех ребер — реальных и искусственных — графа $G(M^*)$. Дадим теперь описание алгоритма.

Описание алгоритма. Шаг 1. Пусть $[c_{ij}]$ — матрица весов ребер графа G . Используя алгоритм кратчайшей цепи, образуем $|X| \times |X|$ -матрицу $\mathbf{D} = [d_{ij}]$, где d_{ij} — вес цепи наименьшего веса, идущей из некоторой вершины $x_i \in X$ в другую вершину $x_j \in X$.

Шаг 2. Найдем то цепное паросочетание M^* для множества X , которое дает наименьший вес (в соответствии с матрицей весов \mathbf{D}). Это можно сделать эффективно с помощью алгоритма минимального паросочетания.

Шаг 3. Если вершина x_a сочетается с другой вершиной x_b , то определим цепь μ_{ab} наименьшего веса (из x_a в x_b), соответствующую весу d_{ab} , делая шаг 1. Добавим искусственные ребра в G , соответствующие ребрам из μ_{ab} , и проделаем это для всех других цепей из множества M^* , в результате чего получится s -граф $G(M^*)$.

Шаг 4. Сумма весов всех ребер графа $G(M^*)$, найденная с использованием матрицы $[c_{ij}]$ (вес искусственного ребра берется равным весу параллельного ему реального ребра), равна минимальному весу цикла, проходящего по G . При этом число прохождений цикла по ребру $\{x_i, x_j\}$ равно общему числу параллельных ребер между x_i и x_j в $G(M^*)$.

Следует заметить, что поскольку на шаге 2 мы используем минимальное паросочетание, никакие две кратчайшие цепи μ_{ij} и μ_{pq} при таком паросочетании (скажем, идущие из x_i в x_j и из x_p в x_q) не могут иметь общего ребра. Если бы они имели общее ребро (x_a, x_b) , как

показано на рис. 36, то сочетание вершин x_i и x_q (использующее подцепи от x_i к x_b и от x_b к x_q) и сочетание пары вершин x_p и x_j (использующее подцепи от x_p к x_a и от x_a к x_j) давало бы общее паросочетание веса $2c_{ab}$, меньшего чем вес первоначального паросочетания, что противоречит предположению о минимальности исходного паросочетания.

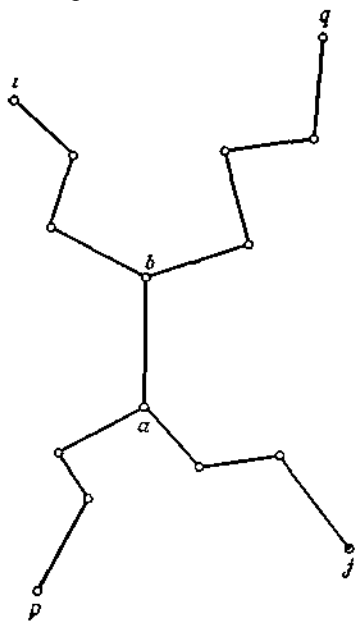


Рис. 36. Цепи μ_{ij} и μ_{pq} , имеющие общее ребро (a, b) .

Следовательно, граф $G(M^*)$ не должен содержать более двух параллельных ребер между любыми двумя вершинами, т. е. оптимальный цикл не проходит ни по какому ребру графа G более чем два раза.

Пример. Рассмотрим граф G , изображенный на рис. 37, имеющий 12 вершин и 22 ребра, веса которых указаны у каждого ребра.

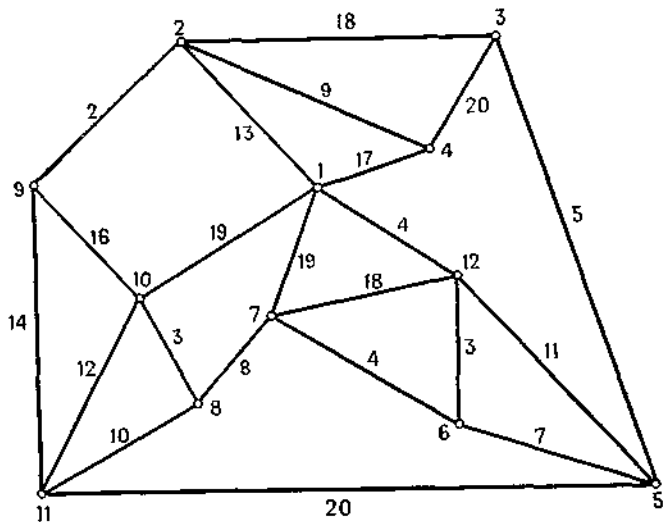


Рис. 37. Граф из примера.

Задача состоит в нахождении цикла, проходящего по каждому ребру G по крайней мере один раз и имеющего минимальный вес. Множество вершин нечетной степени этого графа есть $\{1, 3, 4, 6, 8, 9\}$. Используя алгоритм кратчайшей цепи Дейкстры, найдем матрицу \mathbf{D} из шага 1:

	1	3	4	6	8	9
$\mathbf{D} =$	1	3	4	6	8	9
	—	19	17	7	19	15
	19	—	20	12	24	20
	17	20	—	24	30	11
	7	12	24	—	12	22
	19	24	30	12	—	19
	15	20	11	22	19	—

На шаге 2 алгоритм минимального паросочетания связывает следующие вершины (два других паросочетания также являются минимальными):

- 1 с 6, цепь 1—12—6, вес 7,
- 3 с 8, цепь 3—5—6—7—8, вес 24,
- 9 с 4, цепь 9—2—4, вес 11.

Граф $G(M^*)$, получающийся после выполнения шага 3, изображен на рис. 38; пунктиром показаны искусственные ребра.

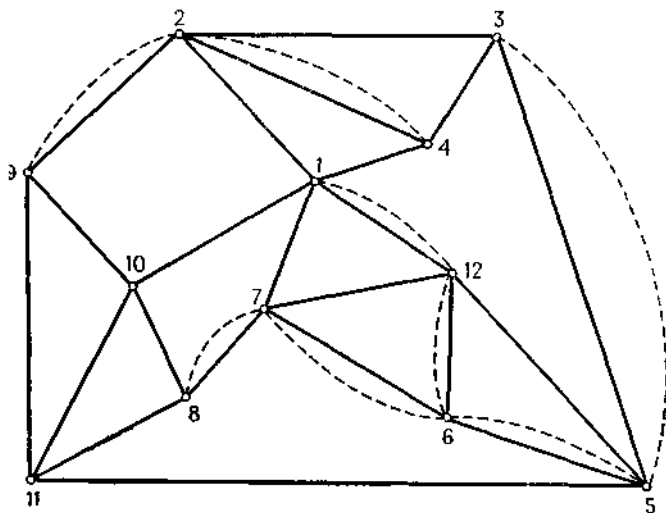


Рис. 38. Граф $G(M^*)$ из примера.

————— реальные ребра; - - - искусственные ребра.

Из этого рисунка видно, что оптимальный цикл в графе G проходит через ребра (9, 2), (2, 4), (3, 5), (5, 6), (1, 12), (12, 6), (7, 6) и (8, 7) дважды, а через все остальные ребра — один раз. Вес этого цикла равен 294.

Как только найден граф $G(M^*)$, могут быть сразу же найдены по алгоритму Флэри, упомянутому ранее, эйлеров цикл этого графа и соответствующий оптимальный цикл, проходящий по первоначальному графу G . Для графа, изображенного на рис. 38, например, один из возможных эйлеровых циклов, полученных по вышеприведенному правилу, дается последовательностью вершин (начиная с вершины 1):

1, 4, 3, 5, 3, 2, 4, 2, 1, 12, 5, 6, 7, 8, 10, 1, 7, 6,
12, 6, 5, 11, 9, 2, 9, 10, 11, 8, 7, 12, 1.

Этот цикл является также соответствующим оптимальным циклом, проходящим по первоначальному графу из рис. 37. Возможны также другие оптимальные циклы, проходящие ребра графа в ином порядке. Однако эти циклы будут проходить дважды по тому же множеству из 8 ребер. Таким образом, как только определены те ребра графа G , по которым оптимальный цикл проходит дважды, можно построить несколько таких циклов в графе G .

5.10.3. Связь между эйлеровыми и гамильтоновыми циклами

Гамильтонов цикл в графе G был определен ранее как простой цикл, проходящий (один, и только один раз) через каждую вершину графа G . Не удивительно, что двойственность между эйлеровыми и гамильтоновыми циклами (замена вершины на ребро и наоборот) приводит к тесной связи между этими двумя понятиями в применении к неориентированному графу G и соответствующему ему *реберному графу*, определяемому ниже.

Определение. Реберный граф G_l графа G имеет столько же вершин, сколько ребер у графа G . (Реберный граф графа G часто обозначается через $L(G)$). Ребро между двумя вершинами графа G_l существует тогда и только тогда, когда ребра графа G , соответствующие этим двум вершинам, смежны (т. е. инцидентны одной и той же вершине графа G).

Например, для графа, показанного на рис. 39(а), граф G_l изображен на рис. 39(б).

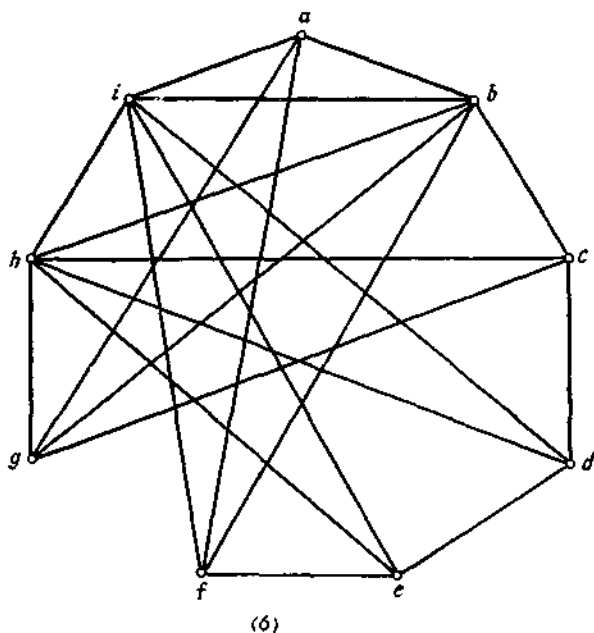
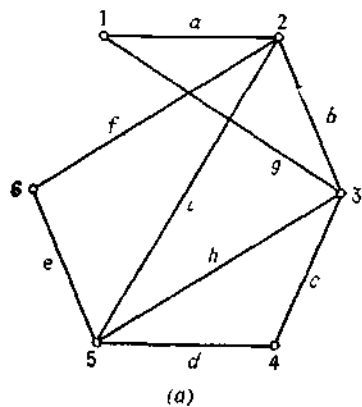


Рис. 39. (а) - граф G . (б) - реберный граф G_l .

Можно показать, что верны два следующие утверждения о взаимоотношении между эйлеровыми и гамильтоновыми циклами.

(1) Если G имеет эйлеров цикл, то G_l имеет как эйлеров, так и гамильтонов циклы.

(2) Если G имеет гамильтонов цикл, то G_1 также имеет гамильтонов цикл.

Обращение этих утверждений, как нетрудно продемонстрировать, неверно. Для графа, изображенного на рис. 39(а), степени всех его вершин четны и поэтому существует эйлеров цикл. Таким образом, граф G_1 , изображенный на рис. 39(б), также имеет эйлеров цикл (поскольку степени всех его вершин опять четны) и, кроме того, имеет гамильтонов цикл, задаваемый последовательностью вершин $a, g, c, d, e, f, b, h, i, a$. Если теперь из графа G удалить ребро b , то новый граф G' не имеет эйлерова цикла, но все еще имеет гамильтонов цикл 1, 2, 6, 5, 4, 3, 1. Реберный граф G'_1 графа G' является тогда графом из рис. 39 (б) с удаленной вершиной b (и без ребер, инцидентных вершине b). Этот граф все еще имеет гамильтонов цикл $a, f, e, d, c, g, h, i, a$.

Задача выяснения существования у данного неориентированного графа G гамильтонова цикла, а также нахождения такого цикла, если он существует, является важной задачей теории графов как с практической, так и с теоретической точек зрения. Если ребрам графа G приписаны некоторые веса $[c_{ij}]$ и граф G содержит несколько гамильтоновых циклов, то большой интерес представляет также задача нахождения такого цикла с минимальным общим весом. Ввиду важности этих вопросов гамильтоновы циклы рассматриваются отдельно.

6. Числа, функции, ядра графов

6.1. Основные числа теории графов

6.1.1. Цикломатическое число

Понятие которое мы собираемся здесь ввести не зависит от „ориентации" Для большей общности введем в рассмотрение не просто графы, а *мультиграфы*; по определению *мультиграф* (X, U) это пара, образованная множеством X вершин и множеством U ребер соединяющих некоторые пары вершин, в противоположность обычным графам у мультиграфа одна и та же пара вершин может соединяться более чем одним ребром

Во многих задачах удобно вместо обычных графов рассматривать мультиграфы.

Пример (химия). Молекула представляется мультиграфом вершины которого обозначены символами таблицы Менделеева (Пойа применил теорию графов к органической химии для подсчета числа изомеров химических соединений). Говорят также что этилен является 2-графом ацетилен — 3-графом) и т. д. (см. рис. 1)

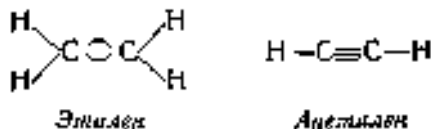


Рис. 1

Рассмотрим мультиграф G с n вершинами, m ребрами, p компонентами связности. Положим

$$\rho(G) = n - p$$

$$\nu(G) = m - \rho(G) = m - n + p$$

$\nu(G)$ называется *цикломатическим числом* мультиграфа G . Его свойства играют важную роль.

Цикломатическим числом графа называется цикломатическое число того 2-графа, который получен, если каждую дугу заменить ребром.

Теорема 1. Пусть G' — мультиграф полученный из мультиграфа G добавлением нового ребра между вершинами a и b ; если a и b совпадают или могут быть соединены цепью в G то

$$\rho(G') = \rho(G), \quad \nu(G') = \nu(G) + 1$$

в противном случае

$$\rho(G') = \rho(G) + 1, \quad \nu(G') = \nu(G)$$

(Непосредственно)

Следствие: $\rho(G) \geq 0, \quad \nu(G) \geq 0$

В самом деле, для графа, образованного всеми вершинами G , но без ребер имеем $\rho = 0, \nu = 0$. Каждое добавление ребра либо увеличивает ρ , не меняя ν , либо увеличивает ν , не меняя ρ . Таким образом, в процессе построения графа G числа ρ и ν могут только возрастать.

Для дальнейшего удобно следующим образом отождествлять циклы мультиграфа с векторами: придадим каждому ребру мультиграфа G произвольную ориентацию, если цикл μ проходит через ребро u_k в направлении его ориентации r_k раз и в противоположном направлении s_k раз, то полагаем $c^k = r_k - s_k$. Вектор

$$(c^1, c^2, \dots, c^k, \dots, c^m)$$

m - мерного пространства \mathbf{R}^m будем называть *вектором-циклом*, соответствующим циклу μ и обозначать через μ , (или опять через μ , если это не может привести к недоразумению)

Циклы μ , μ' , μ'' , называются *независимыми* если соответствующие им векторы линейно независимы. Отметим что это свойство не зависит от выбора ориентации ребер.

Имеет место

Теорема 2. *Цикломатическое число $v(G)$ мультиграфа G равно наибольшему количеству независимых циклов.*

Действительно, возьмем граф без ребер, образованный всеми вершинами G и добавляя к нему ребро за ребром построим данный мультиграф G . В силу теоремы 1 цикломатическое число увеличивается на единицу, когда добавление ребра приводит к образованию новых циклов и не меняется в противоположном случае.

Допустим, что перед добавлением ребра u_k уже имелась база, состоящая из независимых циклов μ_1, μ_2, \dots , и что добавление u_k повлекло за собой возникновение циклов ν_1, ν_2, \dots . Среди новых циклов наверняка имеются простые. Пусть, например ν_1 — простой, $v^k_{\nu_1} = 1$ Очевидно ν_1 не может линейно выражаться через μ_i (ибо $\mu^k_{\nu_1} = \mu^k_{\nu_2} = \dots = 0$).

С другой стороны ν_2 (и аналогично ν_3, \dots) можно линейно выразить через $\nu_1, \mu_1, \mu_2, \dots$. В самом деле вектор $\nu_2 - \mu^k_{\nu_2} \nu_1$ соответствует некоторому циклу, не содержащему u_k (этот цикл получается из ν_2 заменой ребра u_k оставшейся частью ν_1 с измененным направлением обхода) и значит линейно выражается через μ_1, μ_2, \dots . Таким образом каждый шаг, увеличивающий на единицу цикломатическое число, в то же время увеличивает на единицу наибольшее количество независимых циклов. Теорема доказана

Следствие 1. *Граф G не имеет циклов тогда и только тогда, когда $v(G) = 0$*

Следствие 2. *Граф G имеет один единственный цикл тогда и только тогда, когда $v(G) = 1$*

Теорема 3. *В сильно связном графе цикломатическое число равно наибольшему количеству независимых контуров.*

В самом деле, рассмотрим 2-граф, получающийся заменой дуг данного графа G ребрами, и элементарный цикл μ . Вершины, встречающиеся в цикле μ , можно распределить по следующим множествам S точек, обладающих тем свойством, что одна из дуг μ исходит из точки, а другая заходит в нее. Множества S *зависимы*. Если $\alpha_1 \neq 0$, то можно также записать:

$$c_1 = \frac{\alpha_2}{\alpha_1} c_2 + \dots + \frac{\alpha_k}{\alpha_1} c_k,$$

в этом случае говорят, что c_1 *линейно выражается* через c_2, c_3, \dots, c_k .

База векторного подпространства E есть такое множество векторов $\{e_1, e_2, \dots, e_k\}$ из E , что каждый вектор подпространства E линейно выражается через векторы e_i . Наименьшее из чисел k называется *размерностью* подпространства E .

В $E = \mathbf{R}^m$ одна из баз образуется векторами

$$e_1 = (1, 0, 0, \dots, 0), \quad e_2 = (0, 1, 0, \dots, 0), \dots, \quad e_m = (0, 0, 0, \dots, 1),$$

точек, из которых исходит по две дуги μ , множества S'' точек, в которые заходят по две дуги μ (см рис. 2)

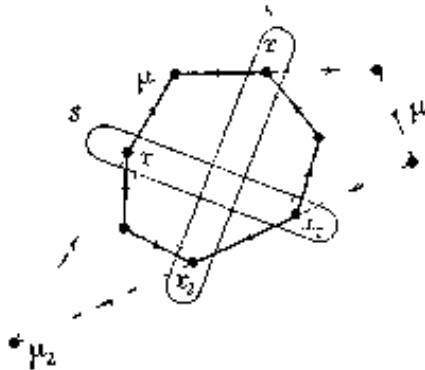


Рис. 2

Так как количество конечных точек равно количеству начальных, то $|S'| = |S''|$. Итак, пусть x'_1, x'_2, \dots, x'_k — элементы S' , а $x''_1, x''_2, \dots, x''_k$ — элементы S'' .

В цикле μ элементы S' чередуются с элементами S'' , и мы предположим нумерацию вершин такой, что первая вершина после x'_1 , не принадлежащая S , есть x_1 . Наконец, если μ_0 — путь, в котором вершина x встречается раньше вершины y , то обозначим через $\mu_0[x, y]$ частичный путь из x в y . Поскольку граф сильно связан, существует контур μ_0 , проходящий через x'_{i+1} и x'_i и содержащий дуги μ на пути от x'_{i+1} к x'_i . Цикл μ является линейной комбинацией контуров, ибо можно написать

$$\begin{aligned} \mu &= \alpha[x_1, x'_1] - \beta[x_2, x'_1] + \gamma[x_0, x'_2] + \dots \\ &= \alpha[x_1, x] + \beta_1[x, x_0] + \beta_2[x_2, x_2] + \gamma_2[x_2, x_2] + \dots \\ &\quad - (\beta_1 + \beta_2 + \dots) \end{aligned}$$

Значит, каждый элементарный цикл есть линейная комбинация контуров, и то же справедливо для произвольного цикла (поскольку он является линейной комбинацией элементарных).

В \mathbf{R}^m контуры образуют базу векторного подпространства, порожденного циклами и, в силу теоремы 2, эта база имеет размерность $\nu(G)$; поэтому наибольшее число независимых контуров равно $\nu(G)$

6.1.2. Хроматическое число

Пусть дано натуральное число p , говорят, что граф G является p -хроматическим, если его вершины можно раскрасить p различными цветами таким образом, чтобы никакие две смежные вершины не были окрашены одинаково. Наименьшее число p , при котором граф G является p -хроматическим называется *хроматическим числом* этого графа и обозначается символом $\chi(G)$

ПРИМЕР. *Географическая карта*

Нарисуем на плоскости карту, содержащую множество X стран, и положим $(x, y) \in U$ если страны x и y граничат между собой.



Рис. 3. 4-хроматическая карта (выбор цветов 0, 1, 2, 3 соответствует значениям некоторой функции Гранди для симметрического плоского графа, изображенного жирными линиями, вершины цвета „0” образуют наибольшее множество, которое можно закрасить в один цвет)

Граф (X, U) — симметрический и обладает к тому же весьма примечательным свойством: его можно начертить на плоскости так, чтобы никакие два ребра не пересекались (в точках, отличных от граничных) Такие графы называются *плоскими*. Известно, что хроматическое число плоского графа никогда не превышает 5, таким образом, пяти красок достаточно для раскрашивания карты (плоской), при котором никакие две соседние страны не окрашиваются в один и тот же цвет

Хроматическим классом графа называется натуральное число q , обладающее следующими свойствами:

1° каждое ребро графа можно окрасить в какой-нибудь из q цветов таким образом, чтобы никакие два смежные ребра не были окрашены одинаково

2° это невозможно сделать с помощью только $q - 1$ цветов.

Хроматический класс графа (X, U) совпадает с хроматическим числом графа $(U, \bar{\Gamma})$, определяемого следующим образом: вершинами его служат ребра исходного графа и $u' \in \bar{\Gamma} u$, когда ребра u и u' исходного графа смежны (см. рис. 4)

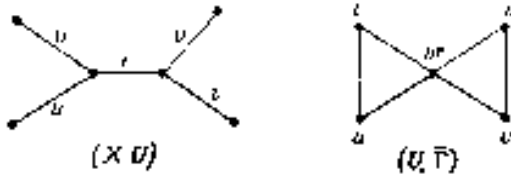


Рис. 4

Таким образом, задача определения хроматического класса сводится к изучению хроматического числа.

Приводим несколько основных результатов относящихся к хроматическим числам.

Теорема Кенига. *Граф является бихроматическим (т. е. имеет хроматическое число ≤ 2) в том и только в той случае, если он не содержит циклов нечетной длины.*

(1) Рассмотрим граф (X, U) без нечетных циклов и покажем что он — бихроматический. Граф можно предполагать связным (в противном случае мы рассмотрели бы все его компоненты связности отдельно). Будем последовательно раскрашивать вершины по следующему правилу:

1° произвольную вершину a окрашиваем в синий цвет

2° если вершина x уже окажется синей, то все смежные с ней вершины окрашиваем в красный цвет, если вершина y — красная, то все смежные с ней окрашиваем в синий цвет.

Так как граф связан, каждая его вершина рано или поздно окажется окрашенной, причем никакая вершина x не будет одновременно синей и красной, ибо иначе x и a находились бы на одном цикле нечетной длины. Следовательно граф — бихроматический.

(2) Если граф — бихроматический, то он, очевидно, не содержит циклов нечетной длины, ибо вершины такого цикла

невозможно окрасить двумя цветами в соответствии с указанным требованием.

Замечание. Свойство

(1) *Граф G не имеет циклов нечетной длины*

равносильно свойству

(2) *граф G не имеет элементарных циклов нечетной длины.*

Непосредственно ясно, что (1) \Rightarrow (2), для доказательства того, что

(2) \Rightarrow (1), допустим, что существует цикл $\mu = [x_0, x_1, \dots, x_p] = x_0$

нечетной длины p . Каждый раз, когда имеются такие две вершины x_j и x_k , что $j < k < p$ и $x_j = x_k$, цикл μ можно разбить на два частичных цикла $\mu[x_j, x_k]$ и $\mu[x_0, x_j] + \mu[x_k, x_0]$, причем ровно один из этих двух циклов имеет нечетную длину.

Ясно, что если продолжить таким же образом разбивать цикл μ , пока это возможно, то всякий раз будет оставаться в точности один цикл нечетной длины. Дойдя в конце концов до элементарных циклов, мы получим противоречие с (2)

Эти результаты позволяют легко распознавать бихроматические графы, что же касается других графов, то для них графические методы определения хроматического числа неизвестны. Отметим, однако, что во многих случаях благодаря следующей теореме действенным орудием оказывается понятие функции Гранди

Теорема. Пусть G — симметрический граф. Чтобы граф G был p -хроматическим необходимо и достаточно, чтобы он допускал функцию Гранди $g(x)$, для которой

$$\max_{x \in X} g(x) \leq p - 1,$$

1° Если такая функция $g(x)$ существует, то граф G является p -хроматическим: в самом деле, достаточно числам $0, 1, \dots, p-1$ поставить в соответствие различные цвета и окрасить каждую вершину x в тот цвет, который отвечает числу $g(x)$

2° Предположим, что граф p -хроматический и докажем, что на G существует функция Гранди, значения которой не превышают $p-1$

Пусть S_1, S_2, \dots, S_{p-1} — множества вершин с одинаковыми цветами. Присоединим к S_0 все вершины из S_1 , не смежные ни с одной из вершин S_0 . Далее присоединим к S_0 все вершины из S_2 не смежные ни с одной из вершин S_0 и вершин S_1 , ранее присоединенных к S_0 . Затем последовательно поступим подобным же образом с множествами S_3, \dots, S_{p-1} . В результате получим множество $\bar{S}_0 \supset S_0$. Пополняя подобным же образом множество $S_1 \setminus \bar{S}_0$ не смежными с ним

вершинами из $S_2 \setminus \bar{S}_0$, а затем из $S_3 \setminus \bar{S}_0$ и т. д., получим множество \bar{S}_1 . Аналогично определяем множество \bar{S}_2 , затем \bar{S}_3 и т.д.

Функция $g(x)$ равная k при $x \in \bar{S}_k$, есть функция Гранди для графа G , что и требовалось

Указания для нахождения хроматического числа. Рассмотрим граф G с n вершинами и m ребрами. Для нахождения его хроматического числа можно воспользоваться эмпирическим способом. Можно использовать и аналитический метод, который систематически дает решение, но требует, вообще говоря, применения ЭВМ. Существует еще рекуррентный способ.

Эмпирический способ состоит в том, чтобы, отправляясь от произвольной раскраски цветами $1, 2, \dots, p$, стараться постепенно исключать какой-либо из этих цветов (который объявляется «критическим цветом»). Для этого рассмотрим вершину x , имеющую этот критический цвет и компоненты связности $C^{jk}_1, C^{jk}_2, \dots$ подграфа, порожденного вершинами двух некритических цветов j и k . Если множества $C^{jk}_1 \cap Gx, C^{jk}_2 \cap Gx$ и т. д., не являются двухцветными, то имеется возможность сразу изменить цвет вершины x . Берем отдельно каждую компоненту C^{jk} , для которой все вершины $C^{jk} \cap Gx$ имеют цвет j , затем меняем ролями в этой компоненте цвета j и k (не меняя цветов других вершин) и, наконец, окрашиваем краской j вершину x (теперь уже не смежную ни с какой вершиной цвета j)

Аналитический метод проверки возможности раскрашивания графа G посредством p цветов состоит в следующем. Каждому раскрашиванию p цветами можно поставить в соответствие систему таких чисел ξ_q^i

(где $i = 1, 2, \dots, n, q = 1, 2, \dots, p$), что

$$\xi_q^i = \begin{cases} 1, & \text{если вершина } x_i \text{ имеет цвет } q, \\ 0, & \text{в противном случае.} \end{cases}$$

Положим

$$r_j^i = \begin{cases} 1, & \text{если } x_i \text{ граничная точка ребра } u_j, \\ 0, & \text{в противном случае.} \end{cases}$$

Задача сводится теперь к отысканию таких целых чисел ξ_q^i , что

$$1^\circ \xi_q^i \geq 0, \quad (i = 1, 2, \dots, n, q = 1, 2, \dots, p),$$

$$2^\circ \sum_{q=1}^p \xi_q^i = 1, \quad (i = 1, 2, \dots, n),$$

$$3^\circ \sum_{k=1}^p r_j^k \xi_q^k \leq 1, \quad (j = 1, 2, \dots, m, \quad q = 1, 2, \dots, p.)$$

Мы получили систему *линейных неравенств*, соответствующую как легко видеть, известным методам „линейных программ“; однако в данном случае требуется выяснить, имеет ли эта система решения в *целых числах*.

Рассмотрим теперь два различных графа G и H . Требуется определить хроматическое число их суммы $G + H$ или их произведения $G \times H$. Имеют место следующие результаты

Теорема. Пусть G — $(p+1)$ хроматический граф, H — $(q+1)$ хроматический граф, обозначим через r наибольшую из d сумм $p' + q'$, где $p' \leq p, q' \leq q$, тогда граф $G + H$ является $(r + 1)$ -хроматическим.

Действительно, всегда можно предположить, что графы G и H симметрические (это никак не изменит ребер графа $G + H$). Построим для G функцию Гранди $g(x)$ с наибольшим значением, не превосходящим p , а для H — функцию Гранти $h(x)$ с наибольшим значением, не превосходящим q в соответствии с предыдущей теоремой. В силу извещной теоремы граф $G + H$ допускает функцию Гранли с наибольшим значением не превосходящим r , отсюда и следует справедливость утверждения.

Например, легко проверить, что если G 6-хроматический, а H 7-хроматический граф, то граф $G + H$ является 8-хроматическим потому, что

$$r = 6 + 1 = (1 \quad 1 \quad 1) = 7.$$

Теорема. Если G и H — два различных графа с хроматическими числами p и q , а $r = \min [p, q]$, то граф $G \times H$ является r -хроматическим

Предположим для определенности, что $p \leq q$ и раскрасим граф G с помощью p цветов; в графе $G \times H$ придадим вершине $\xi=(x, y)$ тот же цвет, который имеет x в G . Тогда смежные вершины графа $G \times H$ будут иметь различные цвета (ибо иначе в G имелись бы одинаково окрашенные смежные вершины). Ч.т.д.

Эту теорему можно выразить еще и так

$$\gamma(G \times H) \leq \min \{ \gamma(G) \times \gamma(H) \}.$$

6.1.3. Число внутренней устойчивости

Рассмотрим граф $G = (X, \Gamma)$. Множество $S \subset X$ называется *внутренне устойчивым*, если никакие две вершины не смежны. Другими словами, если

$$\Gamma S \cap S = \emptyset$$

Обозначим через \mathfrak{S} семейство всех внутренне устойчивых множеств графа. Имеем

$$\emptyset \in \mathfrak{S}$$

$$S \in \mathfrak{S}, A \subset S \Rightarrow A \in \mathfrak{S}$$

По определению, *число внутренней устойчивости* графа G есть

$$a(G) = \max_{S \in \mathfrak{S}} |S|$$

ПРИМЕР 1 (Гаусс). *Задача о восьми ферзях*

Можно ли на шахматной доске расставить восемь ферзей так, чтобы ни один из *них* не находился под ударом какого-либо другого. Эта известная задача сводится к нахождению наибольшего

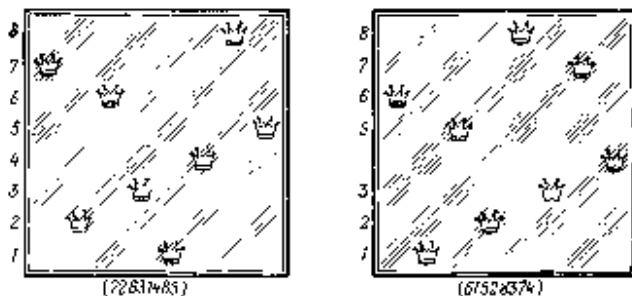


Рис. 5

внутренне устойчивого множества в симметрическом графе с 64 вершинами (клетки шахматной доски), где $y \in \Gamma x$, когда клетки x и y находятся на одной и той же горизонтали, вертикали или диагонали. Задача значительно труднее, чем кажется на первый взгляд. Гаусс первоначально предполагал 76 решений. На самом же деле существует 92 решения, как показывают следующие 12 диаграмм (72631485) (61528374) (58417263)
 (35841726) (46152837) (57263148)
 (16837425) (52763184) (48157263)
 (51468273) (42751863) (35281746)

Каждой перестановке отвечает диаграмма как на рис. 5, а каждая диаграмма кроме той, которая соответствует последней перестановке, порождает восемь различных решений: три возникают при повороте

данной диаграммы на 90° , 180° и 270° , а остальные четыре — при зеркальном отображении каждой из четырех получившихся диаграмм относительно главной диагонали. Последняя перестановка (35281746) дает только четыре решения, так как соответствующая диаграмма, при повороте доски на 180° , переходит сама в себя.

ПРИМЕР 2. *Кликой* симметрического графа называется такое множество $C \subset X$ что

$$x \in C, y \in C \Rightarrow y \in Gx$$

Если X — множество лиц, а $y \in Gx$ означает согласие между x и y , то часто требуется разыскать максимальную клику, т. е. общество с наибольшим числом членов. Определим отображение Γ следующим образом

$$y \in \Gamma x \Leftrightarrow y \notin Gx$$

Задача сводится к нахождению наибольшего внутренне устойчивого множества в графе (X, Γ)

ПРИМЕР 3. *Задача о пятнадцати девушках*, которая была предметом многочисленных математических работ, может быть сформулирована следующим образом: в пансионе воспитываются пятнадцать молодых девушек, которых мы обозначим буквами $a, b, c, d, e, f, g, h, i, j, k, l, m, n, o$, они ежедневно выходят на прогулку тройками. Возможно ли в течение семи дней составлять тройки таким образом, чтобы никакие две девушки не попадали в одну и ту же тройку более одного раза?

Пользуясь соображениями симметрии Кэли нашел следующее решение

Вспомогательная	Пятнадцатеричная	Шестнадцатеричная	Средняя	Четырнадцатеричная	Пятидесятая	Семидесятая
ak	abe	afci	ado	agn	ahj	ac
fgl	enp	bcf	bek	bdj	hino	blq
chm	dfl	def	chl	cek	cdg	dkm
djn	glik	go	egmi	fmh	ef	eln
ejp	hln	iku	fmn	hnl	klo	fgj

Задача о пятнадцати девушках родственна другой, столь же известной задаче, которую назовем *вспомогательной задачей*: можно ли из 15 девушек последовательно составить 35 различных троек так, чтобы никакие две девушки не входили вместе более чем в одну тройку? Для решения вспомогательной задачи достаточно построить граф G , вершинами которого служат все 455 возможных троек, причем две тройки считаются смежными, когда в эти тройки входят одни и те же две девушки. Тогда требуется найти наибольшее внутренне устойчивое множество. Имеем $\alpha(G) \leq 35$, так как никакая девушка не может входить более, чем в семь различных троек, а это

дает самое большее $15 \times 7 \times \frac{1}{3} = 35$ троек, поэтому всякое внутренне

устойчивое множество из 35 троек максимально.

Чтобы показать, что решение вспомогательной задачи дает также ответ на задачу о пятнадцати девушках, построим граф G' , вершинами которого служат какие-либо 35 троек, дающие решение вспомогательной задачи, причем две тройки считаются смежными, когда в них входит одна и та же девушка. Если хроматическое число $\gamma(G')$ равно 7, то задача о пятнадцати девушках решена. Если же $\gamma(G') > 7$, то надо выбрать другое множество троек, дающее решение вспомогательной задачи. Нетрудно проверить, что существуют такие решения вспомогательной задачи, которые не приводят к решению задачи о пятнадцати девушках.

Замечание 1. Хроматическое число $\gamma(G)$ и число внутренней устойчивости $\alpha(G)$ связаны неравенством

$$\alpha(G), \gamma(G) \geq |X|$$

В самом деле, можно разбить X на $\gamma(G)$ внутренне устойчивых множеств, образованных вершинами одинакового цвета и содержащих соответственно $m_1, m_2, \dots, m_\gamma$ вершин. Поэтому

$$\begin{aligned} |X| &= m_1 + m_2 + \dots + m_\gamma + \\ &+ m_\gamma \leq \alpha(G) + \alpha(G) + \dots + \alpha(G) = \gamma(G) \alpha(G) \end{aligned}$$

Зачечание 2. Можно поставить вопрос: не являются ли связи между обоими понятиями более тесными и нельзя ли найти хроматическое число окрашивая сначала в цвет (1) наибольшее внутренне устойчивое множество S_1 , затем в цвет (2) наибольшее внутренне устойчивое множество S_2 подграфа, порожденного вершинами $X \setminus S_1$, далее в цвет (3) наибольшее внутренне устойчивое множество оставшегося подграфа, и т. д. Оказывается, это не так, что видно на примере графа, изображенного на рис. 6 (его хроматическое число, очевидно, равно 4), вершины, изображенные белыми кружками образуют единственное наибольшее внутренне устойчивое множество. Но если их окрасить в один цвет, то остальные вершины a, b, e, d надо было бы окрасить с помощью только трех цветов, а это, очевидно, невозможно.

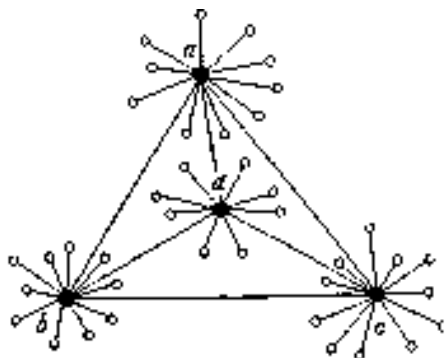


Рис. 6

Иногда для двух графов G и H возникает вопрос о нахождении числа внутренней устойчивости произведения $G \times H$

ПРИМЕР (Шеннон). *Задача об информационной емкости множества сигналов.*

Рассмотрим простейший случай одного передатчика, который может передавать пять сигналов a, b, c, d, e . При приеме каждый из этих сигналов может быть истолкован двояко: сигнал a дает p или q , сигнал b дает q или r , и т. д., как показано на схеме, изображенной на рис. 7.

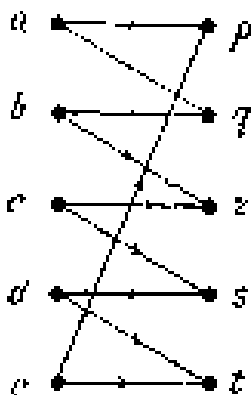


Рис. 7

Какое наибольшее число сигналов можно принять, не рискуя спутать их друг с другом? Задача сводится к нахождению наибольшего внутренне устойчивого множества S графа G (рис. 8), где две вершины

смежны, если они представляют такие сигналы, которые можно спутать при приеме. Очевидно $\alpha(G)=2$ и можно взять $S = \{a, c\}$.

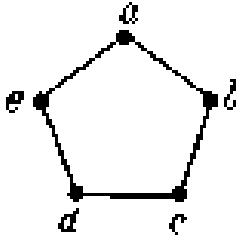


Рис. 8

Вместо однобуквенных сигналов можно пользоваться „словами,, из двух букв с таким расчетом, чтобы эти слова не путались друг с другом при приеме. Из букв a и c , которые нельзя спутать, составляется код: aa, ac, ca, cc . Таким образом, получаются $[\alpha(G)]^4 = 4$ слова. Но можно составить еще более богатый код: aa, bc, ce, db, ed . (Непосредственно проверяется, что никакие два из этих слов спутать одно с другим при приеме нельзя).

Назовем здесь *произведением двух графов* $G=(X, U)$ и $H=(Y, V)$ граф $G \times H$, вершинами которого служат пары xu , где $x \in X, u \in Y$, причем две вершины xu и $x'u'$ смежны, если они удовлетворяют одному из следующих условий:

- 1° $x = x', (y, y') \in V,$
- 2° $(x, x') \in U, y = y';$
- 3° $(x, x') \in U (y, y') \in V.$

Для графа G , изображенного на рис. 8, два слова xu и $x'y'$ могут быть перепутаны, если они являются двумя смежными вершинами графа произведения $G \times G = G^2$ и богатство кода с двухбуквенными словами выражается числом $\alpha(G^2) = 5$, а в более общем случае, богатство кода с n -буквенными словами выражается числом внутренней устойчивости графа-произведения

$$G^n = G \times G \times \dots \times G$$

Лемма 1. Для двух графов G и H

$$\alpha(G \times H) \geq \alpha(G) \times \alpha(H).$$

Действительно, если S и T — наибольшие внутренние устойчивые множества, соответственно для G и H , то декартово произведение $S \times T$ является внутренне устойчивым в графе $G \times H$, откуда

$$\alpha(G \times H) \geq |S \times T| = |S| \times |T| = \alpha(G) \times \alpha(H).$$

Эта лемма подсказывает следующее определение: назовем емкостью графа G число

$$\theta(G) = \sup_n \sqrt[n]{\alpha(G^n)}$$

Имеем $\theta(G) \geq \alpha(G)$. Мы собираемся показать, что почти всегда $\theta(G) = \alpha(G)$

Между прочим, Шеннон установил, что граф G , изображенный на рис 8, является единственным графом с числом вершин менее шести, для которого $\theta(G) \neq \alpha(G)$, фактически его емкость $\theta(G)$ не удалось определить, и известно лишь, что

$$\sqrt{5} \leq \theta(G) \leq \frac{5}{2}$$

Рассмотрим однозначное отображение σ множества X в себя. Такое отображение называется *сохранным*, если

$$y \neq x, \quad y \notin \Gamma x \Rightarrow \sigma(y) \neq \sigma(x), \quad \sigma(y) \notin \Gamma \sigma(x)$$

Это отображение сохраняет свойство пары вершин «быть несмежными и различными».

Лемма 2. *Сохранное отображение σ переводит внутренне устойчивое множество S во внутренне устойчивое множество $\sigma(S)$, и при этом $|S| = |\sigma(S)|$*

В самом деле, ввиду однозначности отображения σ имеем $|\sigma(S)| \leq |S|$, а так как σ сохранно, то $|\sigma(S)| = |S|$.

Лемма 3. *Если множество $\sigma(X)$ внутренне устойчиво, то число внутренней устойчивости графа G есть $\alpha(G) = |\sigma(X)|$.*

Действительно, раз $\sigma(X)$ внутренне устойчиво, то

$$|\sigma(X)| \leq \max_{S \in \mathfrak{S}} |S| = \alpha(G).$$

С другой стороны, если S_0 —наибольшее внутренне устойчивое множество, то в силу леммы 2

$$|\sigma(X)| \geq |\sigma(S_0)| = |S_0| = \alpha(G).$$

Отсюда

$$\alpha(G) = |\sigma(X)|.$$

Теорема. (Шеннон) *Если хотя бы для одного из графов G и H существует сохранное отображение σ , переводящее множество вершин этого графа во внутренне устойчивое множество, то*

$$\alpha(G \times H) = \alpha(G) \alpha(H).$$

Достаточно показать, что $\alpha(G \times H) \leq \alpha(G) \alpha(H)$. Пусть σ — сохранное отображение для G , при котором $\sigma(X)$ внутренне устойчиво, и пусть σ_0 — отображение множества вершин графа $G \times H$ в себя, определенное следующим образом

$$\sigma_0(x, y) = [\sigma(x) \ y].$$

Отображение σ_0 переводит две несмежные различные вершины $\xi=(x, y)$ $\xi' = (x' y')$ в две несмежные различные вершины $(\sigma x, y)$ и $(\sigma x' y')$ и поэтому сохранно.

Если S_0 — наибольшее внутренне устойчивое множество графа $G \times H$, то $\alpha(G \times H) = |S_0| = |\sigma_0(S_0)|$ в силу леммы 2, распределим элемент $\sigma_0(S_0)$ по разным классам в зависимости от первой буквы каждого слова. Согласно лемме 3 получим $|\sigma(X)| = \alpha(G)$ различных классов. Поскольку никакие два элемента из $\sigma_0(S_0)$ не смежны, каждый класс имеет самое большее $\alpha(H)$ элементов, значит

$$\alpha(G \times H) = |\sigma_0(S_0)| \leq \alpha(G) \alpha(H)$$

Следствие. Если множество вершин графа G при помощи сохранного отображения σ можно перевести во внутренне устойчивое множество, то емкость этого графа совпадает с числом внутренней устойчивости

В самом деле

$$\alpha(G \times G) = [\alpha(G)]^2$$

$$\alpha(G \times G \times G) = [\alpha(G)]^2 \alpha(G) = [\alpha(G)]^3$$

и т. д.

отсюда

$$\theta(G) = \sup_n \sqrt[n]{\alpha(G^n)} = \alpha(G).$$

6.1.4. Число внешней устойчивости

Пусть дан граф $G = (X, \Gamma)$. Говорят что множество $T \subset X$ внешне устойчиво, если для каждой вершины $x \notin T$ имеем $\Gamma x \cap T \neq \emptyset$.

Если \mathfrak{T} — семейство всех внешне устойчивых множеств графа, то

$$X \in \mathfrak{T}$$

$$T \in \mathfrak{T}, A \supset T \Rightarrow A \in \mathfrak{T}$$

По определению, число внешней устойчивости графа G есть

$$\beta(G) = \min_{T \in \mathfrak{T}} |T|$$

Задача, которая нас интересует, заключается в построении внешне устойчивого множества с наименьшим числом элементов

ПРИМЕР 1. Задача о часовых. В тюрьме города N около каждой камеры может быть поставлен часовой. Однако часовой стоящий, например, у камеры x_0 видит также, что происходит в камерах x_1, x_2, x_3 и x_4 , от каждой из которых к камере x_0 идет коридор как показано на рис. 9.

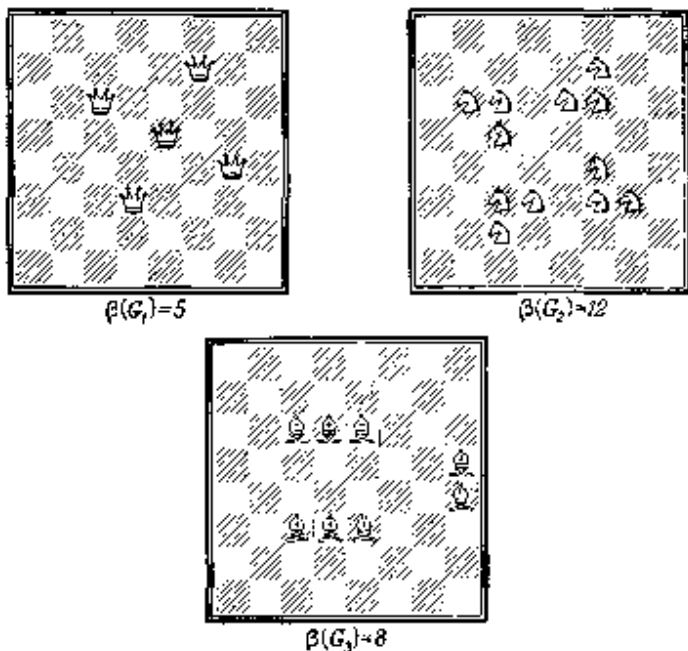


Рис. 10

ПРИМЕР 3. Задача о шести красных дисках

Любителям французских ярмарок хорошо известна следующая игра: большой белый диск (радиус которого примем за единицу) лежит на столе. Требуется потностью покрыть его шестью маленькими красными дисками (радиуса $\rho < 1$), которые последовательно кладут на стол, а положив, более уже не перемешают. При каком наименьшем радиусе ρ это возможно?

Задача сводится к нахождению наименьшего внешне устойчивого множества T для бесконечного графа (X, U) , где X —множество точек белого диска, а $(x, y) \in U$, когда расстояние $d(x, y)$ между x и y не превышает ρ .

Для отыскания наибольшего внутренне устойчивого множества или наименьшего внешне устойчивого множества можно последовательно рассматривать каждое множество $T \subset X$ и проверять, удовлетворяет ли оно всем требуемым условиям. Конечно, в общем случае этот метод исключения практически недоступен, и надо прибегнуть к какому-либо алгоритму, т.е. к сокращенному способу таких вычислений.

Ниже мы изложим такой алгоритм для определения наибольшего внутренне устойчивого множества, непосредственно вытекающий из дальнейших теорий (для бихроматических графов и для общего случая). Так, например, решить задачу о восьми ферзях при наличии хорошего алгоритма можно менее чем за три минуты, в то время как способ исключений потребует десятка часов (без привлечения ЭВМ). Кроме того, следует заметить, что для конкретной расстановки невозможность увечичения числа ферзей доказывается весьма простым рассуждением; в более общем случае тоже нужен простой критерий для проверки: является ли внутренне устойчивое множество наибольшим, и только хорошо задуманный алгоритм может нас удовлетворить

Нахождение наименьшего внешне устойчивого множества представляет собой более элементарную задачу, которую можно решать следующим образом

Алгоритм для нахождения наименьшего внешне устойчивого множества. Рассмотрим для примера граф $G=(X, \Gamma)$ изображенный на рис. 11 и определим отображение Δ множества $X=\{ a, b, \dots \}$ и новое множество

$$\bar{X} = \{ \bar{a}, \bar{b}, \dots, \}$$

следующим образом

$$\bar{y} \in \Delta x \Leftrightarrow y=x$$

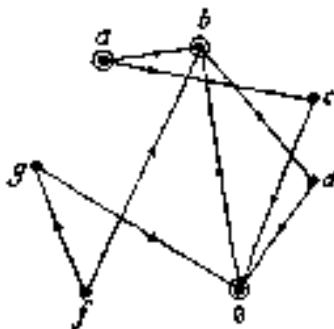


Рис. 11

Тем самым построен так называемый простой граф, который мы обозначим через (X, \bar{X}, Δ) (рис. 12), если T — внешне устойчивое

множество графа G , то $\Delta T = \bar{X}$, наоборот, если $\Delta T = \bar{X}$, то множество T внешне устойчиво в G . Задача свелась, таким образом, к определению наименьшего множества $T \subset X$, для которого $\Delta T = \bar{X}$.

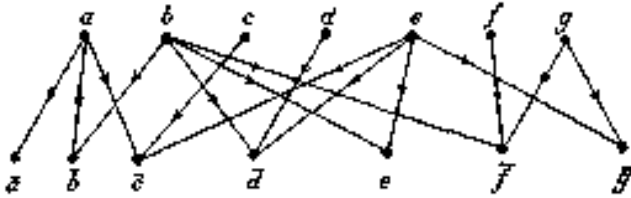


Рис. 12

1° Удаляем из простого графа каждую такую вершину x что, $\Delta x \subset \Delta y$ для некоторой вершины $x \neq y$ (в самом деле, с точки зрения нашей задачи, вершина y будет почностью заменять вершину x). В нашем примере мы удаляем, таким образом вершины c, d, f .

2° Если в простом графе имеется висячее ребро (x, \bar{y}) , то, очевидно, $x \in T$. В данном примере множеству T заведомо принадлежит вершина a .

3° Исключим из простого графа вершину a , уже входящую в T , и множество $\Delta a = \{\bar{a}, \bar{b}, \bar{c}\}$, в результате получается граф, изображенный на рис. 13.

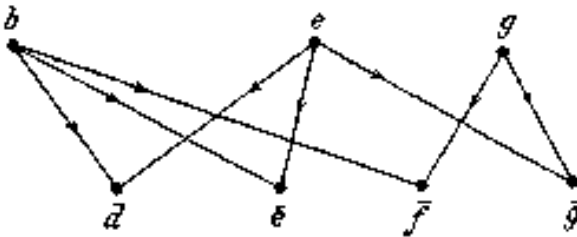


Рис. 13

4° Снова пытаемся удалить некоторую вершину, как в 1° или исключить вершину, заведомо принадлежащую T , как в 2°. Если упростить граф уже нельзя (как в данном примере), то назовем его неприводимым. Временно отнесем в T произвольную вершину, скажем b .

5° Исключим, как в 3°, вершину b и множество $\Delta b = \{\bar{d}, \bar{e}, \bar{f}\}$.

6° Продолжаем упрощение, как выше. Из полученного графа можно исключить вершину g , так как $\Delta g \subset \Delta e = \{\bar{g}\}$. Включая в T последнюю вершину e , получаем решение $T = \{a, b, e\}$.

Ести в 4° попробовать вместо b включить в T вершину g или e , то результат не улучшится, поэтому найденное в 6° решение удовлетворяет поставленным условиям

6.2. Ядра графа

6.2.1 Теоремы существования и единственности

Пусть $G = (X, \Gamma)$ — конечный или бесконечный граф. Множество $S \subset X$ называется *ядром* графа, если S устойчиво как внутренне, так и внешне, т. е. если

$$x \in S \Rightarrow \Gamma x \cap S = \emptyset \quad (1)$$

$$x \notin S \Rightarrow \Gamma x \cap S \neq \emptyset \quad (2)$$

Из условия (1) вытекает, что ядро S не содержит петель, из условия (2) — что S содержит все такие вершины x , для которых $\Gamma x = \emptyset$, очевидно, пустое множество \emptyset не может быть ядром

ПРИМЕР 1 (фон Нейман — Моргенштерн). Понятие ядра первоначально было введено в теории игр под названием „решения“.

Пусть имеется n игроков (1), (2), ..., (n), которые могут советоваться при выборе ситуации x из множеств X , если игрок (i) предпочитает ситуацию a ситуации b , то пишем $a \geq^i b$ и ясно, что \geq^i есть тотальный квазипорядок. Так как отдельные предпочтения \geq^i могут не быть совместимы, то приходится вводить понятие *эффективного предпочтения* \mathbf{f} . Если ситуация a эффективно предпочитается ситуации b , то это означает, что существует множество игроков, которые, считая a лучше b , имеют возможность добиться того, чтобы удобная им точка зрения одержала верх, в этом случае пишем $a \mathbf{f} b$. Если кроме того, $b \mathbf{f} c$, то существует множество игроков, способных сделать так, чтобы одержала верх ситуация b , но так как второе множество не обязательно совпадает с первым, то может не быть $a \mathbf{f} c$, отношение \mathbf{f} не транзитивно.

Теперь рассмотрим граф (X, Γ) , где Γx означает множество ситуаций, эффективно предпочитаемых ситуации x . Пусть S — ядро графа (если оно существует). Фон Нейман и Моргенштерн предлагают ограничивать игру элментами S . Внутренняя устойчиость S выражает тот факт, что никакая ситуация из S не может эффективно предпочитаться какой-либо другой ситуации из S , а это обеспечивает известную сплоченность. Внешняя устойчиость означает, что любой ситуации x не из S эффективно предпочитается некоторая ситуация из S , в силу чего ситуация x сразу же отвергается

ПРИМЕР 2. Рассмотрим некоторый граф (X, Γ) , допускающий функцию Гранди $g(x)$; множество $S = \{x/g(x) = 0\}$ служит ядром этого графа, так как

$$x \in S \Rightarrow \Gamma x \cap S = \emptyset. \quad (1)$$

$$x \notin S \Rightarrow g(x) \neq 0, \quad \text{значит } \Gamma x \cap S \neq \emptyset \quad (2)$$

ПРИМЕР 3. Не всякий граф обладает ядром. В этом читатель может самостоятельно убедиться на примере графа изображенного на рис. 1

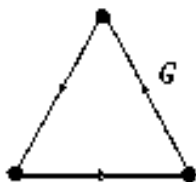


Рис. 1

если граф имеет ядро S_0 , то его числа устойчиости удовлетворяют условию

$$\alpha(G) \max_{S \in \emptyset} |S| \geq |S_0| \geq \min_{T \in \exists} |T| = \beta(G).$$

У графа G , изображенного на рис. 1, нет ядра, поскольку

$$\alpha(G) = 1 < 2 = \beta(G).$$

Напротив, у графа G' , изображенного на рис. 2, есть два ядра S' и S'' .

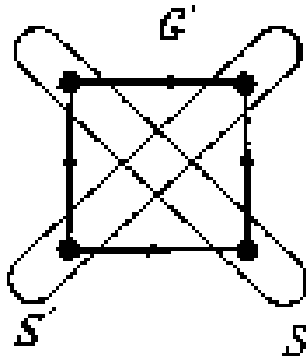


Рис.2.

Предложим теперь критерии, позволяющие узнавать, имеет ли граф ядро и является ли оно единственным.

Теорема 1. Если S — ядро графа (X, Γ) , то множество S — максимальное в семействе \wp внутренне устойчивых множеств,

$$A \in \wp, A \supset S \Rightarrow A = S.$$

Пусть A , внутренне устойчивое множество, содержащее ядро S . Предположим, что A строго содержит S и покажем, что это приводит к противоречию. В самом деле, тогда существовала бы такая вершина a , что $a \in A, a \notin S$, откуда $\Gamma a \cap S = \emptyset$ и, значит, $\Gamma a \cap A \neq \emptyset$ в противоречии с условием $A \in \wp$.

Теорема 2. В симметрическом графе без петель каждое максимальное множество семейства \wp внутренне устойчивых множеств представляет собой ядро

Пусть S — максимальное множество из \wp ; надо показать что для любой вершины $x \notin S$ имеет место $\Gamma x \cap S \neq \emptyset$. В самом деле если $\Gamma x \cap S = \emptyset$ для некоторой вершины $x \in S$, то множество $A = S \cup \{x\}$ внутренне устойчиво (поскольку $x \notin \Gamma x$) и в то же время $A \supset S, A \neq S$, что противоречит предположению о максимальной S в \wp .

Следствие. Симметрический граф без петель обладает ядром.

В самом деле, образуем вспомогательный граф $(\wp, \bar{\Gamma})$, вершинами которого служат внутренне устойчивые множества данного симметрического графа, а $S \in \bar{\Gamma} S'$ тогда и только тогда, когда $S \supset S'$. Вспомогательный граф — индуктивный, следовательно по лемме

Цорна, существует вершина $S \in \wp$ без строго последующих. Множество S является максимальным внутренне устойчивым и значит, в силу теоремы 2, ядром.

В случае, когда данный симметрический граф конечен, это следствие становится очевидным и процесс нахождения ядра состоит в следующем.

Берем произвольную вершину x_0 и полагаем $S_0 = \{x_0\}$; затем берем некоторую вершину $x_1 \notin GS_0$ и полагаем $S_1 = \{x_0, x_1\}$, далее берем вершину $x_2 \notin GS_1$, и т. д. Так как граф конечен, то рано или поздно мы получим $GS_n = X$ и S_n , как максимальное множество в \wp , будет ядром. *Характеристической функцией* $\varphi_S(x)$ множества S называется функция

$$\varphi_S(x) = \begin{cases} 1 & \text{при } x \in S \\ 0 & \text{при } x \notin S \end{cases}$$

Если $Gx = \emptyset$, то условимся считать, что

$$\max_{y \in Gx} \varphi_S(y) = 0$$

Теорема 3. *Для того чтобы множество S было ядром, необходимо и достаточно, чтобы для характеристической функции $\varphi_S(x)$ выполнялось соотношение*

$$\varphi_S(x) = 1 - \max_{y \in Gx} \varphi_S(y)$$

1° Пусть S — ядро. В силу внутренней устойчивости

$$\varphi_S(x) = 1 \Rightarrow x \in S \Rightarrow \max_{y \in Gx} \varphi_S(y) = 0.$$

В силу внешней устойчивости

$$\varphi_S(x) = 0 \Rightarrow x \notin S \Rightarrow \max_{y \in Gx} \varphi_S(y) = 1.$$

Отсюда получается требуемое соотношение

2° Пусть $\varphi_S(x)$ — характеристическая функция некоторого множества S , если рассматриваемое соотношение выполнено, то

$$x \in S \Rightarrow \varphi_S(x) = 1 \Rightarrow \max_{y \in Gx} \varphi_S(y) = 0 \Rightarrow Gx \cap S = \emptyset,$$

$$x \notin S \Rightarrow \varphi_S(x) = 0 \Rightarrow \max_{y \in Gx} \varphi_S(y) = 1 \Rightarrow Gx \cap S \neq \emptyset.$$

Следовательно S — ядро.

Теорема 4. *Прогрессивно конечный граф обладает ядром. Доказательство* получается сразу, если заметить, что характери-

стическая функция $\varphi_S(x)$, удовлетворяющая соотношению предыдущей теоремы, по индукции определяется на множествах

$$\begin{aligned} X(0) &= \{x/\Gamma x = \emptyset\}, \\ X(1) &= \{x/\Gamma x \subset X(0)\}, \\ X(2) &= \{x/\Gamma x \subset X(1)\}, \\ &\dots\dots\dots \end{aligned}$$

Таким путем $\varphi_S(x)$ будет определена на всем X .

Теорема Ричардсона. *Конечный граф, не содержащий контуров нечетной длины, обладает ядром*

Пусть (X, Γ) — конечный граф без нечетных контуров. будем последовательно определять множества $Y_0, Y_1, Y_2, \dots \subset X$ следующим образом

1° Берем $Y_0 = \emptyset$, обозначим через B_0 базу подграфа, порождаемого множеством $X \setminus Y_0$; эта база существует в силу известной теоремы.

Полагаем $Y_1 = B_0 \cup \Gamma^{-1} B_0$.

2° Если множество Y_n уже определено, то обозначим через B_n какую-либо базу подграфа, порождаемого множеством $X \setminus Y_n$, удовлетворяющую условию $B_n \subset \Gamma^{-1}(\Gamma^{-1} B_{n-1} \setminus Y_{n-1})$. Легко видеть, что такая база всегда существует, далее полагаем

$$Y_{n+1} = Y_n \cup B_n \cup \Gamma^{-1} B_n$$

Тогда

$$\emptyset = Y_0 \subset \subset Y_1 \subset \subset Y_2 \subset \subset \dots$$

Так как граф предполагается конечным, то существует такой номер m что $Y_m = X$, пусть

Покажем что S — ядро графа (X, Γ)

1° S внешне устойчиво, ибо если $x \notin S$, то $x \in \Gamma^{-1} B_k \setminus Y_k$ для некоторого номера k , значит $\Gamma x \cap B_k \neq \emptyset$ и $\Gamma x \cap S \neq \emptyset$.

2° S внутренне устойчиво. В самом деле, никакие два элемента из B_n не могут быть смежны (ибо B_n является базой некоторого подграфа).

Рассмотрим две смежные вершины, одна из которых $\in B_n$, другая $\in B_p$, где $p < n$ (если такие вершины есть). Имеем $B_n \cap \Gamma^{-1} B_p = \emptyset$, ибо

$\Gamma^{-1} B_p \subset Y_{p+1}$ и $B_n \subset X \setminus Y_n \subset X \setminus Y_{p+1}$. Точно так же $B_p \cap \Gamma^{-1} B_n = \emptyset$, так как в противном случае можно было бы построить такой путь

$\mu = [x_0, x_n, y_{n-1}, x_{n-1}, y_{n-2}, x_{n-2}, \dots, y_p, x_p]$, что

6.2.2. Приложение к функциям Гранди

Условия существования, установленные для ядер, можно распространить на функции Гранди благодаря ступенчатым результатам

Теорема 5. *Если каждый подграф графа (X, Γ) обладает ядром, то граф (X, Γ) допускает функцию Гранди.*

В самом деле, пусть S_0 — ядро графа (X, Γ) , далее S_1 — ядро подграфа, порожденного множеством $X_1 = X_0 \setminus S_0$, затем S_2 — ядро подграфа, порожденного множеством $X_2 = X_1 \setminus S_1$ и т. д. Вообще положим $X_\alpha = X_{\alpha-1} \setminus S_{\alpha-1}$, если α — порядковое число первого рода, $X_\alpha = \bigcap_{\beta < \alpha} X_\beta$, если α — порядковое число второго рода

(заметим что $\beta > \alpha$ влечет $X_\alpha \supset \supset X_\beta$, и в силу принципа индукции, существует порядковое число γ , при котором $X_\gamma = \emptyset$). Множества X_α образуют разбиение X , и каждой вершине x мы отнесем порядковое число $g(x)$ следующим образом

$$g(x) = \alpha \Leftrightarrow x \in X_\alpha.$$

Покажем, что $g(x)$ — функция Гранди

(1) Пусть $g(x) = \alpha$, покажем, что для любого $\beta < \alpha$ найдется вершина $y \in Gx$, удовлетворяющая условию $g(y) = \beta$.

Так как $x \in S_\alpha$ и $\alpha > \beta$, то $x \in X_\alpha \subset X_\beta$, $x \notin S_\beta$. Так как S_β внешне устойчивое множество подграфа, порожденного множеством X_β , то в S_α существует вершина $y \in Gx$, иначе говоря, в Gx есть такая вершина y , что $g(y) = \beta$.

(2) Пусть $g(x) = \alpha$, в Gx не существует такой вершины y , что $g(y) = \alpha$, ибо, иначе, множество S_α не было бы внутренне устойчивым

Условия (1) и (2) вместе выражают, что $g(x)$ есть функция Гранди

Следствие 1. *Семиметрический граф допускает функцию Гранди в том и только в том случае, если он не имеет петель.*

(Ясно, что граф с петлями не может допускать функций Гранди)

Следствие 2. *Граф без контуров нечетной длины, являющийся или конечным или локально конечным, или индуктивным, допускает функцию Гранди*

Замечание. Пусть графы G_1, G_2, \dots, G_n , обладают ядрами.

Имеет ли ядро их сумма (или их произведение)? Для определения ядра суммы графов служит понятие функции Гранди (впрочем, это мотивирует также важность понятия ядра для теории функций Гранди).

Используя теорему 5, мы видим, что если все подграфы графа G_i обладают ядрами (при всех i), то сумма

$$G_1 + G_2 + \dots + G_n$$

допускает функцию Гранди g , а значит и ядро $S = \{x/g(x) = 0\}$.

Ниже мы познакомимся с применениями этого важного замечания.

6.3. Игры на графе

6.3.1. Игра Ним

Граф (X, Γ) дает возможность определить некоторую игру двух игроков, которых мы назовем (A) и (B) . Положениями этой игры служат вершины графа. Начальная вершина x_0 выбирается жеребьевкой и противники играют поочередно: сперва игрок (A) выбирает вершину x_1 в множестве Γx_0 , затем (B) выбирает вершину x_2 в множестве Γx_1 , после этого (A) опять выбирает вершину x_3 в Γx_2 и т. д. Если один из игроков выбрал вершину x_k , для которой $\Gamma x_k = \emptyset$, то партия оканчивается. Игрок, выбравший вершину последним выиграл, а его противник проиграл. Ясно, что если граф не является прогрессивно конечным, то партия может никогда не кончиться.

В честь известного развлечения, которое здесь обобщено, будем, описанную только что игру, называть *игрой Ним*, а определяющий ее граф обозначать через (X, Γ) . Сейчас наша задача состоит в том, чтобы характеризовать выигрышные положения, т.е. те вершины графа, выбор которых обеспечивает выигрыш партии независимо от ответа противника. Главным результатом является следующая

Теорема 1. Если граф имеет ядро S и если один из игроков выбрал вершину в ядре, то этот выбор обеспечивает ему выигрыш или ничью.

Действительно, если игрок (A) выбрал вершину $x_1 \in S$, то либо $\Gamma x_1 = \emptyset$ и тогда он уже выиграл партию, либо его противник (B) вынужден выбрать вершину $x_2 \in X \setminus S$, а значит следующим ходом игрок (A) может выбрать x_3 опять в S и продолжать в том же духе. Если в какой-либо определенный момент один из игроков выиграл, выбрав вершину x_k , для которой $\Gamma x_k = \emptyset$, то $x_k \in S$ и выигравшим партнером необходимо является (A) . Ч. и т.д.

Основной метод для хорошего игрока состоит, следовательно, в вычислении какой либо функции Гранди, если она существует. С помощью этой функции $g(x)$ получаем ядро

Хотя последние образуют множество, симметричное относительно главной диагонали, в их расположении наблюдается нерегулярность.

ПРИМЕР 2. Граф, изображенный на рис. 2, представляет игру Ним для которой не существует ни функции Гранди, ни ядра; предыдущий метод, поэтому здесь неприменим.

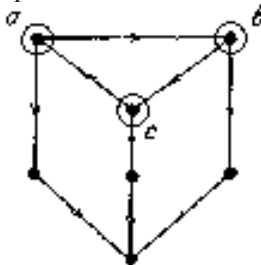


Рис. 2

Тем не менее, каждый игрок может застраховать себя от проигрыша ограничивая свой выбор одной из вершин a , b , c .

Рассмотрим теперь различные игры Ним

(X_1, Γ_1) , (X_2, Γ_2) , ..., (X_n, Γ_n) и допустим, что два игрока хотят играть по очереди по следующему правилу: в очередной момент игры партнер, которому нужно ходить, бирает одну из игр Ним и делает ход в этой игре, не трогая остальных. Первый, кто совсем не сможет играть, проигрывает. Эта ситуация претставляет собой игру Ним на графе, который есть не что иное, как сумма графов (X_i, Γ_i) , способ выигрыша дается следующим результатом

Теорема 2. *Рассмотрим игры (X_1, Γ_1) , (X_2, Γ_2) , ..., (X_n, Γ_n) , допускающие соответственно функции Гранди g_1, g_2, \dots, g_n ; если играют в сумму этих игр, то выигрыш или ничья обеспечены выбором таких положений*

$$x = (x_1, x_2, \dots, x_n) \in X_1 \times X_2 \times \dots \times X_n$$

для которых

$$g(x) = g_1(x_1) + g_2(x_2) + \dots + g_n(x_n) = 0.$$

Это непосредственно вытекает из известной теоремы.

ПРИМЕР 3. *Простая игра Нам или Фан Тан.* Два игрока поочереди берут спички из n кучек. Игрок, которому нужно ходить, имеет право выбрать какую-нибудь кучку (не пустую) и взять из нее одну или несколько спичек. Тот, кто сумеет забрать последние спички, выигрывает.

Эта ситуация есть сумма игр $(X_1, \Gamma_1), (X_2, \Gamma_2), \dots, (X_n, \Gamma_n)$; в игре (X_k, Γ_k) любой элемент x_k из X_k выражает состояние k -й кучки, а $g_k(x_k)$ равно соответствующему числу спичек.

В силу теоремы 2 выбор является выигрышным тогда и только тогда, когда d -сумма количеств спичек, оставшихся в различных кучках, равна нулю.

Мы собираемся рассмотреть различные обобщения теоремы 2, прежде всего, если $p \leq n$ и если $(X_1, \Gamma_1), (X_2, \Gamma_2), \dots, (X_n, \Gamma_n)$ — данные игры Ним, то их сумма порядка p по определению есть игра Ним, в которой каждый игрок, имея перед собой все игры (X_k, Γ_k) , выбирает p из них и играет одновременно в эти p игры, не трогая остальных, сумма порядка n соответствует произведению графов, а сумма порядка 1 — сумме графов. Для выигрыша имеем следующий результат

Теорема 3. Пусть игры Нам $(X_1, \Gamma_1), (X_2, \Gamma_2), \dots, (X_n, \Gamma_n)$ транзитивны (т. е. $\Gamma_k = \Gamma_l$ для всех k, l), и пусть g_1, g_2, \dots, g_n — их функции Гранди (она всегда существуют) сумма порядка p этих n игр допускает функцию Гранди g , значение которой для $x = \{x_1, x_2, \dots, x_n\}$ вычисляется следующий образом: найдем двоичные разложения

$$g_k(x_k) = (c^1_k c^2_k \dots)$$

и положим

$$g(x) = \left[\sum_k c_k \right]_{p+1} + (p+1) \left[\sum_k c_k \right]_{(p+1)+} + (p+1)^2 \left[\sum_k c_k^2 \right]_{(p+1)+} +$$

где $[m]_{p+1}$ означает остаток от деления m на $p+1$

(„ m по модулю $p+1$ “)

Выбор x является выигрышным тогда и только тогда, когда $g(x) = 0$

ПРИМЕР. Игры Ним порядка p (Мур). Перед двумя игроками находятся n кучек спичек. Пусть p — целое число $< n$, игрок, который должен сделать очередной ход, выбирает p кучек, а из каждой выбранной кучки вынимает одну или несколько спичек. Взявший последние спички, выигрывает.

Чтобы узнать, является ли выбор $x = (x_1, x_2, \dots, x_n)$ выигрышным, надо сосчитать в двоичной системе количество спичек

$$g_k(x_k) = (c^1_k c^2_k \dots)$$

для k -й кучки — и проверить, что для каждого номера r

$$\left[\sum_{k=1}^n c_k^r \right]_{(p+1)} = 0$$

Рассмотрим теперь прогрессивно конечную игру Ним с функцией Гранди $g(x)$. Предположим, что на X определена операция $+$, т. е. закон относящий каждой паре

$$(x, y) \in X \times X$$

некоторую вершину $z = x+y$, если $S \subset X \quad T \subset X$, то положим

$$S + T = \{s+t \in S \quad t \in T\}.$$

Часто для выигрыша удается применить следующий результат

Теорема 4 (Гранди). *Если прогрессивно конечный граф (X, Γ) снабжен операцией $+$, удовлетворяющей условию*

$$\Gamma(x+y) = (\Gamma x + y) \cup (x + \Gamma y),$$

то для его функции Гранди g при $z = x+y$ имеет место формула

$$g(z) = g(x+y) = g(x) + g(y) \tag{1}$$

Мы докажем эту формулу с помощью индукции, выбирая последовательно точки $z = x+y$ в множествах

$$X(0) = \{x \in X \mid \Gamma x = \emptyset\},$$

$$X(1) = \{x \in X \mid \Gamma x \subset X(0)\},$$

$$X(2) = \{x \in X \mid \Gamma x \subset X(1)\},$$

.....

При $z \in X(0)$ формула (1) верна потому, что

$$z \in X(0) \Rightarrow (\Gamma x + y) \cup (x + \Gamma y) = \Gamma z = \emptyset \Rightarrow x, y \in X(0),$$

откуда

$$g(z) = 0 = 0 + 0 = g(x) + g(y)$$

Допустим теперь, что формула (1) так же доказана для всех z из $X(\alpha-1)$ и покажем, что она остается справедливой для любой вершины $z = x+y$ из $X(\alpha)$. Если бы было не так, то выполнялось бы одно из двух неравенств

$$g(z) > g(x) + g(y)$$

$$g(z) < g(x) + g(y)$$

1° Если $g(z) > g(x) + g(y)$, то существует такая вершина $z_1 \in \Gamma z$, что

$$g(z_1) = g(x) + g(y).$$

Можно написать (с точностью до перестановки x и y) $z_1 = x_1 + y$,

где $x_1 \in \Gamma x$, так как $z_1 \in \Gamma z \subset X(\alpha-1)$, то

$$g(z_1) = g(x_1 + y) = g(x_1) + g(y).$$

Отсюда

$$g(x) + g(y) = g(x_1) + g(y).$$

Из этого, в силу известной теоремы, вытекает, что $g(x) = g(x_1)$, но это невозможно, так как $x_1 \in \Gamma x$.

2° Если $g(z) < g(x) + g(y)$, то можно определить такое число γ , что

$$g(z) = \gamma + g(y), \quad \gamma < g(x).$$

Так как в Γx существует вершина x_i , для которой $\gamma=g(x_i)$, то очевидно

$$g(z) = g(x_i)+g(y).$$

Так как $x_i+y \in \Gamma z \subset X(\alpha-1)$, то

$$g(x_i+y) = g(x_i)+g(y) = g(z).$$

Но это невозможно, поскольку $x_i+y \in \Gamma z$.

ПРИМЕР. Два противника играют поочередно, выбирая кучку спичек среди n кучек и разделяя выбранную кучку на две неравные части, последний, кто еще имеет возможность это сделать, выигрывает.

Положение x задается количествами $\bar{x}_1, \bar{x}_2, \dots, \bar{x}_k$ спичек

в кучках.

Положим

$$\begin{aligned} X+y &= (\bar{x}_1, \bar{x}_2, \dots, \bar{x}_k) + (\bar{y}_1, \bar{y}_2, \dots, \bar{y}_l) = \\ &= (\bar{x}_1, \bar{x}_2, \dots, \bar{x}_k, \bar{y}_1, \bar{y}_2, \dots, \bar{y}_l). \end{aligned}$$

Операция $+$ удовлетворяет условиям:

- (1) $\Gamma(x+y) = (\Gamma x + y) \cup (x + \Gamma y)$ (основной закон),
- (2) $x + y = y + x$ (коммутативность),
- (3) $x + (y+z) = (x+y) + z$ (ассоциативность),
- (4) $x+z = y+z \Rightarrow x = y$,
- (5) существует положение e (без спичек), такое, что $x+e=x$ для всех x .

Вершина z называется *неприводимой*, если невозможно написать $z=x+y$, $x \neq e$, $y \neq e$. Неприводимым вершинам здесь отвечают конфигурации вида $z = (\bar{x}_i)$ с одной единственной кучкой. Имеем

$$\begin{aligned} g(x) &= g(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_k) = g((\bar{x}_1) + (\bar{x}_2) + \dots + (\bar{x}_k)) = \\ &= g(\bar{x}_1) + g(\bar{x}_2) + \dots + g(\bar{x}_k). \end{aligned}$$

Поэтому достаточно располагать значениями g для неприводимых положений, чтобы сразу, простым d -сложением, найти остальные значения. Непосредственно получаем

$x_1 =$	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
$g(x) =$	0	0	1	0	2	1	0	2	1	0	2	1	3	2	1	3	2	4	3	0

В более общем случае прогрессивно конечной игры, в которой существует единственная такая вершина e , что $\Gamma e = \emptyset$, а операция $+$ удовлетворяет условиям (1) (2) (3) (4) (5), Адамс и Венсон доказали,

что *каждое положение разлагается в конечную сумму неприводимых положений, причем это разложение единственно.*

6.3.2. Общее определение игры (с полной информацией)

Общее определение игры весьма простое для ограниченных партий, требует большой осторожности в случае отказа от этого условия. *Игрок* — это лицо, с которым связан один из эпитетов: *активный* или *пассивный*, для двух игроков (*A*) и (*B*) игра определяется

- 1° множеством X , элементы которого называются *положениями игры*;
- 2° многозначным отображением Γ множества X в себя, называемым *правилом игры*;
- 3° однозначным отображением θ множества A в множество $\{0, 1, 2\}$, называемым *ходом*. Предполагается, что $\theta(x)=0$ тогда и только тогда, когда $\Gamma x = \emptyset$, при $\theta(x) = 1$ говорят, что в положении x *очередной ход должен сделать игрок (A)*, а при $\theta(x) = 2$ — что *ход игрока (B)*.
- 4° двумя вещественными функциями $f(x)$ и $g(x)$, определенными и ограниченными на X , называемыми *функциями предпочтения* соответственно для игроков (*A*) и (*B*)

Партия осуществляется следующим образом: начальное положение $x_0 \in X$ выбирается по жребию, ести, например, $\theta(x_0) = 1$, то игрок (*A*), которому нужно сдтать ход, должен выбрать положение x_1 в множестве Γx_0 , в свою очередь игрок, который должен сделать следующий ход [т. е. (*A*), если $\theta(x_1)=1$, или (*B*), если $\theta(x_1) = 2$, выбирает положение x_2 в множестве Γx_1 , и так далее. Если один из игроков выбрал такое положение x_k , для которого $\Gamma x_k = \emptyset$, то партия окончена (разумеется, партия может и никогда не окончиться). Если $S \subset X$ —множество положений, встретившихся в течение партии, то выигрыш игрока (*A*), по определению, есть

$$f(S) = \sup_{y \in S} f(y), \quad \text{если игрок (A) активный,}$$

$$f(S) = \inf_{y \in S} f(y), \quad \text{если игрок (A) пассивный,}$$

Аналогично, посредством функции g определяется выигрыш игрока (*B*) Цель игрока (*A*) состоит в получении возможно большего выигрыша. Обычно, $|f(S)|$ выражается денежной суммой, которую (*A*) должен получить в случае $f(S) \geq 0$ и уплатить в случае $f(S) \leq 0$.

Положим, $X_A = \{x/\theta(x) = 1\}$, $X_B = \{x/\theta(x) = 2\}$; надо отметить, что игра — это не что иное, как граф (X, Γ) , снабженный в каждой вершине x вектором $(\theta(x), f(x), g(x))$

Подграф графа (X, Γ) определяет *подигру*, а частичный граф — *частичную игру*. Заметим, что при вышеуказанных определениях, каждая подигра и каждая частичная игра сами являются играми (это неверно при других более обычных определениях). Игра представляет собой в точности „структуру“, заданную на абстрактном множестве X

ПРИМЕР 1. *Игра Ним*. В случае игры Ним на графе (X, Γ) *положение игры* есть пара (\bar{x}, i) , где $\bar{x} \in \bar{X}$, а i равно 1 или 2, смотря по тому, сделан последний ход игроком (B) или игроком (A) ; имеем

$$\theta(\bar{x}, i) = i, \quad \text{если } \bar{\Gamma} \bar{x} \neq \emptyset,$$

$$\theta(\bar{x}, i) = 0, \quad \text{если } \bar{\Gamma} \bar{x} = \emptyset,$$

$$\Gamma(\bar{x}, 1) = \bar{\Gamma} \bar{x} \times \{2\},$$

$$\Gamma(\bar{x}, 2) = \bar{\Gamma} \bar{x} \times \{1\}.$$

Если $\bar{\Gamma} \bar{x} = \emptyset$, то

$$f(\bar{x}, 2) = g(\bar{x}, 1) = 1.$$

Во всех остальных случаях $f(x) = g(x) = 0$ — оба игрока активны.

ПРИМЕР 2. *Шахматы*. Два игрока (A) и (B) играют соответственно „белыми“ и „черными“. *Положение* $x = (\bar{x}_1, \bar{x}_2, \bar{x}_3, \dots, i)$ представляется клетками $\bar{x}_1, \bar{x}_2, \bar{x}_3, \dots$ шахматной доски, на которых расположены различные фигуры, и индексом i , равным 1 или 2, — *теоретическим ходом* в положении x , $\bar{x} = (\bar{x}_1, \bar{x}_2, \bar{x}_3, \dots)$, называется так же *диаграммой* положения x .

Игрок (A) активен; $f(x) = 1$, когда x — матовое положение для черных, во всех же остальных случаях $f(x) = 0$.

ПРИМЕР 3. *Погоня*. На поверхности \bar{S} („море“) одна из двух лодок преследует другую. Эту ситуацию можно уподобить игре, в которой два игрока (A) и (B) ходят поочередно, например, каждую секунду, перемещая свои лодки. Тогда, \bar{x}_1 — точка моря \bar{S} , в которой находится преследующая лодка, а \bar{x}_2 — точка, где находится преследуемая лодка, и, если i — теоретический ход, то положением игры будет $x = (\bar{x}_1, \bar{x}_2, i)$. Пусть $\bar{B}_i(\bar{x}_i) \subset \bar{S}$ — „круг“ с центром \bar{x}_i , имеющий своим радиусом наибольшее расстояние, которое может проплыть лодка \bar{x}_i за одну секунду. Имеем

$$\Gamma(\bar{x}_1, \bar{x}_2, 1) = \bar{B}_1(\bar{x}_1) \times \{\bar{x}_2\} \times \{2\}$$

$$\Gamma(\bar{x}_1, \bar{x}_2, 2) = \{\bar{x}_1\} \times \bar{B}_2(\bar{x}_2) \times \{1\}.$$

Обозначим череч $d(\bar{x}, \bar{y})$ расстояние между точками \bar{x} и \bar{y} , если целью преследователя является возможно большее приближение к преследуемому, то игрок (А) — активный и

$$f(\bar{x}_1, \bar{x}_2, i) = -d(\bar{x}_1, \bar{x}_2).$$

Если цель преследователя состоит в том, чтобы обязательно настигнуть преследуемую, то (А) активен и

$$f(\bar{x}_1, \bar{x}_2, i) = 1 \quad \text{при} \quad \bar{x}_1 = \bar{x}_2,$$

$$f(\bar{x}_1, \bar{x}_2, i) = 0 \quad \text{при} \quad \bar{x}_1 \neq \bar{x}_2.$$

Если цель преследуемого — не дать себя настигнуть, то игрок (В) — пассивный и

$$g(\bar{x}_1, \bar{x}_2, i) = 1 \quad \text{при} \quad \bar{x}_1 = \bar{x}_2,$$

$$g(\bar{x}_1, \bar{x}_2, i) = 0 \quad \text{при} \quad \bar{x}_1 \neq \bar{x}_2.$$

В нашем случае, цель игрока (А) существенно *активна*, в то время, как как цель игрока (В) существенно *пассивна*. *Подигра* получится, если запретить некоторые области моря \bar{S} для одной из лодок. *Частичная игра* поучится, если уменьшить максимальную скорость какой либо из лодок.

6.3.3. Стратегии

В игре $(X, \Gamma, \theta, f, g)$ стратегия игрока (А) есть, по определению, такое однозначное отображение σ множества X_A в X , что

$$\sigma x \in \Gamma x \quad (x \in X_A)$$

Игрок (А) *придерживается* стратеги σ , если он а priori решает " в каждом положении x , в котором ход мой, я буду в течение всей партии выбирать положение σx ". Множество всех возможных стратегий σ игрока (А) обозначается символом Σ_A .

Если начальное положение x_0 фиксировано и если игроки (А) и (В) придерживаются соответственно стратегий σ и τ , то партия полностью определена и мы обозначим через $\langle x_0, \sigma, \tau \rangle$ множество встречающихся в ней положений. Выигрыш игрока (А) в таком случае равен

$$f(x_0, \sigma, \tau) = \sup \{f(x) / x \in \langle x_0, \sigma, \tau \rangle\}, \quad \text{если игрок (А) активен,}$$

$$f(x_0, \sigma, \tau) = \inf \{f(x) / x \in \langle x_0, \sigma, \tau \rangle\}, \quad \text{если игрок (А) пассивен.}$$

Пара (σ, τ_0) , по определению, есть *равновесие* (для положения x), если

$$f(x, \sigma, \tau_0) \leq f(x, \sigma_0, \tau_0), \quad (\sigma \in \Sigma_A),$$

$$g(x, \sigma_0, \tau) \leq g(x, \sigma_0, \tau_0), \quad (\tau \in \Sigma_B).$$

Иначе говоря, если перед началом партии игроки договорятся придерживаться стратегий σ_0 и τ_0 , то такое решение будет обладать определенной устойчивостью. Игрок, который попытается изменить

свою стратегию в ходе партии будет наказан за это уменьшением выигрыша.

Пара (σ_0, τ_0) , являющаяся равновесием для любого положения x , называется *абсолютным равновесием* игры, обобщая теорему существования ядра у прогрессивно конечного графа, получаем следующий основной результат

Теорема Цермело — фон Неймана. *Если граф (X, Γ) некоторой игры прогрессивно конечен, а множества $f(X) = \{f(x) / x \in X\}$ и $g(X)$ конечны, то игра допускает абсолютное равновесие (σ_0, τ_0) .*

Как и ранее, последовательно определяем множества $X(\alpha)$, полагая

$$X(0) = \{x / \Gamma x = \emptyset\};$$

$X(\alpha) = \{x / \Gamma x \subset X(\alpha - 1)\}$, если α — порядковое число первого рода;

$$X(\alpha) = \bigcup_{\beta < \alpha} X(\beta) \text{ если } \alpha \text{ — порядковое число второго рода.}$$

Так как граф прогрессивно конечен, то в силу известной теоремы, существует порядковое число γ такое, что $X = X(\gamma)$. Будем последовательно определять равновесие (σ_0, τ_0) для всех $X(\alpha)$.

В $X(0)$ отображение σ_0 не обязано быть определено, однако мы положим

$$\sigma_0 x = x \quad (x \in X(0))$$

Если α — порядковое число первого рода, а равновесие (σ_0, τ_0) уже определено в $X(\alpha - 1)$, то определяем равновесие $(\bar{\sigma}_0, \bar{\tau}_0)$ в $X(\alpha)$ следующим образом. Пусть $x \in X_A$, тогда

$$1^\circ \text{ в случае } x \in X(\alpha - 1) \text{ полагаем } \bar{\sigma}_0 x = \sigma_0 x,$$

2° в случае $x \in X(\alpha) \setminus X(\alpha - 1)$ полагаем $\sigma_0 x = y$, где $y \in \Gamma x$, и

$$f(y, \sigma_0, \tau_0) = \max_{z \in \Gamma x} f(z, \sigma_0, \tau_0).$$

Очевидно, (σ_0, τ_0) есть равновесие для подигры, порожденной множеством $X(0)$. Допустим теперь, что (σ_0, τ_0) — равновесие для подигры, порожденной множеством $X(\alpha - 1)$, и покажем, что тогда $(\bar{\sigma}_0, \bar{\tau}_0)$ является равновесием для подигры, порожденной $X(\alpha)$, т. е.

$$(1) \quad f(x, \bar{\sigma}_0, \bar{\tau}_0) \leq f(x, \bar{\sigma}_0, \bar{\tau}_0) \quad (\bar{\sigma} \in \sum_A, x \in X(\alpha))$$

$$(1') \quad g(x, \bar{\sigma}_0, \bar{\tau}_0) \leq g(x, \bar{\sigma}_0, \bar{\tau}_0) \quad (\bar{\tau} \in \sum_B, x \in X(\alpha))$$

Так как предполагается, что неравенства (1) и (1') уже выполнены для $x \in X(\alpha - 1)$, то можно считать, что $x \in X(\alpha) \setminus X(\alpha - 1)$

Если $x \in X_A$, то пусть $\bar{\sigma}_0 x = z$, ввиду того, что $z \in X(\alpha - 1)$, имеем

$$f(z, \sigma_0, \tau_0) \leq f(z, \sigma_0, \tau_0) \leq f(y, \sigma_0, \tau_0)$$

Будем выводить неравенство (1) во всех возможных случаях

1° если игрок (А) активен, а $f(x) \leq f(z, \sigma, \tau_0)$, то
 $f(x, \bar{\sigma}_0, \bar{\tau}_0) = f(z, \sigma, \tau_0) \leq f(y, \sigma_0, \tau_0) \leq f(x, \bar{\sigma}_0, \bar{\tau}_0)$

2° если игрок (А) активен, а $f(x) \geq f(z, \sigma, \tau_0)$, то
 $f(x, \bar{\sigma}_0, \bar{\tau}_0) = f(x) \leq f(x, \bar{\sigma}_0, \bar{\tau}_0)$

3° если игрок (А) пассивен, а $f(x) \leq f(y, \sigma_0, \tau_0)$, то
 $f(x, \bar{\sigma}_0, \bar{\tau}_0) \leq f(x) = f(x, \bar{\sigma}_0, \bar{\tau}_0)$

4° если игрок (А) пассивен, а $f(x) \geq f(y, \sigma_0, \tau_0)$, то
 $f(x, \bar{\sigma}_0, \bar{\tau}_0) \leq f(z, \sigma, \tau_0) \leq f(y, \sigma_0, \tau_0) = f(x, \bar{\sigma}_0, \bar{\tau}_0)$

Таким же путем, заменив f на g , докажем неравенство (1')

Следствие 1. Если граф (X, Γ) некоторой игры прогрессивно конечен, то для любого $\varepsilon > 0$ существует такая пара (σ_0, τ_0) , что

$$f(x, \sigma, \tau_0) \leq f(x, \sigma_0, \tau_0) + \varepsilon \quad (x \in X, \sigma \in \Sigma_A)$$

$$g(x, \sigma_0, \tau) \leq g(x, \sigma_0, \tau_0) + \varepsilon \quad (x \in X, \tau \in \Sigma_B)$$

В самом деле, заменим функции предпочтения f и g такими функциями f' и g' , чтобы множества

$$f'(X) = \{ f'(x) / x \in X \}$$

$$g'(X) = \{ g'(x) / x \in X \}$$

были конечными и чтобы было

$$|f'(x) - f(x)| \leq \frac{\varepsilon}{2} \quad (x \in X)$$

$$|g'(x) - g(x)| \leq \frac{\varepsilon}{2} \quad (x \in X)$$

Это всегда возможно, поскольку f и g предполагаются ограниченными на X . Так как существует равновесие с f' и g' , то будет существовать и ε -равновесие с f и g

Следствие 2. Если граф (X, Γ) некоторой игры прогрессивно конечен, то для любого $\varepsilon > 0$ существуют такие стратегии σ_0 и τ_0 , что

$$f(x_0, \sigma, \tau_0) - \varepsilon \leq f(x_0, \sigma_0, \tau_0) \leq f(x_0, \sigma_0, \tau) + \varepsilon$$

(каковы бы ни были $\sigma \in \Sigma_A, \tau \in \Sigma_B$).

Иными словами, в партии с начальным положением x_0 игрок (А) посредством стратегии σ_0 может гарантировать выигрыш $f(x_0, \sigma_0, \tau_0)$ (с точностью ε), с другой стороны, в силу первого неравенства, никакая другая стратегия σ не даст ему возможности гарантировать больший выигрыш (с точностью ε). Поэтому σ_0 будет *благоприятной (оптимальной) стратегией* для (А).

Чтобы доказать это следствие достаточно положить $f(x) = -g(x)$ и рассматривать игрока (А) как активного, а (В) — как пассивного, тогда

$$f(x_0, \sigma, \tau) = -g(x_0, \sigma, \tau)$$

и высказанное утверждение получается сразу.

Каждый раз, когда выигрыши обоих игроков одинаковы по абсолютной величине и противоположны по знаку, говорят, что игра представляет собой *поединок*. Значение

$$f(\sigma, \tau) = -g(\sigma, \tau)$$

называется *результатом* игры. (A) стремится получить как можно больший результат, а (B) — как можно меньший. Наибольший результат, который (A) может гарантировать посредством стратегии, есть

$$\alpha_0 = \sup_{\sigma} \inf_{\tau} f(\sigma, \tau).$$

Наименьший результат, который (B) может гарантировать посредством стратегии, есть

$$\beta_0 = \inf_{\tau} \sup_{\sigma} f(\sigma, \tau).$$

Заметим что $\alpha_0 \leq \beta_0$, ибо если каждый из игроков (A) и (B) будет придерживаться благоприятной стратегии, то результат окажется не меньше α_0 и не больше α_0 . Но справедливо и значительно более сильное утверждение

Теорема 5. *Если в поединке граф (X, Г) прогрессивно конечен, то $\alpha_0 = \beta_0$.*

В самом деле, ввиду следствия 2

$$\begin{aligned} \sup_{\sigma} f(\sigma, \tau_0) &\leq f(\sigma_0, \tau_0) + \varepsilon \\ \inf_{\tau} f(\sigma_0, \tau) &\geq f(\sigma_0, \tau_0) - \varepsilon \end{aligned}$$

откуда

$$\beta_0 \leq f(\sigma_0, \tau_0) + \varepsilon \leq \alpha_0 + 2\varepsilon.$$

Поскольку ε произвольно, $\beta_0 \leq \alpha_0$, а так как противоположное неравенство выполняется всегда, то $\beta_0 = \alpha_0$.

ПРИМЕР 1. Следствий из этой теоремы очень много. Для прогрессивно конечной игры Ним, где результат может быть только +1 или —1, получается уже известное нам утверждение: при данном начальном положении x_0 либо существует стратегия для (A), позволяющая ему наверняка выиграть ($\alpha_0 = +1$), либо существует стратегия для (B) обеспечивающая выигрыш ему ($\alpha_0 = -1$)

ПРИМЕР 2. Рассмотрим следующую игру: (A) и (B) ходят по очереди, передвигая жетон по ребру графа, изображенного на рис. 3. Если по окончании партии окажется, что жетон прошел четное число раз через вершину, отмеченную кружком (a или b) то (A) платит

до того, как началось производство продукта p_j), в зависимости от комбинации (p_i, p_j) . Стоимость перенастройки аппаратуры постоянна и не зависит от продукта, который только что произведен, или от продукта, следующего за ним. Разумеется, не требуется никаких затрат, если перенастройка аппаратуры не нужна. (Сформулированная выше задача может возникать в следующих двух случаях. Или стоимость перенастройки действительно не зависит от продукта, или, другая возможность, стоимость в каждом случае неизвестна и в качестве приближения берется средняя постоянная стоимость.) Предположим, что эти продукты производятся в непрерывном цикле, так что после производства последнего из n продуктов снова возобновляется в том же фиксированном цикле производство первого продукта.

Возникает вопрос о том, может ли быть найдена циклическая последовательность производства продуктов p_i ($i = 1, 2, \dots, n$), не требующая перенастройки аппаратуры. Пусть G — ориентированный граф, вершины которого представляют продукты, а существование дуги (x_i, x_j) означает, что продукт p_j может следовать за продуктом p_i без перенастройки аппаратуры. Тогда ответ на поставленный вопрос зависит от того, имеет ли этот орграф гамильтонов цикл (точнее, гамильтонов контур) или нет. (Гамильтонов цикл в орграфе — это ориентированный цикл (контур), проходящий ровно один раз через каждую вершину графа)

Если не существует циклической последовательности продуктов, не требующей перенастройки аппаратуры, то какова должна быть последовательность производства с наименьшими затратами на перенастройку, т. е. требующая наименьшего числа необходимых перенастроек? Ответ на этот вопрос может быть получен с помощью итеративного применения алгоритма нахождения гамильтонова цикла в орграфе. Как именно это может быть сделано, обсуждается позже в этой главе.

Таким образом, метод, позволяющий ответить на вопрос, содержит ли какой-либо орграф гамильтонов цикл, имеет прямые приложения в задачах упорядочения или планирования операций. В равной степени важно использование такого метода в качестве основного шага в алгоритмах решения других, на первый взгляд далеких от данной тематики, задач теории графов.

В этой главе рассматриваются следующие две задачи.

Задача 1. Дан ориентированный граф G , требуется найти в G гамильтонов цикл (или все циклы), если существует хотя бы один такой цикл.

Задача ii. Дан *полный* орграф G , дугам которого приписаны произвольные веса $C = [c_{ij}]$, найти такой гамильтонов цикл (цепь), который имеет наименьший общий вес. Задача нахождения гамильтонова цикла с наименьшим весом хорошо известна в литературе как *задача коммивояжера*. Следует отметить, что если орграф G не полный, то его можно рассматривать как полный орграф, приписывая отсутствующим дугам бесконечный вес.

Алгоритмы решения задачи коммивояжера и ее вариантов имеют большое число практических приложений в различных областях человеческой деятельности. Рассмотрим, например, задачу, в которой грузовик выезжает с центральной базы для доставки товаров данному числу потребителей и возвращается назад на базу. Стоимость перевозки пропорциональна пройденному грузовиком расстоянию, и при заданной матрице расстояний между потребителями маршрут с наименьшими транспортными затратами получается как решение соответствующей задачи коммивояжера. Аналогичные типы задач возникают при сборе почтовых отправлений из почтовых ящиков, составлении графика движения школьных автобусов по заданным остановкам и т. д. Задача очень легко обобщается и на тот случай, когда доставкой (сбором) занимаются несколько грузовиков, хотя эту задачу можно также переформулировать как задачу коммивояжера большей размерности. Другие приложения включают составление расписания выполнения операций на машинах, проектирование электрических сетей, управление автоматическими линиями и т. д. Вполне очевидно, что сформулированная выше задача (i) является частным случаем задачи (ii). В самом деле, приписывая случайным образом дугам заданного орграфа G конечные веса, получаем задачу коммивояжера. (Здесь, как обычно, предполагается, что орграф G «расширен» до некоторого полного орграфа G' и дугам, отсутствующим в G , но принадлежащим полному орграфу G' , приписаны бесконечные веса.)

Если решение для этой задачи, т. е. кратчайший гамильтонов цикл (так (для краткости) называется гамильтонов цикл с наименьшим общим весом), имеет конечное значение, то это решение является гамильтоновым циклом орграфа G (т. е. ответом на задачу i). Если же решение имеет бесконечное значение, то G не имеет гамильтонова цикла. С другой стороны можно дать еще одну интерпретацию задачи i). Рассмотрим снова *полный* орграф G_1 с общей матрицей весов дуг $[c_{ij}]$ и рассмотрим задачу нахождения такого гамильтонова цикла, в котором *самая длинная* дуга (то есть дуга с наибольшим весом) минимальна. Эту задачу можно назвать *минимаксной задачей коммивояжера*, оттеняя ее «минимаксную природу» (по сравнению с

классической задачей коммивояжера), которую в той же терминологии можно было бы назвать *минусуммной* задачей. Покажем теперь, что задача (i) действительно эквивалентна минимаксной задаче коммивояжера.

В вышеупомянутом полном орграфе G_1 мы можем наверняка найти гамильтонов цикл. Пусть это будет цикл Φ_1 , и пусть вес самой длинной его дуги равен ζ_1 . Удалив из G_1 любую дугу, вес которой не меньше ζ_1 , получим орграф G_2 . Найдем в орграфе G_2 гамильтонов цикл Φ_2 , и пусть вес его самой длинной дуги равен ζ_2 . Удалим из G_2 любую дугу, вес которой не меньше ζ_2 , и так будем продолжать до тех пор, пока не получим орграф G_{m+1} , не содержащий никакого гамильтонова цикла. Гамильтонов цикл Φ_m в G_m (с весом ζ_m) является тогда по определению решением минимаксной задачи коммивояжера, так как из отсутствия гамильтонова цикла в G_{m+1} следует, что в G_1 не существует никакого гамильтонова цикла, не использующего по крайней мере одну дугу с весом, большим или равным ζ_m . Таким образом, алгоритм нахождения гамильтонова цикла в орграфе решает также минимаксную задачу коммивояжера. Наоборот, если мы располагаем алгоритмом решения последней задачи, то гамильтонов цикл в произвольном орграфе G может быть найден с помощью построения полного орграфа G_1 с тем же самым множеством вершин, что и в G , дугам которого, соответствующим дугам из G , приписаны единичные веса, а остальным дугам — бесконечные веса. Если решение минимаксной задачи коммивояжера для G_1 имеет конечный вес (на самом деле равный единице), то в графе G может быть найден соответствующий гамильтонов цикл. Если же решение имеет бесконечный вес, то в графе G не существует никакого гамильтонова цикла. Следовательно, две указанные задачи можно рассматривать как эквивалентные, поскольку было продемонстрировано, что алгоритм нахождения гамильтонова цикла позволяет решать минимаксную задачу коммивояжера и наоборот.

Ввиду того что обе сформулированные выше задачи (i) и (ii) часто встречаются в практических ситуациях и (как мы увидим позже) задачу (i) саму по себе решить намного проще, чем как подзадачу задачи (ii), мы обе эти задачи рассмотрим отдельно в частях настоящей главы.

7.2. Гамильтоновы циклы в графе

Пока неизвестно никакого простого критерия или алгебраического метода, позволяющего ответить на вопрос, существует или нет в произвольном графе G гамильтонов цикл. Критерии существования, данные в работах Поша [1], Нэша-Уильямса и Оре, представляют теоретический интерес, но являются слишком общими и не пригодны для произвольных графов, встречающихся на практике. Алгебраические методы определения гамильтоновых циклов не могут быть применены к задачам с более чем несколькими десятками вершин, так как они требуют слишком большого времени работы и большой памяти компьютера. Более приемлемым является способ Робертса и Флореса, который не предъявляет чрезмерных требований к памяти компьютера, но время в котором зависит экспоненциально от числа вершин в графе. Однако другой неявный метод перебора имеет для большинства типов графов очень небольшой показатель роста времени вычислений в зависимости от числа вершин. Он может быть использован для нахождения гамильтоновых циклов в очень больших графах. В настоящей главе мы опишем алгебраический метод и два способа перебора.

7. 2.1. Алгебраический метод

Этот метод основан на работе Йоу, Даниэльсона и Дхавана и включает в себя построение всех простых цепей с помощью последовательного перемножения матриц.

«Внутреннее произведение вершин» цепи $x_1, x_2, x_3, \dots, x_{k-1}, x_k$ определяется как выражение вида $x_2 \cdot x_3 \cdot \dots \cdot x_{k+1}$, не содержащее две концевые вершины x_1 и x_k . «Модифицированная матрица смежности» $\mathbf{B} = [\beta(i, j)]$ — это $(n \times n)$ -матрица, в которой $\beta(i, j) = x_j$, если существует дуга из x_i в x_j и нуль в противном случае. Предположим теперь, что у нас есть матрица $\mathbf{P}_l = [p_l(i, j)]$, где $p_l(i, j)$ — сумма внутренних произведений всех простых цепей длины l ($l \geq 1$) между вершинами x_i и x_j для $x_i \neq x_j$. Положим $p_l(i, i) = 0$ для всех i . Обычное алгебраическое произведение матриц $\mathbf{B} \cdot \mathbf{P}_l = \mathbf{P}'_{l+1} = [p'_{l+1}(s, t)]$ определяется как

$$p'_{l+1}(s, t) = \sum_k \beta(s, k) \cdot p_l(k, t), \quad (1)$$

т. е. $p'_{l+1}(s, t)$ является суммой внутренних произведений всех цепей из x_s в x_t длины $l + 1$. Так как все цепи из x_k в x_p представленные внутренними произведениями из $p_l(k, t)$, являются простыми, то

среди цепей, получающихся из выражения (1), не являются простыми лишь те, внутренние произведения которых в $p_l(k, t)$ содержат вершину x_s . Таким образом, если из $p'_{l+1}(s, t)$ исключить все слагаемые, содержащие x_s (а это можно сделать простой проверкой), то получим $p_{l+1}(s, t)$. Матрица $\mathbf{P}_{l+1} = [p_{l+1}(s, t)]$, все диагональные элементы которой равны 0, является тогда матрицей всех простых цепей длины $l + 1$.

Вычисляя затем $\mathbf{B} \cdot \mathbf{P}_{l+1}$, находим \mathbf{P}_{l+2} и т. д., пока не будет построена матрица \mathbf{P}_{n+1} , дающая все гамильтоновы цепи (имеющие длину $n - 1$) между всеми парами вершин. Гамильтоновы циклы получаются тогда сразу из цепей в \mathbf{P}_{n-1} и тех дуг из G , которые соединяют начальную и конечную вершины каждой цепи. С другой стороны, гамильтоновы циклы даются членами внутреннего произведения вершин, стоящими в любой диагональной ячейке матрицы $\mathbf{B} \cdot \mathbf{P}_{n-1}$ (все диагональные элементы этой матрицы одинаковые)

Очевидно, что в качестве начального значения матрицы \mathbf{P} (т. е. \mathbf{P}_1) следует взять матрицу смежности \mathbf{A} графа, положив все ее диагональные элементы равными нулю.

1. Пример. Рассмотрим граф, изображенный на рис. 1,

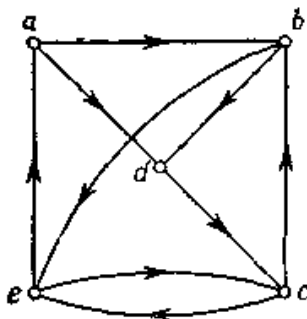


Рис. 1 Граф из примера 1.

матрица смежности которого равна

$$\mathbf{A} = \begin{array}{c} \\ \\ \\ \\ \\ \end{array} \begin{array}{ccccc} & a & b & c & d & e \\ a & 0 & 1 & 0 & 1 & 0 \\ b & 0 & 0 & 0 & 1 & 1 \\ c & 0 & 1 & 0 & 0 & 1 \\ d & 0 & 0 & 1 & 0 & 0 \\ e & 1 & 0 & 1 & 0 & 0 \end{array}$$

и модифицированная матрица смежности

$$\mathbf{B} = \begin{array}{c} \\ \\ \\ \\ \\ \end{array} \begin{array}{ccccc} & a & b & c & d & e \\ a & 0 & b & 0 & d & 0 \\ b & 0 & 0 & 0 & d & e \\ c & 0 & b & 0 & 0 & e \\ d & 0 & 0 & c & 0 & 0 \\ e & a & 0 & c & 0 & 0 \end{array}$$

Положим $\mathbf{P}_1 \equiv \mathbf{A}$. Матрица $\mathbf{P}'_2 = \mathbf{B} \cdot \mathbf{P}'_1$ получается равной

$$\mathbf{P}'_2 = \begin{array}{c} \\ \\ \\ \\ \\ \end{array} \begin{array}{ccccc} & a & b & c & d & e \\ a & 0 & 0 & d & b & b \\ b & e & 0 & d+e & 0 & 0 \\ c & e & 0 & \underline{e} & b & b \\ d & 0 & c & 0 & 0 & c \\ e & 0 & a+c & 0 & a & \underline{e} \end{array}$$

Матрица \mathbf{P}_2 почти такая же, как \mathbf{P}'_2 , — только подчеркнутые элементы в \mathbf{P}'_2 надо заменить нулями. Матрица $\mathbf{P}_3 = \mathbf{B} \cdot \mathbf{P}_2$ равна

$$\mathbf{P}'_3 = \begin{array}{c} \\ \\ \\ \\ \\ \end{array} \begin{array}{ccccc} & a & b & c & d & e \\ a & \underline{be} & dc & bd+be & 0 & dc \\ b & 0 & \underline{dc+ea+ec} & 0 & ea & dc \\ c & be & ea+\underline{ec} & \underline{bd+be} & ea & 0 \\ d & ce & 0 & 0 & \underline{cb} & cb \\ e & \underline{ce} & 0 & ad & ab+cb & \underline{ab+cb} \end{array}$$

\mathbf{P}_3 получается из \mathbf{P}'_3 после замены подчеркнутых элементов нулями:

	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>
<i>a</i>	<u>dce</u>	0	0	<u>bea</u>	<i>bdc + bdc</i>
<i>b</i>	<i>dce</i>	0	<i>ead</i>	<u><i>eab + ecb</i></u>	<u><i>dc</i></u> <i>b</i>
$\mathbf{P}_4 =$ <i>c</i>	0	0	<u><i>ead</i></u>	<i>bea + eab</i> + <i>ecb</i>	<u><i>bdc</i></u>
<i>d</i>	<i>cbe</i>	<i>cea</i>	0	<u><i>cea</i></u>	0
<i>e</i>	<u><i>cbe</i></u>	<i>adc + cea</i>	<i>abd + abe</i>	<u><i>cea</i></u>	<u><i>adc</i></u>

Матрица \mathbf{P}_4 гамильтоновых цепей равна

	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>
<i>a</i>	0	0	0	0	<i>bdc + dcb</i>
<i>b</i>	<i>dce</i>	0	<i>ead</i>	0	0
$\mathbf{P}_4 =$ <i>c</i>	0	0	0	<i>bea + eab</i>	0
<i>d</i>	<i>cbe</i>	<i>cea</i>	0	0	0
<i>e</i>	0	<i>adc</i>	<i>abd</i>	0	0

Гамильтоновы цепи *abdce* и *adcbe*, соответствующие элементу (1,4) вышенаписанной матрицы, дают гамильтоновы циклы *abdcea* и *adcbea*, если добавить замыкающую дугу (*e, a*). Все другие гамильтоновы цепи в \mathbf{P}_4 приводят к тем же самым двум гамильтоновым циклам, и поэтому в графе *G* существует только два таких цикла.

Недостатки этого метода совершенно очевидны. В процессе умножения матриц (т. е. когда *l* увеличивается) каждый элемент матрицы \mathbf{P}_l будет состоять из все большего числа членов вплоть до некоторого критического значения *l*, после которого число членов снова начнет уменьшаться. Это происходит вследствие того, что для малых значений *l* и для графов, обычно встречающихся на практике, число цепей длины *l + 1*, как правило, больше, чем число цепей длины *l*, а для больших значений *l* имеет место обратная картина. Кроме того, так как длина каждого члена внутреннего произведения вершин увеличивается на единицу, когда *l* увеличивается на единицу, то объем

памяти, необходимый для хранения матрицы \mathbf{P}_l , растет очень быстро вплоть до максимума при некотором критическом значении l , после которого этот объем снова начинает уменьшаться.

Небольшая модификация вышеприведенного метода позволяет во много раз уменьшить необходимый объем памяти и время вычислений. Так как нас интересуют только гамильтоновы циклы и, как было отмечено выше, они могут быть получены из членов внутреннего произведения любой диагональной ячейки матрицы $\mathbf{B} \cdot \mathbf{P}_{n-1}$, то необходимо знать только элемент $p_{n-1}(1,1)$. При этом на каждом этапе не обязательно вычислять и хранить всю матрицу \mathbf{P}_l , достаточно лишь найти первый столбец из \mathbf{P}_l . Эта модификация уменьшает необходимый объем памяти и время вычислений в n раз.

7. 2.2. Метод перебора Робертса и Флореса

В противоположность алгебраическим методам, с помощью которых пытаются найти сразу все гамильтоновы циклы и при реализации которых приходится хранить поэтому все цепи, которые могут оказаться частями таких циклов, метод перебора имеет дело с одной цепью, непрерывно продлеваемой вплоть до момента, когда либо получается гамильтонов цикл, либо становится ясно, что эта цепь не может привести к гамильтонову циклу. Тогда цепь модифицируется некоторым систематическим способом (который гарантирует, что в конце концов будут исчерпаны все возможности), после чего продолжается поиск гамильтонова цикла. В этом способе для поиска требуется очень небольшой объем памяти и за один раз находится один гамильтонов цикл.

Следующая схема перебора, использующая обычную технику возвращения, была первоначально предложена Робертсом и Флореом. Начинают с построения $(k \times n)$ -матрицы $\mathbf{M} = [m_{ij}]$, где элемент m_{ij} есть i -я вершина (скажем x_q), для которой в графе $G = (X, \Gamma)$ существует дуга (x_p, x_q) . Вершины x_q в множестве $\Gamma(x_i)$ можно упорядочить произвольно, образовав элементы j -го столбца матрицы \mathbf{M} . Число строк k матрицы \mathbf{M} будет равно наибольшей полустепени исхода вершины.

Метод состоит в следующем. Некоторая начальная вершина (скажем, x_1) выбирается в качестве отправной и образует первый элемент множества S , которое каждый раз будет хранить уже найденные вершины строящейся цепи. К S добавляется первая вершина (например, вершина a) в столбце x_1 . Затем к множеству S добавляется первая возможная вершина (например, вершина b) в столбце a , потом добавляется к S первая возможная вершина (например, вершина c) в

столбце b и т. д. Под «возможной» вершиной мы понимаем вершину, еще не принадлежащую S . Существуют две причины, препятствующие включению некоторой вершины на шаге r в множество $S = \{x_1, a, b, c, \dots, x_{r-1}, x_r\}$. Или (1) в столбце x_r нет возможной вершины, или (2) цепь, определяемая последовательностью вершин в S , имеет длину $n - 1$, т. е. является гамильтоновой цепью.

В случае (2)

(а) в графе G существует дуга (x_r, x_1) и поэтому найден гамильтонов цикл, или

(б) дуга (x_r, x_1) не существует и не может быть получен никакой гамильтонов цикл.

В случаях (1) и (2б) следует прибегнуть к *возвращению*, в то время как в случае (2а) можно прекратить поиск и напечатать результат (если требуется найти только один гамильтонов цикл, или (если нужны все такие циклы) произвести печать и прибегнуть к возвращению. Возвращение состоит в удалении последней включенной вершины x_r из S , после чего остается множество $S = \{x_1, a, b, c, \dots, x_{r-1}\}$, и добавлении к S первой возможной вершины, следующей за x_r , в столбце x_{r-1} матрицы \mathbf{M} . Если не существует никакой возможной вершины, делается следующий шаг возвращения и т. д.

Поиск заканчивается в том случае, когда множество S состоит только из вершины x_1 и не существует никакой возможной вершины, которую можно добавить к S , так что шаг возвращения делает множество S пустым. Гамильтоновы циклы, найденные к этому моменту, являются тогда всеми гамильтоновыми циклами, существующими в графе.

Пример. Рассмотрим граф, изображенный на рис. 2.

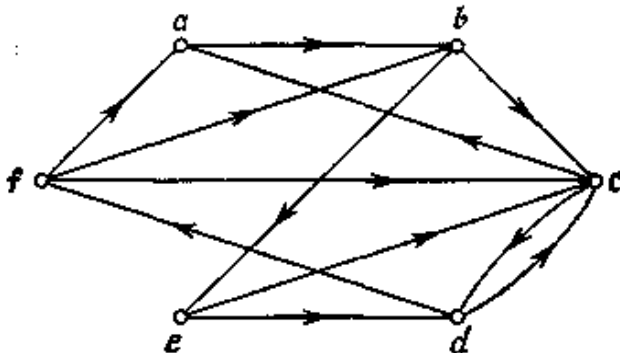


Рис. 2. Граф из примера.

Матрица M приводится ниже, вершины в каждом столбце расположены в алфавитном порядке:

	<i>a</i>	<i>b</i>	<i>c</i>	<i>d</i>	<i>e</i>	<i>f</i>
1	<i>b</i>	<i>c</i>	<i>a</i>	<i>c</i>	<i>c</i>	<i>a</i>
$M = 2$	—	<i>e</i>	<i>d</i>	<i>f</i>	<i>d</i>	<i>b</i>
3	—	—	—	—	—	—

Поиск всех гамильтоновых циклов производится так (вершина a берется в качестве отправной вершины):

<i>Множество S</i>	<i>Комментарии</i>
1. a	Добавляем первую возможную вершину в столбце a (т. е. вершину b).
→ 2. a, b	Добавляем первую возможную вершину в столбце b (т. е. вершину c).
3. a, b, c	Первая вершина (a) в столбце c не является возможной ($a \notin S$), добавляем следующую вершину в столбце (т. е. вершину d).
4. a, b, c, d	Добавляем вершину f .
5. a, b, c, d, f	В столбце f нет возможной вершины, возвращение.
6. a, b, c, d	В столбце d не существует возможной вершины, следующей за f . Возвращение.
7. a, b, c	Аналогично предыдущему. Возвращение.
8. a, b	Добавляем вершину e .
→ 9. a, b, e	Добавляем вершину c .
10. a, b, e, c	Добавляем вершину d
11. a, b, e, c, d	Добавляем вершину f .
12. <u>a, b, e, c, d, f</u>	Гамильтонова цепь. Дуга ($f a$) дает гамильтонов цикл. Возвращение.
13. a, b, e, c, d	Возвращение
14. a, b, e, c	Возвращение.
15. a, b, e	Добавляем вершину d .
16. a, b, e, d	Добавляем вершину f .
17. a, b, e, d, f	Добавляем вершину c .
18. <u>a, b, e, d, f, c</u>	Гамильтонова цепь. Цепь замыкается дугой (c, a). Возвращение.
19. a, b, e, d, f	Возвращение.
20. a, b, e, d	Возвращение.
21. a, b, e	Возвращение.
22. $a, b,$	Возвращение.
23. a	Возвращение.
24. \emptyset	Конец поиска.

Улучшение основного метода.

Допустим, что на некотором этапе поиска построенная цепь задается множеством $S = \{x_1, x_2, \dots, x_r\}$ и что следующей вершиной, которую предполагается добавить к S , является $x^* \notin S$. Рассмотрим теперь две следующие ситуации, в которых вершина является изолированной в подграфе, остающемся после удаления из $G = (X, \Gamma)$ всех вершин, образующих построенную до этого цепь.

(а) Если существует такая вершина $x \in X - S$, что $x \in \Gamma(x_r)$ и $\Gamma^{-1}(x) \subseteq S$ (см. рис. 3а), то, добавляя к S любую вершину x^* , отличную от x , мы не сможем в последующем достигнуть вершины x ни из какой конечной вершины построенной цепи, и, значит, эта цепь не сможет привести нас к построению гамильтонова цикла.

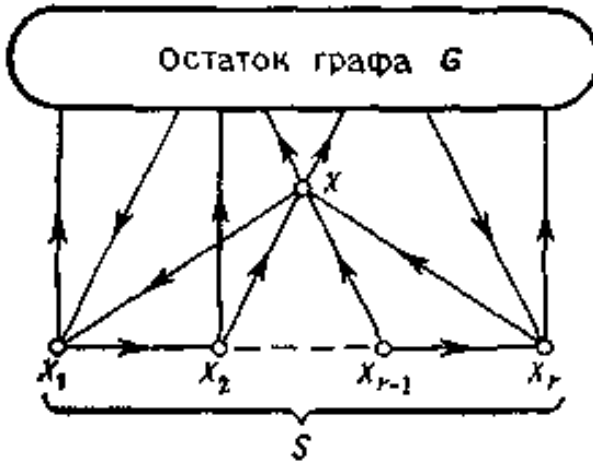


Рис. 3а. ($x \in \Gamma(x_r)$) и $\Gamma^{-1}(x) \subseteq S$. Следующей дугой должна быть (x_r, x) .

Таким образом, в этом случае x является единственной вершиной, которую можно добавить к S для продолжения цепи.

(б) Если существует такая вершина $x \in X - S$, что $x \notin \Gamma^{-1}(x_1)$ и $\Gamma(x) \subset S \cup \{x^*\}$ для некоторой другой вершины x^* , то x^* не может быть добавлена к S , так как тогда в остающемся подграфе не может существовать никакой цепи между x и x_r . Цепь, определяемая множеством $S \cup \{x^*\}$, не может поэтому привести к гамильтонову циклу, а в качестве кандидата на добавление к множеству S следует рассмотреть другую вершину, отличную от x^* (см. рис. 3 б).

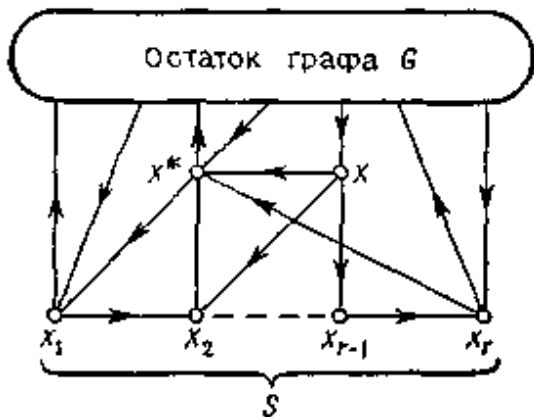


Рис. 3б. ($x \notin \Gamma^{-1}(x_1)$ и $\Gamma(x) \subset S \cup \{x^*\}$). Следующей дугой не должна быть (x, x^*) .

В только что приведенном примере ситуация (а) возникает на шаге 2, когда множество S есть $\{a, b\}$. Если заметить теперь, что для вершины e справедливо соотношение $\Gamma^{-1}(e) = \{b\} \subset S$, то становится ясным, что e следует добавить в качестве очередной вершины к множеству $\{a, b\}$. Поэтому в приведенном примере можно опустить шаги 3—8 и сразу от шага 2 перейти, как показано стрелкой, к шагу 9.

Проверка условий (а) и (б) будет, конечно, замедлять итеративную процедуру, и для небольших графов (менее чем с 20 вершинами) не получается никакого улучшения первоначального алгоритма Робертса и Флореса. Но для больших графов эта проверка приводит к заметному сокращению необходимого времени вычислений, уменьшая его обычно в 2 или более раз. Подробные результаты вычислений для различных методов приводятся на рис. 7 в разд. 7.3.

7.2.3. Мультицепной метод

После внимательного изучения операций алгоритма перебора Робертса и Флореса становится очевидным, что даже после сделанного улучшения не слишком много внимания уделяется оставшейся части графа, в которой берется последовательность вершин, продолжающих построенную цепь. Обычно построение цепи S_0 в процессе поиска (S_0

рассматривается и как упорядоченное множество вершин, и как обычное множество) подразумевает существование еще каких-то цепей в других частях графа. Эти предполагаемые цепи либо помогают быстрее построить гамильтонов цикл, либо указывают на отсутствие такого цикла, содержащего цепь S_0 , что позволяет сразу прибегнуть к возвращению.

Метод, описанный в этом разделе, был предложен первоначально Селби для неориентированных графов; здесь дается его небольшое видоизменение для ориентированных графов. Метод состоит в следующем.

Допустим, что на некотором этапе поиска построена цепь S_0 и возможны цепи S_1, S_2, \dots . Рассмотрим какую-либо «среднюю» вершину одной из этих цепей (слово «средняя» здесь означает любую вершину, отличную от начальной и конечной). Поскольку эта вершина уже включена в цепь с помощью двух дуг, то очевидно, что все другие дуги, входящие или выходящие из такой вершины, могут быть удалены из графа. Для любой начальной вершины вышеуказанных цепей можно удалить все дуги, исходящие из нее (за исключением дуги, включающей эту вершину в цепь), а для любой конечной вершины можно удалить все дуги, оканчивающиеся в ней (опять-таки за исключением дуги, включающей ее в цепь). Кроме того, за исключением случая, когда существует только одна цепь (скажем, S_0), проходящая через все вершины графа G (т. е. когда S_0 — гамильтонова цепь), любая имеющаяся дуга, ведущая из конца любой цепи в начальную вершину этой же цепи, может быть удалена, так как такая дуга замыкает не гамильтоновы циклы.

Удаление всех этих дуг даст граф со многими вершинами — всеми «средними» вершинами цепей — в котором только одна дуга оканчивается в каждой вершине и только одна дуга исходит из нее. Все эти «средние» вершины и дуги, инцидентные им, удаляются из G , а вместо них для каждой цепи вводится единственная дуга, идущая от начальной вершины цепи до ее конечной вершины. В результате всего этого получается *редуцированный граф* $G_k = (X_k, \Gamma_k)$, где k — индекс, показывающий номер шага поиска.

Рассмотрим теперь продолжение цепи S_0 (сформированной в результате поиска), осуществляемое путем добавления вершины x_j , которая является возможной в смысле алгоритма Робертса и Флореса, т. е. в G_k существует дуга, исходящая из конечной вершины цепи S_0 — обозначим эту вершину $e(S_0)$ — и входящая в вершину x_j . Добавление x_j к S_0 осуществляется так:

(1) Сначала удаляются из G_k все необходимые дуги, т. е.

- (а) все дуги, оканчивающиеся в x_j или исходящие из $e (S_o)$, за исключением дуги $(e (S_o), x_j)$;
 - (б) все дуги, выходящие из x_j в начальную вершину пути S_o ;
 - (в) если окажется, что x_j является начальной вершиной другой цепи S_p , то следует удалить также любую дугу, ведущую из конечной вершины цепи S_j в начальную вершину цепи S_o .
- (2) Обозначим граф, оставшийся после удаления всех дуг, через $G'_k = (X'_k, \Gamma'_k)$.

Если существует вершина x в графе G'_k , не являющаяся конечной ни для одной из цепей S_o, S_1, \dots и которая после удаления дуг имеет полустепень захода, равную единице, т. е. $|\Gamma^{-1}_{k'}(x)| = 1$, то выкинуть все дуги, исходящие из вершины $v = \Gamma^{-1}_{k'}(x)$, за исключением дуги (v, x) .

Если существует вершина x графа G'_k , не являющаяся начальной ни для какой цепи и которая после удаления дуг имеет полустепень исхода, равную единице, т. е. $|\Gamma_{k'}(x)| = 1$, то выкинуть все дуги, исходящие из вершины x , за исключением дуги $(x, \Gamma_{k'}(x))$.

Перестроить все цепи и удалить дуги, ведущие из конечных в начальные вершины.

Повторять шаг 2 до тех пор, пока можно удалять дуги.

- (3) Удалить из оставшегося графа G'_k все вершины, полустепени захода и исхода которых равны единице, т. е. вершины, которые стали теперь «средними» вершинами цепей. Это удаление производится так, как это было описано выше, в результате чего получается новый редуцированный граф G_{k+1} , заменяющий предыдущий граф G_k .

Совершенно очевидно, что если добавление вершины x_j к цепи S_o делает полустепень захода или полустепень исхода (или обе) некоторой вершины x в конце шага 2 равной нулю, то не существует никакого гамильтонова цикла. В этом случае вершина x_j удаляется из множества S_o и в качестве другой вершины x_p , позволяющей продолжить цепь S_o , выбирается некоторая другая вершина из множества $\Gamma_{k'}[e(S_o)]$, и так до тех пор, пока не будет исчерпано все множество $\Gamma_{k'}[e(S_o)]$ и придется прибегнуть к возвращению (т. е. $e(S_o)$ удаляется из S_o и заменяется другой вершиной и т. д.). Отметим, что операция возвращения предполагает хранение достаточной информации об удаленных дугах в шагах 1 и 2 на каждом этапе k , чтобы можно было по графу G_{k+1} восстановить граф G_k при любых k , если приходится прибегать к возвращению.

Если (на некотором этапе) в конце шага 2 установлено, что только одна цепь проходит далее через все вершины, то в этом случае существование гамильтонова цикла может быть выявлено

непосредственно. Если цикл при этом не найден (или если он найден, но надо найти все гамильтоновы циклы), то нужно прибегнуть к возвращению.

Лучший вычислительный путь состоит в том, чтобы при каждой итерации шага 2 (за исключением последней) проверять отличие от нуля полустепеней захода и исхода всех вершин графа G'_k . Поэтому как только одна из них станет равной нулю, сразу же применяется операция возвращения, и если найдена гамильтонова цепь, то получается и гамильтонов цикл без проверки существования дуги возврата (здесь дуга, замыкающая гамильтонову цепь).

Если не возникает ни один из вышеупомянутых случаев, т. е. если (на некотором этапе k) в конце шага 2 остается более чем одна цепь и все полустепени являются ненулевыми, то нельзя еще сделать никаких выводов. Тогда вершина x_j добавляется к S_o и выбирается другая вершина для дальнейшего продолжения цепи. Шаги 1, 2, 3 повторяются, начиная с нового редуцированного графа.

Пример. В настоящем примере будет показано, как итеративный процесс шага 2 приводит ко многим цепям, получающимся из основной цепи S_o , и как эти цепи могут привести к быстрому окончанию производимого поиска.

Рассмотрим часть графа, изображенную на рис. 4.

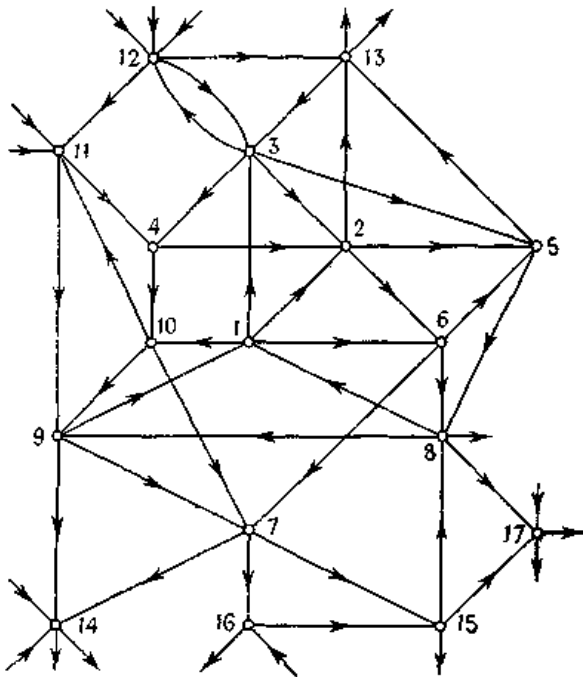


Рис. 4. Граф из примера.

Сначала полустепени захода и исхода каждой вершины больше единицы.

Первая итерация. Начнем с вершины 1 как начальной вершины строящейся цепи S_0 , и попробуем добавить к S_0 , например, вершину 6, так что цепь S_0 будет $\{1, 6\}$.

Шаг 1 описанного метода удалит тогда из G дуги $(1,2)$, $(1,3)$, $(1, 10)$, $(2,6)$.

Шаг 2 укажет теперь вершину 10, как имеющую полустепень захода 1 и дающую возможную цепь $S_1 = \{4,10\}$, после чего удаляется дуга $(4,2)$. Второе применение шага 2 дает вершину 2 с полустепенью захода 1 и возможную цепь $S_2 = \{3,2\}$, после чего удаляются дуги $(3,12)$, $(3,5)$ и $(3, 4)$. Третье применение шага 2 даст вершину 4 с полустепенью захода 1 и приводит к продолжению цепи S_1 после добавления дуги $(11, 4)$, так что цепь S_1 есть теперь $\{11,4,10\}$. В силу шага 2 дуга $(11, 9)$ удаляется, и то же относится к дуге $(10,11)$, замыкающей цепь S_1 . Четвертое применение шага 2 не приводит к дальнейшим удалениям, и первый редуцированный граф,

получающийся в конце шага 3, показан на рис. 5а, где цепи S_0 , S_1 и S_2 изображены жирными линиями.

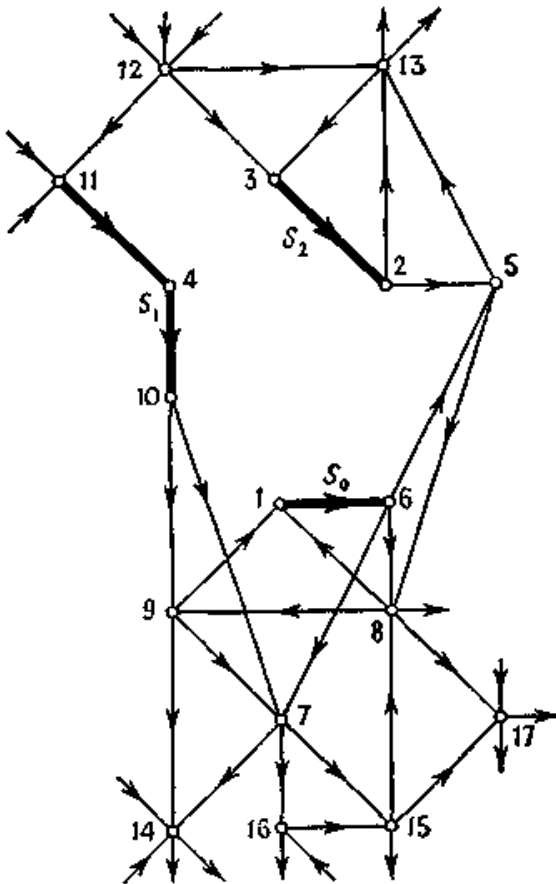


Рис. 5а. Первый редуцированный граф G_1 .

Вторая итерация. Так как существует более чем одна цепь и так как полустепени захода и исхода всех вершин ненулевые, то мы продолжаем расширять цепь S_0 дальше. Пусть, например, из множества $\Gamma(e(S_0)) = \{8, 5, 7\}$ выбрана вершина 8, так что цепь S_0 превратится в цепь $\{1, 6, 8\}$. Шаг 1 даст теперь вершину 5, обе полустепени которой равны 1, и вершину 1 с полустепенью захода, равной 1. Взяв сначала вершину 5 и сделав два раза шаг 2, придем к добавлению дуг $(2, 5)$ и $(5, 13)$ для расширения цепи S_2 , которая превращается теперь в $\{3, 2, 5,$

13}. Дуги (2, 13), (12,13) и дуга возврата (13, 3) удаляются. Второе применение шага 2 дает вершину 3 с полустепенью захода 1, а также вершину 1 (с полустепенью захода 1), полученную на предыдущем этапе. Взяв вершину 3, добавим к S_2 дугу (12, 3), после чего получим цепь {12, 3, 2, 5, 13} и удалим дугу (12, 11). Третье применение шага 2 все еще дает вершину 1 с полустепенью захода 1 и больше не дает никаких вершин. К цепи S_0 добавляется дуга (9,1), цепь S_0 превращается в {9, 1, 6, 8}, а дуги (9, 7), (9,14) и дуга возврата (8,9) удаляются. Четвертое применение шага 2 укажет две вершины 9 и 7 с полустепенями захода 1. Взяв сначала вершину 9, добавим дугу (10,9), соединяющую цепи S_1 и S_0 и дающую новую цепь $S_0 = \{11, 4, 10, 9, 1, 6, 8\}$. Дуга (10, 7) удаляется.

Пятое применение шага 2 дает вершину 7 с полустепенью захода 0. Это говорит о том, что описанный поиск не может привести ни к какому гамильтонову циклу.

Граф G'_1 , получающийся в конце четвертого применения шага 2, показан на рис. 56.

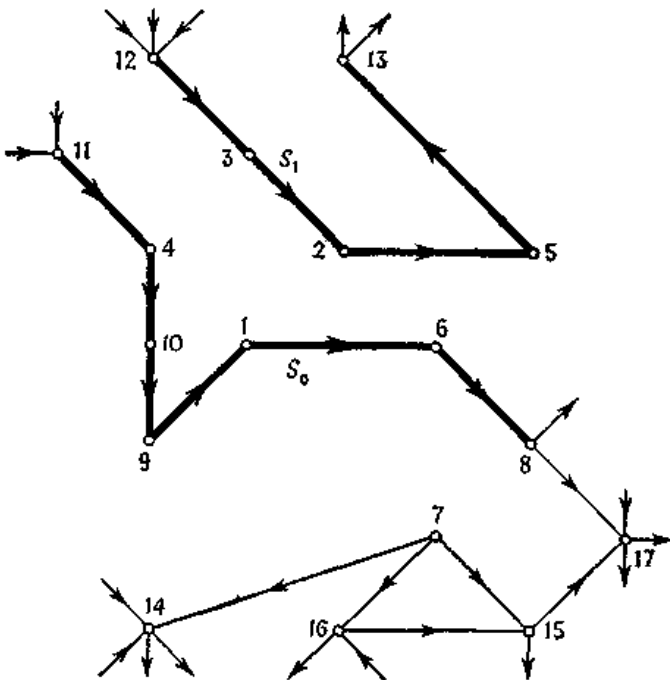


Рис. 56. Граф G'_1 после четвертого применения шага 2 на второй итерации.

Теперь следует восстановить все удаленные во время второй итерации дуги, чтобы вернуться к графу G_1 из рис. 5а, и произвести операцию возвращения, т. е. удалить из S_0 вершину 8, заменить ее другой возможной вершиной (5 или 7) и продолжать применять шаги 1, 2, 3 и т. д.

Из приведенного примера видно, что это очень сильный метод поиска. После двух итераций он позволяет сделать вывод, что никакой гамильтонов цикл не может содержать в качестве своей части цепь $\{1, 6, 8\}$. Это 1/12 всех попыток поиска, поскольку полустепень исхода вершины 1 равна 4, а вершины 6 — 3. Конечно, указанный вывод относится только к части графа, изображенной на рис. 4, которая сама может принадлежать к намного большему графу. Можно поэтому ожидать, что при заданном среднем значении степеней вершин поиск лишь слабо зависит от размера (числа вершин) графа. Этот факт будет продемонстрирован на экспериментальных данных в следующем разделе.

Напротив, метод Робертса и Флореса (или его улучшенный вариант) для достижения того же результата требует очень длительного поиска, так как в нем существенно перебираются все цепи, начинающиеся с 1, 6, 8, . . . , прежде чем произойдет возвращение к 1, 6, Очевидно, что затрачиваемая работа зависит как от размера графа, так и от степеней вершин.

7.3. Сравнение методов поиска гамильтоновых циклов

В настоящем разделе сравниваются первоначальный вариант алгоритма Робертса и Флореса, его улучшенный вариант и мультицепной метод. Эти три метода сравниваются по необходимому времени вычисления для нахождения одного гамильтонова цикла, если таковой существует, ИЛИ доказательства его отсутствия. Проверка была проведена на случайно выбранных графах, степени вершин которых лежат в предписанных границах. Всего было использовано около 200 графов, для которых приводятся средние результаты. Во всех графах оказались гамильтоновы циклы.

На рис. 6 показана зависимость требуемого алгоритмом Робертса и Флореса времени вычисления от числа вершин графа; степени вершин лежат в пределах 3 — 5.

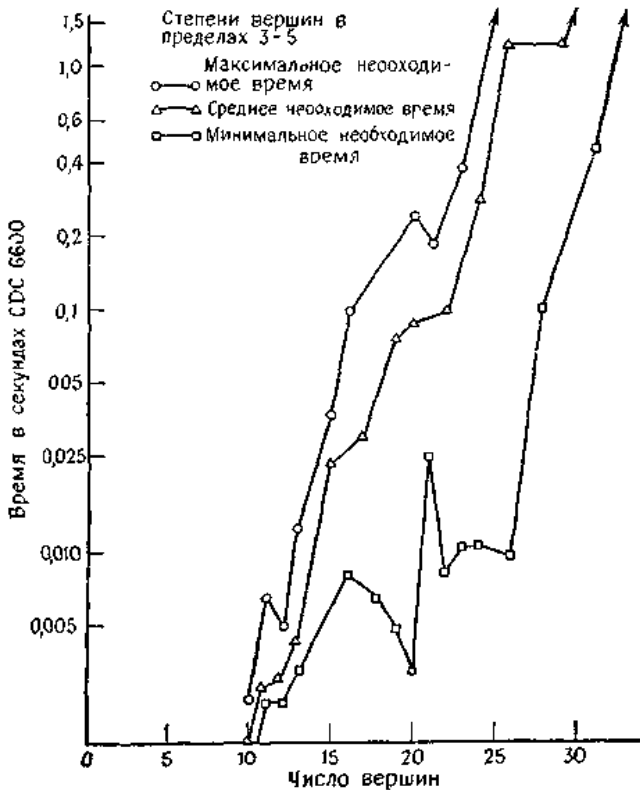


Рис. 6. Вычислительная реализация алгоритма Робертса и Флореса.

Ввиду сильных вариаций требуемого времени для графов одинаковых размеров приводятся три кривые, характеризующие среднее, максимальное и минимальное время, полученное для различных графов с одинаковым числом вершин. Следует заметить, что на рис. 6 применен полулогарифмический масштаб, что говорит об экспоненциальном характере зависимости. Формула, дающая приближенную зависимость времени T от числа вершин n графа со степенями вершин в пределах 3 — 5, такова:

$$T = 0,85 \cdot 10^{-4} \cdot 10^{0,155n} \text{ (секунд на CDC 6600).}$$

Улучшенный вариант алгоритма Робертса и Флореса не намного лучше первоначального алгоритма. Необходимое время вычисления в нем все еще зависит (более или менее) экспоненциально от n . Зависимость отношения времен вычисления при использовании этих двух

алгоритмов для неориентированных графов со степенями вершин 3 — 5 приведена на рис. 7.

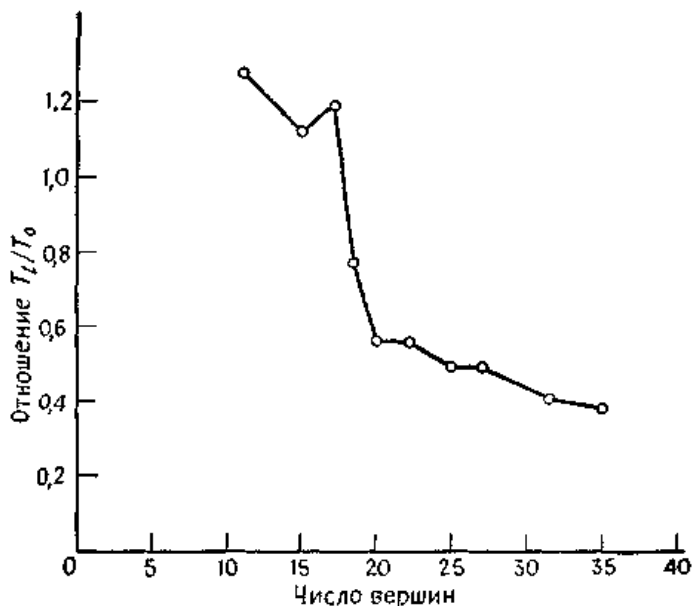


Рис. 7. Реализация улучшенного алгоритма Робертса и Флореса. T_0 — время вычисления для первоначального метода, T_i — время вычисления для улучшенного метода.

Из этого рисунка видно, что «улучшенный» вариант действительно хуже для графов малых размеров, хотя для больших графов (с более чем 20 вершинами) он позволяет сэкономить более 50% времени вычисления. По отношению к тому же вышеупомянутому множеству графов мультицепной алгоритм оказался очень эффективным. Это видно из рис. 8, на котором показано необходимое для этого алгоритма время вычисления (здесь применен линейный масштаб).

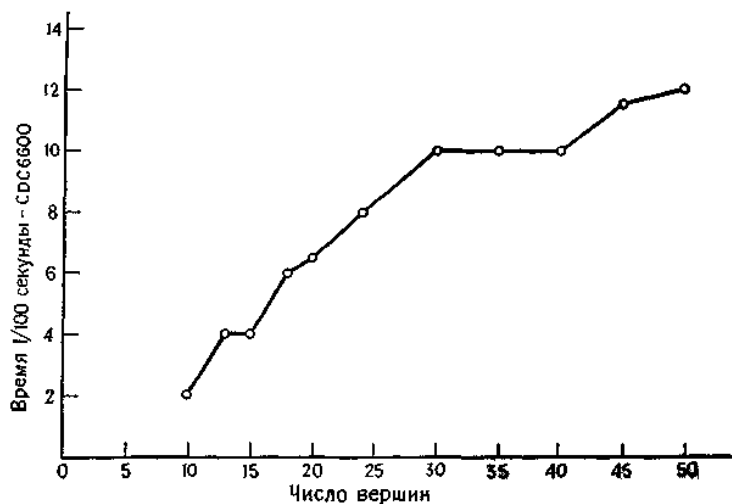


Рис. 8. Вычислительная реализация мультицепного алгоритма для

График показывает, что время растет *очень медленно* в зависимости от числа вершин и поэтому алгоритм применим для очень больших графов. Другим преимуществом этого метода является очень слабая вариация времени вычисления для различных графов одинакового размера, и поэтому можно оценить с разумной степенью достоверности время вычисления, необходимое для различных задач. Кроме того, эксперименты показывают, что для графов, степени вершин которых лежат в вышеприведенных пределах 3 — 5, метод по существу не чувствителен к степеням вершин.

Вычислительные результаты, показанные на рис. 6 — 8, относятся к поиску одного гамильтонова цикла в графе. Небезынтересно сказать несколько слов о вычислениях с тремя алгоритмами, когда искались все гамильтоновы циклы. Так, для неориентированного графа с 20 вершинами со степенями вершин 3 — 5 потребовалось 2 с, чтобы найти все гамильтоновы циклы, следуя алгоритму Робертса и Флореса (этих циклов оказалось 18). Улучшенный вариант того же алгоритма потребовал 1,2 с, а мультицепной алгоритм — 0,07 с. Вычисления проводились на ЭВМ CDC 6600.

7.4. Простая задача планирования

Во введении к данной главе мы описали пример промышленной фирмы, выпускающей продукты p_1, p_2, \dots, p_n с использованием единственного типа аппаратуры в циклическом режиме, и поставили вопрос о порядке производства продуктов, требующем наименьшего возможного числа перенастроек аппаратуры. Был построен граф $G = (X, A)$, вершины x_i которого представляют продукты p_i ($i = 1, 2, \dots, n$), а дуги (x_i, x_j) показывают, что продукт p_j можно производить вслед за продуктом p_i без перенастройки.

Если граф $G = (X, A)$ имеет гамильтонов цикл, например $x_{i_1}, x_{i_2}, x_{i_3}, \dots, x_{i_n}$, то соответствующая последовательность продуктов $x_{i_1}, x_{i_2}, \dots, x_{i_n}$ может быть произведена на аппаратуре без какой-либо перенастройки, так как по определению гамильтонова цикла

$(x_{i_k}, x_{i_{k+1}}) \in A$ для всех $k = 1, 2, \dots, (x_{i_{n+1}} \equiv x_{i_1})$ и A определено выше как множество $\{(x_i, x_j) / c_{ij} = 0\}$.

Если G не имеет никакого гамильтонова цикла, то мы можем построить граф $G_1 = (X_1, A_1)$, где

$$X_1 = X \cup \{y_1\}$$

и

$$A_1 = A \cup \{(x, y_1) \mid x \in X\} \cup \{(y_1, x) \mid x \in X\},$$

т. е. ввести в граф G фиктивную вершину y_1 вместе с дугами, ведущими в нее и исходящими из нее в каждую действительную вершину графа G .

Если у графа G_1 существует гамильтонов цикл, то он будет иметь вид $x_{i_1}, x_{i_2}, \dots, x_{i_r}, \boxed{y_1}, x_{i_{r+1}}, \dots, x_{i_n}$, что приводит к следующей последовательности производства продуктов

$$p_{i_{r+1}}, p_{i_{r+2}}, \dots, p_{i_n}, p_{i_1}, p_{i_2}, \dots, p_{i_r}$$

с единственной операцией перенастройки после окончания производства последнего продукта и началом производства первого. Таким образом, фиктивная вершина играет роль метки, показывающей то место в последовательности, где необходима перенастройка аппаратуры. В терминах графа фиктивная вершина и ассоциированные с ней дуги обеспечивают существование цепи между двумя действительными вершинами. Таким образом, если существует

последовательность продуктов с одной перенастройкой, т. е. если в G существует гамильтонов цикл, при условии, что можно использовать одну «дугу» $(x_i, x_j) \notin A$, то добавление вершины y_l к G всегда приведет к существованию в G_l гамильтонова цикла, так как лишняя необходимая «дуга» (x_i, x_j) может быть заменена двумя фиктивными дугами (x_i, y_l) и (y_l, x_j) .

Если граф G_1 не имеет гамильтонова цикла, то мы построим граф $G_2 = (X_2, A_2)$ вместографа G_1 , где

$$X_2 = X_1 \cup \{y_2\}$$

и

$$A_2 = A_1 \cup \{(x, y_2) \mid x \in X\} \cup \{(y_2, x) \mid x \in X\},$$

и аналогично будем действовать далее.

Теорема 1. *Если граф $G_m = (X_m, A_m)$, определяемый как*

$$X_m = X \cup \left[\bigcup_{j=1}^m \{y_j\} \right], \quad (2)$$

$$A_m = A \cup \left[\bigcup_{j=1}^m \{(x, y_j) \mid x \in X\} \right] \cup \left[\sum_{j=1}^m \{(y_j, x) \mid x \in X\} \right], \quad (3)$$

содержит гамильтонов цикл, а граф G_{m-1} такого цикла не содержит, то число m есть минимальное число необходимых перенастроек, и если гамильтонов цикл в G_m имеет вид

$$x_{i_1}, \dots, x_{i_\alpha}, \boxed{y_1}, x_{i_{\alpha+1}}, \dots, x_{i_\beta}, \boxed{y_2}, x_{i_{\beta+1}}, \dots, x_{i_\gamma}, \boxed{y_3}, \\ x_{i_{\gamma+1}}, \dots, x_{i_\delta}, \boxed{y_m}, x_{i_{\delta+1}}, \dots, x_{i_n},$$

То продукты должны производиться в последовательности

$$P_{i_{\alpha+1}}, \dots, P_{i_\beta}, \text{ затем } P_{i_{\beta+1}}, \dots, P_{i_\gamma}, \dots \text{ и т. д. } \dots$$

$$\text{затем } P_{i_{\delta+1}}, \dots, P_{i_n}, P_{i_1}, \dots, P_{i_\alpha}.$$

Доказательство. Доказательство следует непосредственно из рассуждений, предшествующих теореме, с использованием индукции.

7. 4.1. Вычислительные аспекты

В рассуждениях предыдущего раздела граф G каждый раз пополнялся единственной фиктивной вершиной. Если оптимальное решение задачи содержит m перенастроек аппаратуры, то нужно будет сделать $m + 1$ попыток отыскать гамильтоновы циклы в графах G, G_1, \dots, G_m , причем только последняя из этих попыток окажется успешной и приведет к решению задачи. Очевидно, что число m ограничено сверху величиной

n и в общем случае в практических задачах оно будет лишь небольшой частью числа n . Тем не менее с вычислительной точки зрения необходимы различные процедуры для проверки наличия в графе гамильтоновых циклов и построения таких циклов, так как оказывается, что проще найти гамильтонов цикл в графе, имеющем такой цикл, чем доказать, что не существует никакого гамильтонова цикла в графе, его не имеющем. Этот факт подсказывает, что более подходящим будет алгоритм, который начинает с верхней границы B для оптимального (минимального) числа перенастроек аппаратуры и последовательно дает и проверяет графы G_B, G_{B-1}, \dots и т. д. до тех пор, пока не будет найден граф G_{m-1} , не имеющий гамильтоновых циклов. Краткое описание такого алгоритма дается ниже.

Шаг 1. Найти верхнюю границу B для оптимального (минимального) числа необходимых перенастроек аппаратуры (см. приложение).

Шаг 2. Используя формулы (2) и (3), построить граф G_B .

Шаг 3. Определить, имеет ли граф G_B гамильтонов цикл. Если да, то хранить этот цикл в виде вектора H , записав его на место предыдущего хранящегося цикла, и перейти к шагу 4 или шагу 5.

Шаг 4. $B \leftarrow B - 1$, перейти к шагу 2.

Шаг 5. Останов. $m = B + 1$ является минимальным числом перенастроек аппаратуры, и последняя последовательность в H является требуемой последовательностью производства продуктов.

Шаг 1 приведенного выше алгоритма требует дальнейших пояснений. Очевидно, что чем более точной будет начальная верхняя граница B , тем меньшее число итераций основного алгоритма придется сделать. Процедура получения хорошей оценки для верхней границы дана в приложении. Существование гамильтонова цикла в графе G_B может быть установлено с использованием мультицепного алгоритма разд. 7.2.3.

7.5. Задача коммивояжера

Задача коммивояжера тесно связана с несколькими другими задачами теории графов, обсуждаемыми в других частях этой книги. Здесь мы изучим две такие связи — с задачей о назначениях (см. далее) и с задачей о кратчайшем остове. Обе эти задачи могут быть решены намного проще, чем задача коммивояжера, и можно исследовать эти связи для того, чтобы разработать эффективный метод решения последней задачи.

7.5.1. Нижняя граница из задачи о назначениях

Линейную задачу о назначениях для графа с произвольной матрицей весов $C = [c_{ij}]$ можно сформулировать так.

Пусть $[\xi_{ij}]$ — $(n \times n)$ -матрица, в которой $\xi_{ij} = 1$, если вершина x_i «назначена» к вершине x_j и $\xi_{ij} = 0$, если x_i не назначена к x_j . Таковую же схему можно использовать и в задаче коммивояжера, полагая $\xi_{ij} = 1$, если коммивояжер едет непосредственно из x_i в x_j , и $\xi_{ij} = 0$ в противном случае. В этой последней задаче можно положить $c_{ii} = \infty$ ($i = 1, \dots, n$), чтобы устранить бессмысленные решения с $\xi_{ii} = 1$.

Теперь задача о назначениях формулируется так.

Найти величины ξ_{ij} , минимизирующие

$$z = \sum_{j=1}^n \sum_{i=1}^r c_{ij} \xi_{ij}, \quad (4)$$

при условии, что

$$\sum_i \xi_{ij} = \sum_j \xi_{ij} = 1 \quad (5)$$

(для всех $i, j = 1, 2, \dots, n$)

$$\xi_{ij} = 0 \text{ или } 1. \quad (6)$$

Условие (5) просто гарантирует, что решение будет циклическим, т. е. в каждую вершину входит и из нее выходит одна дуга.

Уравнения (4) — (6) вместе с дополнительным ограничением, состоящим в том, что решение должно давать *единственный* цикл (гамильтонов), а не просто некоторое число несвязных циклов также могут быть использованы для формулировки задачи коммивояжера. Заметим, что дополнительное условие $c_{ii} = \infty$ можно интерпретировать как ограничение, устраняющее возможность появления в решении задачи о назначениях циклов длины 1. Так как добавление любого ограничения в задаче о назначениях не может увеличить минимального значения z , определяемого из уравнений (4) — (6), то это значение z является нижней границей веса решения задачи коммивояжера для графа с матрицей весов $[c_{ij}]$.

7.5.2. Нижняя граница из задачи о кратчайшем остове

Для графа с симметричной матрицей весов (расстояний) C , т. е. для неориентированного графа, нижнюю границу для решения задачи коммивояжера можно получить по кратчайшему остову графа следующим образом. Пусть установлено, что ребро (x_1, x_2) входит в

оптимальный цикл задачи коммивояжера. Если это ребро удалить из цикла, то получится цепь, состоящая из $n - 1$ ребер, проходящая через все вершины, начиная с вершины x_1 , и оканчивающаяся в x_2 . Так как вес кратчайшего остова, скажем L , является нижней границей для веса этой цепи, то длина кратчайшего остова плюс $c(x_1, x_2)$ будет нижней границей для веса оптимального решения задачи коммивояжера. В общем случае заранее не известно, какие ребра содержатся в оптимальном цикле, но самое длинное ребро в цикле должно быть не короче, чем

$$\max_{x_i} [c(x_i, s)].$$

Здесь s — вторая ближайшая вершина к вершине x_i . Таким образом, величина

$$L + \max_{x_i} [c(x_i, s)] \quad (7)$$

является нижней границей веса решения в задаче коммивояжера

7.5.3. Двойственность

Определим $G(\text{TSP})$ как остовный подграф неориентированного графа G , образованный вершинами и теми ребрами графа G , которые использованы в оптимальном цикле коммивояжера. Аналогично определим графы $G(\text{AP})$ и $G(\text{SST})$, образованные теми же вершинами, но ребра которых берутся из оптимальных решений задачи о назначениях и задачи о кратчайшем остове соответственно.

Граф $G(\text{TSP})$ обладает следующими свойствами:

- (1) граф является связным, т. е. каждую вершину можно соединить с любой другой вершиной цепью,
- (2) степень каждой вершины равна 2, т. е. каждая вершина инцидентна двум ребрам.

На рис. 9 (а) приведен пример 5-вершинного графа $G(\text{TSP})$.

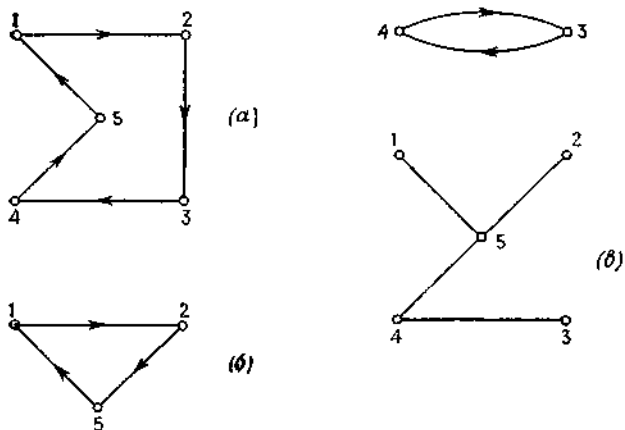


Рис. 9. Графы задач коммивояжера, о назначениях и о кратчайшем остове.

(а) $G(TSP)$. (б) $G(AP)$. (в) $G(SST)$.

Граф $G(AP)$ не обязательно обладает свойством (1), как это видно из рис. 9 (б), но по определению обладает свойством (2). Если, однако, окажется, что для решения задачи о назначениях выполняется и свойство (1), то это решение будет также решением задачи коммивояжера.

Для графа $G(SST)$ свойство (1) выполняется по определению, но он может не иметь свойства (2). Если же окажется, что для кратчайшего остова свойство (2) выполняется — за исключением двух «конечных» вершин (скажем, x_1 и x_2), которые необходимо имеют степень 1, — то кратчайший остов будет также кратчайшей цепью, проходящей через все n вершин. Если, кроме того, ребро (x_1, x_2) входит в оптимальный цикл задачи коммивояжера, то ребра кратчайшего остова вместе с ребром (x_1, x_2) дают решение задачи коммивояжера. (Если же (x_1, x_2) не обязательно принадлежит оптимальному циклу задачи коммивояжера, то необходима некоторая модификация.

Таким образом, **решения задач о назначениях и кратчайшем остове являются двойственными** в том смысле, что они обладают свойствами, являющимися дополнительными по отношению к свойствам задачи коммивояжера. Теперь обнаруживаются два пути, могущие привести к решению последней задачи.

(I) Использовать решение задачи о назначениях (для которого выполняется свойство (2)) и добиться, чтобы это решение подчинялось свойству (1).

(II) Использовать решение задачи о кратчайшем остове (для которого выполняется свойство (1)) и добиться выполнения свойства (2).

В разд. 7.6 мы исследуем путь, основанный на методе (II), а в разд. 7.7 будет рассмотрен метод (I). Следует, однако, помнить, что хотя задача о назначениях определена для графов с произвольной структурой весов, кратчайший остов определяется только для неориентированных графов, т. е. для графов с симметричной ($c_{ij} = c_{ji}$) матрицей весов. Для несимметричных матриц весов (т. е. ориентированных графов) ранее было введено понятие древесности — аналог понятия остова. Таким образом, то, что было сказано о связи между задачей коммивояжера и кратчайшим остовом для неориентированных графов, имеет точный эквивалент, относящийся к связи между задачей коммивояжера и кратчайшей древесностью в случае ориентированных графов. В следующем разделе, однако, мы ограничимся симметричной задачей, так как распространение на более общий случай производится очевидным способом.

7.6. Задача коммивояжера и задача о кратчайшем остове

Как уже отмечалось в предыдущем разделе, симметричная задача коммивояжера тесно связана с задачей о кратчайшем остове графа G , и *открытая* задача коммивояжера (т. е. задача нахождения кратчайшей гамильтоновой *цепи* в графе) практически эквивалентна задаче нахождения кратчайшего остова графа G , с тем ограничением, что никакая вершина не должна иметь степень, большую чем 2. Ввиду того, что открытая задача коммивояжера несколько более сильно связана с задачей о кратчайшем остове, чем замкнутая, мы начнем с рассмотрения сначала этой задачи, отложив на конец раздела обсуждение тех небольших изменений, которые необходимо сделать в замкнутой (обычной) задаче коммивояжера.

Задача нахождения кратчайшей гамильтоновой цепи была впервые исследована Део и Хакими, давшими ее формулировку на языке линейного программирования. Для полного графа с n вершинами их формулировка содержит $n(n+1)$ переменных и $n(n+3)/2+1$ ограничений, которые формулируются явно. Наряду с этим имеется очень большое число ограничений, которые нельзя сформулировать явно, но которые рассматриваются неявно, причем за один раз (после каждого итерационного применения симплекс-метода) вводятся немногие из них. Хотя метод линейного программирования и дает

всегда решение, он не будет здесь больше обсуждаться, так как обладает врожденной громоздкостью и неэффективностью.

7.6.1. Определения

Пусть $G = (X, A)$ — реберно-взвешенный неориентированный граф, а d_i^G — степень вершины x_i в графе G . Число вершин в G будет обозначаться через $n = |X|$.

Если задан остов $T = (X, A_T)$ графа G и степень вершины x_i в дереве T обозначена через d_i^T , можно определить близость этого остова к гамильтоновой цепи одним из следующих двух способов:

$$\varepsilon_T = \sum_{d_i^T > 2} (d_i^T - 2) \quad (8(a))$$

или

$$\varepsilon_T = \sum_{i=1}^n |d_i^T - 2| - 2. \quad (8(б))$$

Выражение (8)(а) учитывает близость только по тем вершинам, для которых $d_i^T > 2$, в то время как (8)(б) принимает во внимание и вершины степени 1. Оба эти выражения для гамильтоновой цепи дают $\varepsilon_T = 0$ и можно считать, что чем больше ε_T , тем сильнее дерево T отличается от гамильтоновой цепи.

Теперь мы рассмотрим две следующие задачи.

Задача (а). *Кратчайшая гамильтонова цепь.* Найти кратчайший остов $T^* \equiv (X, A^*)$ графа G , такой, что степени всех вершин не превышают 2.

Задача (б). *Кратчайшая гамильтонова цепь с отмеченными концевыми вершинами.* Заданы две вершины x_1 и x_2 ($x_1, x_2 \in X$), найти кратчайший остов $T^*_{1,2} \equiv (X, A_{1,2})$, такой, что степени всех вершин не превышают 2, а степени вершин x_1 и x_2 равны 1.

Теперь нужно проверить подразумеваемое в сформулированных задачах утверждение о том, что дерево T , все вершины которого имеют степень, не больше чем 2, является на самом деле гамильтоновой цепью. Действительно, так как T — дерево, то степень никакой его вершины не может быть равной нулю и, следовательно, $d_i^T = 1$ или 2 для всех i . Пусть q вершин имеют степень 1, а $n - q$ имеют степень 2. Число ребер в дереве равно

$$m_T = \frac{1}{2} \sum_{i=1}^n d_i^T, \quad (9)$$

так как при суммировании каждое ребро считается дважды, по одному разу для каждой из его конечных вершин.

Из уравнения (9) получим

$$m_T = \frac{1}{2} [q + 2(n - q)] = n - \frac{q}{2}.$$

Так как число ребер в дереве равно $n - 1$, то $q = 2$ и поэтому ровно две вершины имеют степень 1, а $n - 2$ вершин — степень 2, т. е. T является гамильтоновой цепью.

7.6.2. Алгоритм поиска, использующий дерево решений

7.6.2.1. Решение задачи (а). Мы рассмотрим сначала задачу (а) и объясним суть алгоритма, использующего дерево решений, на одном примере.

Пример. В качестве примера возьмем полный граф с 6 вершинами со следующей матрицей весов:

	1	2	3	4	5	6
1	0	4	10	18	5	10
2	4	0	12	8	2	6
3	10	12	0	4	18	16
4	18	8	4	0	14	6
5	5	2	18	14	0	16
6	10	6	16	6	16	0

$C = [c_{ij}] =$

С помощью метода построения всех остовных деревьев графа, легко найти кратчайший остов этого графа (без ограничений на d^T_i); он показан на рис. 10 (а).

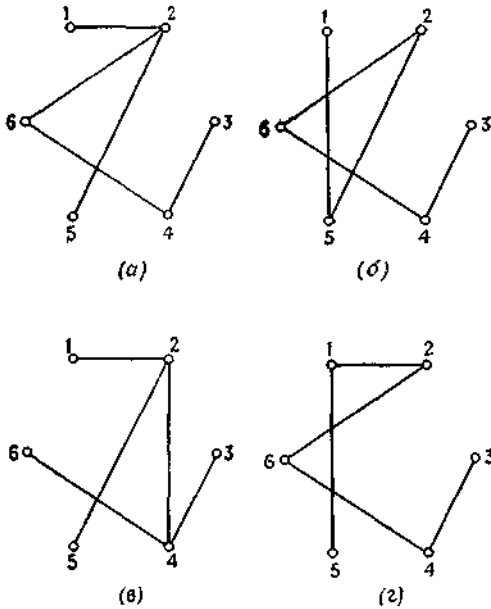


Рис. 10. Кратчайшие остовы в процессе поиска.
 (б) $T^*(A)$, вес 22. (б) $T^*(B)$, вес 23. (в) $T^*(C)$, вес 24. (г) $T^*(D)$, вес 25.

Из этого рисунка видно, что $d^T_2 = 3$, а так как нам нужен кратчайший остов, для которого $d^T_i \leq 2$, то можно сказать, что в окончательном ответе должно отсутствовать по крайней мере одно из ребер (2,1), (2,6) или (2,5). Таким образом, решение первоначальной задачи является решением по крайней мере одной из следующих трех частных задач, изображаемых узлами B, C и D дерева решений на рис. 11.

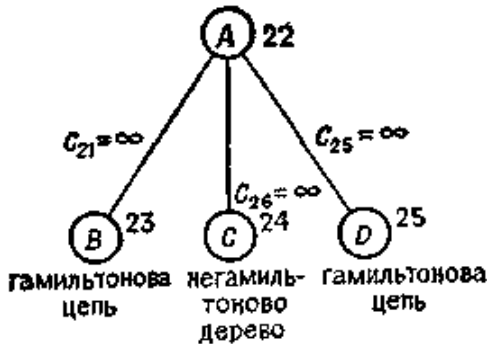


Рис 11. Поиск с деревом решений

На этом рисунке узел A представляет первоначальную задачу, а узлы B , C и D отвечают задачам, матрицы весов которых те же самые, что и матрица весов задачи A , но только ребрам $(2,1)$, $(2,6)$ или $(2,5)$ соответственно приписан бесконечный вес.

На рис. 11 число, стоящее около узла, равно весу кратчайшего остова, соответствующего частной задаче, определяемой этим узлом. Следует заметить, что на любой стадии ветвления наименьший из весов у свободных узлов (т. е. узлов, в которых еще не произошло ветвление) является нижней границей для веса окончательного ответа.

Найдем теперь кратчайшие остовы $T(B)$, $T(C)$ и $T(D)$, соответствующие узлам B , C и D . Результаты изображены соответственно на рис. 10 (б), (в) и (г); веса этих остовов указаны вблизи узлов.

Остовы $T(B)$ и $T(D)$ являются гамильтоновыми цепями (все $d_i^T \leq 2$), в то время как $T(C)$ гамильтоновой цепью не является. Кратчайшим из этих двух остовов будет $T(B)$ с весом 23. Нижняя граница веса для окончательного ответа равна наименьшему из весов остовов $T(B)$, $T(C)$ и $T(D)$ и равна также 23. Таким образом, $T(B)$ дает оптимальный ответ, т. е. является кратчайшей гамильтоновой цепью. Следует заметить, что в узле C не нужно производить дальнейшее ветвление, так как любая гамильтонова цепь, полученная при таком ветвлении будет иметь вес, не меньший чем 24.

Итак, алгоритм, использующий дерево решений, позволил в этом частном примере получить оптимальный ответ, применяя алгоритм кратчайшего остова только 4 раза.

Другим побочным преимуществом данного метода является то, что он часто дает дополнительные гамильтоновы цепи, близкие к оптимальным, которые в практических задачах можно использовать в качестве альтернативного приемлемого решения.

7.6.2.2. Решение задачи (б). В этой задаче надо найти кратчайшую гамильтонову цепь с двумя заданными концевыми вершинами x_1 и x_2 . Эту задачу можно свести к задаче (а), используя следующую теорему.

Теорема 2. Пусть $C = [c_{ij}]$ — матрица реберных весов первоначального графа G и k — достаточно большое положительное число, большее чем вес любой гамильтоновой цепи. Тогда решение задачи (а) с матрицей реберных весов C' , где

$$\left. \begin{aligned} c'_{j1} &= c_{j1} + k, \\ c'_{1j} &= c_{1j} + k, \\ c'_{2j} &= c_{2j} + k, \\ c'_{j2} &= c_{j2} + k, \end{aligned} \right\} \text{(для всех } x_j \neq x_1, x_2) \quad (10)$$

$$\begin{aligned} c'_{ij} &= c_{ij} + 2k, & \text{(для всех } x_i, x_j = x_1, x_2) \\ c'_{ij} &= c_{ij}, & \text{(для всех } x_i, x_j \neq x_1, x_2) \end{aligned}$$

является решением задачи (б) с первоначальной матрицей весов.

Доказательство. Каждая гамильтонова цепь относится к одной из следующих категорий:

- (1) x_1 и x_2 не совпадают ни с одной из ее концевых вершин;
- (2) одной из ее концевых вершин является или x_1 , или x_2 ;
- (3) одной концевой вершиной является x_1 , а другой x_2 . Вес гамильтоновой цепи с матрицей весов C' больше веса этой же цепи с матрицей C на величину:

$4k$, если цепь относится к категории (1);

$3k$, если цепь относится к категории (2);

$2k$, если цепь относится к категории (3).

Так как k превосходит вес любой гамильтоновой цепи, то вес (для матрицы C') самой длинной гамильтоновой цепи категории (3) меньше, чем вес самой короткой гамильтоновой цепи категории (2), а вес самой длинной гамильтоновой цепи категории (2) меньше, чем вес самой короткой цепи категории (1). Таким образом, решение задачи (а) с матрицей весов C' даст кратчайшую гамильтонову цепь категории (3), т. е. она будет решением задачи (б) с матрицей весов C . Это доказывает теорему.

В связи с этим следует отметить, что относительные веса всех гамильтоновых цепей в пределах одной категории не изменяются при преобразовании (10).

7.6.3. Алгоритм штрафования вершин

В этом алгоритме используется та же идея, что и доказанная выше теорема 2, а именно такое преобразование матрицы весов C , при котором различные категории деревьев штрафуются по-разному, но в то же время относительные веса всех деревьев в пределах одной категории не изменяются. Цель при этом состоит в том, чтобы разделить различные категории и сделать категорию гамильтоновых цепей более привлекательной (менее штрафуемой), а тогда применение алгоритма минимального дерева автоматически приведет к кратчайшей гамильтоновой цепи.

Теорема 3. Если матрица C преобразована в матрицу C' так, что

$$c'_{ij} = c_{ij} + p(i) + p(j) \quad \text{для } i, j = 1, 2, \dots, n, \quad (11)$$

где $p(k)$ — n -мерный вектор, образованный действительными положительными и отрицательными постоянными, то относительные веса всех гамильтоновых цепей для матрицы C' , имеющих указанные концевые вершины, не изменяются.

Доказательство. Пусть F_C — вес произвольной гамильтоновой цепи с матрицей C , имеющей концевые вершины x_1 и x_2 . Так как каждая вершина соединена точно с двумя другими вершинами (за исключением концевых вершин, каждая из которых соединена только с одной другой вершиной), то вес F_C той же самой гамильтоновой цепи с матрицей C' отличается от F_C на величину

$$F_{C'} - F_C = 2 \sum_{j \neq 1, 2} p(j) + p(1) + p(2). \quad (12)$$

Это постоянная величина, не зависящая от выбранной гамильтоновой цепи. Отсюда следует теорема.

7.6.3.1. Решение задачи (б). Применение алгоритма нахождения кратчайшей гамильтоновой цепи с указанными концевыми вершинами (задача (б)), основанного на теореме 3, будет продемонстрировано на примере.

Пример. Рассмотрим полный граф G с 10 вершинами, матрица весов C_0 которого дается табл. 1, и предположим, что мы хотим найти кратчайшую гамильтонову цепь с концевыми вершинами 8 и 9.

Матрица весов примера

	1	2	3	4	5	6	7	8	9	10
1	0	28	31	28	22	36	50	67	40	74
2	28	0	31	40	41	64	74	80	63	101
3	31	31	0	14	53	53	53	50	42	83
4	28	40	14	0	50	41	39	41	28	69
5	22	41	53	50	0	40	61	86	53	78
6	36	64	53	41	40	0	24	58	22	39
7	50	74	53	39	61	24	0	37	11	30
8	67	80	50	41	86	58	37	0	36	60
9	40	63	42	28	53	22	11	36	0	41
10	74	101	83	69	78	39	30	60	41	0

В соответствии с теоремой 2 добавим к каждому элементу, стоящему либо в строках 8 и 9, либо в столбцах 8 и 9 таблицы 1, достаточно большое число k (скажем, 1000). Получившуюся после этого добавления таблицу назовем результирующей матрицей весов C .

Далее поступаем следующим образом.

Находим кратчайший остов графа G . Если он окажется гамильтоновой цепью, то задача решена. В рассматриваемом примере кратчайший остов, показанный на рис. 12 (а), не является гамильтоновой цепью.

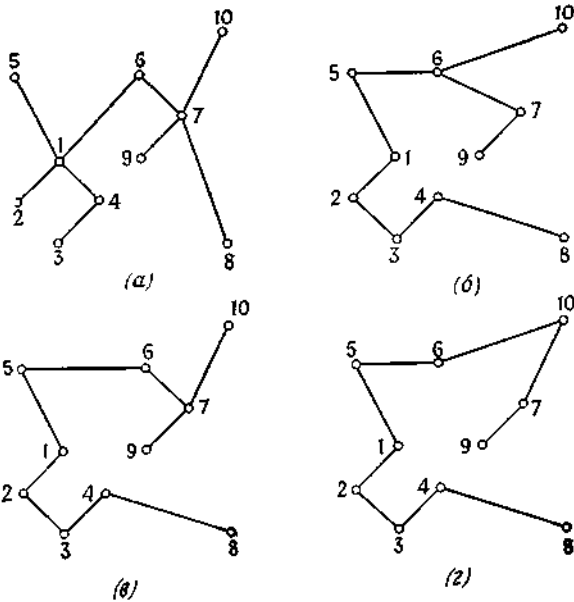


Рис. 12. Кратчайшие остовы при итерациях со штрафами.

Степени вершин 1 и 7 равны 4, а не 2, как это требуется для гамильтоновой цепи. Следуя духу теоремы 3, мы можем «оштрафовать» эти вершины. Допустим, что мы произвольно выбрали малый шаг (скажем, 5 единиц), так что все штрафы кратны ему. (Можно выбрать величину шага равной 1, но это будет неоправдано мало и приведет к увеличению числа требуемых итераций.) Таким образом, если мы решили штрафовать только те вершины, степени которых больше чем 2, то

$$p(i) = 5(d_i^T - 2). \quad (13)$$

Заметим, что метод штрафования вершин выбран произвольно и существует много других альтернатив (см. разд. 7.6.3.4).

В соответствии с уравнением (13) штрафы вершин 1 и 7 равны $p(1) = p(7) = 10$, причем для всех остальных вершин они равны 0. Следуя (11), вычислим новую матрицу весов и найдем для нее кратчайший остов. Результат показан на рис. 12 (б), из которого видно, что это дерево много ближе к гамильтоновой цепи, поскольку для него $\varepsilon = 1$ вместо $\varepsilon = 4$ в предыдущем случае.

Поступая как и ранее, оштрафуем вершину 6 и положим $p(6) = 5$. (Следует заметить, что этот новый штраф добавляется к предыдущей

матрице весов, а не к первоначальной матрице C .) Таким образом, штрафы равны $p(1) = p(7) = 10$, $p(6) = 5$ и $p(i) = 0$ для всех остальных вершин. Соответствующий кратчайший остов изображен на рис. 12 (в), и для него $\varepsilon = 1$, как и для предыдущего остова. Оштрафуем вершину 7, полагая $p(7) = 5$; новый кратчайший остов будет тем же, что и на рис. 12 (б). Продолжая далее, штрафуем вершину 6 еще раз, полагая $p(6) = 5$, и опять находим кратчайший остов. Результат показан на рис. 12 (г), из которого видно, что теперь это гамильтонова цепь и, следовательно, по теореме 3 — кратчайшая гамильтонова цепь. Вес этой цепи с первоначальной матрицей весов C_0 равен 258.

Здесь полезно отметить, что если на некотором этапе получается кратчайший остов, уже встречавшийся ранее, то это не означает, что произошло заикливание, и любой такой цикл автоматически устраняется при продолжении алгоритма. (Однако, как будет выяснено ниже, сходимость последовательности кратчайших остовов к кратчайшей гамильтоновой цепи не может быть гарантирована без конкретизации выбираемого способа штрафования вершин.) Мы получили кратчайшую гамильтонову цепь с концевыми вершинами 8 и 9 за 5 итераций, включая 5 вычислений кратчайших остовов, в то время как полный перебор содержал бы $8! = 40320$ вычислений матриц весов и сравнений цепей.

Альтернативой к вышеописанному методу штрафов является следующий. Вместо штрафования только тех вершин, для которых $d_i^T > 2$ (с использованием (13)), можно штрафовать только те вершины (за исключением двух выделенных конечных вершин), степени которых $d_i^T = 1 < 2$, причем величина штрафа $p(i)$ отрицательна и равна выбранной единице шага. С другой стороны, можно применить комбинацию этих двух политик наложения штрафов или воспользоваться совершенно другими политиками, описанными в разд. 7.6.3.4. Здесь следует отметить, что выбор политики наложения штрафов может существенно влиять на число необходимых для получения решения итераций.

Предположим в качестве примера, что на втором этапе описанных вычислений — когда кратчайший остов изображен на рис. 12 (б) — мы решили воспользоваться второй политикой наложения штрафов и вместо того, чтобы положить $p(6) = 5$, взяли $p(10) = -5$. Получившийся кратчайший остов был бы таким, какой показан на рис. 12 (г), т. е. он немедленно дал бы кратчайшую гамильтонову цепь.

Интересно отметить, что не существует единственного вектора штрафов $p(i)$, $i = 1, \dots, n$, для которого матрица весов будет преобразовываться так, что кратчайший остов будет и гамильтоновой цепью и, следовательно, кратчайшей такой цепью. В приведенном

примере окончательное значение вектора штрафов при первой политике такое: $p(1) = p(6) = 10$, $p(7) = 15$, все остальные $p(i) = 0$; в то же время при второй политике, принятой, начиная с шага 2, для окончательного вектора штрафов $p(1) = p(7) = 10$, $p(10) = -5$, все остальные $p(i) = 0$.

7.6.3.2. Решение задачи (а). Мы проиллюстрировали сначала применение алгоритма штрафования вершин к задаче (б), так как к этой задаче непосредственно применима теорема 3. Подобно тому как для решения задачи (б) с помощью алгоритма поиска, использующего дерево решений, она была сведена к задаче (а), так и сейчас, чтобы решить задачу (а) с помощью алгоритма штрафования вершин, мы сведем ее к задаче (б). Это осуществляется посредством следующего приема.

К множеству X вершин графа G добавим еще две вершины, например x_1 и x_2 , и образуем новое множество $X' = X \cup (x_1, x_2)$. К множеству ребер A графа G добавим два множества ребер:

$$S_1 = \bigcup_{j=1}^n \{(x_1, x_j)\} \quad \text{и} \quad S_2 = \bigcup_{j=1}^n \{(x_2, x_j)\},$$

и образуем новое множество $A' = A \cup S_1 \cup S_2$. Для всех $j = 1, \dots, n$ зададим веса a — для ребер (x_1, x_j) и b — для ребер (x_2, x_j) , где a и b — две произвольные постоянные.

Теорема 4. *Решение задачи (б) для графа $G' = (X', A')$ с концевыми вершинами x_1 и x_2 является решением задачи (а) для графа G .*

Доказательство. Любая гамильтонова цепь графа G' с концевыми вершинами x_1 и x_2 может рассматриваться как гамильтонова цепь, проходящая через вершины графа G с добавленными ребрами (x_1, x_i) и (x_2, x_j) , $x_i, x_j \in X$. Таким образом, вес гамильтоновой цепи из G' равен, скажем, $F' = F + a + b$, где F — вес некоторой гамильтоновой цепи из G , и, следовательно, вес F' минимален только тогда, когда F минимален, т. е. когда F соответствует кратчайшей гамильтоновой цепи из G . Теорема доказана.

7.6.3.3. Сходимость метода штрафования вершин. Метод штрафования вершин, описанный выше, был развит Кристофидесом и, в несколько иной форме, примерно в то же время Хелдом и Карпом. Последние авторы показали также, что итерационный метод не обязательно сходится, и в качестве подходящего примера предложили граф, изображенный на рис. 13. Несмотря на этот большой недостаток, метод является тем не менее очень ценным по двум причинам.

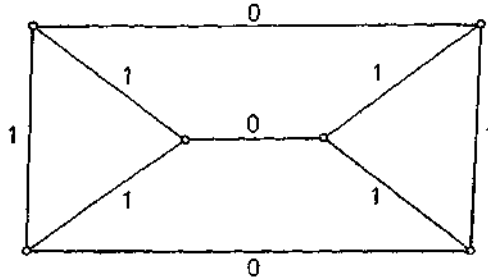


Рис. 13.

Во-первых, в большинстве случаев он сходится (как будет показано ниже, все 33 случайно выбранные задачи были решены этим методом по крайней мере при одной стратегии наложения штрафов).

Во-вторых, допустим, что итерации остановлены в некоторый момент, когда кратчайший остов с модифицированной матрицей есть T' , штрафы вершин $p(i)$, $i = 1, 2, \dots, n$, и степени вершин $d_i^{T'}$; рассматриваем задачу, когда нужно найти кратчайшую гамильтонову цепь с концевыми вершинами x_1 и x_2 . Вес остова T' с модифицированной матрицей равен

$$\bar{F}_{T'} = F_{T'} + \sum_{i \neq 1, 2} p(i) d_i^{T'} + p(1) + p(2),$$

где $F_{T'}$ — вес остова T' , вычисленный для первоначальной матрицы весов.

Если H — кратчайшая гамильтонова цепь графа, то вес цепи H с модифицированной матрицей равен

$$\bar{F}_H = F_H + 2 \sum_{i \neq 1, 2} p(i) + p(1) + p(2).$$

Так как по определению $\bar{F}_{T'} < \bar{F}_H$, то разность

$$f(p) \equiv \bar{F}_H - \bar{F}_{T'} = F_H - [F_{T'} + \sum_{i \neq 1, 2} p(i) (d_i^{T'} - 2)] \quad (14)$$

является мерой близости дерева T' к кратчайшей гамильтоновой цепи H , причем $f(p) = 0$, если $T' = H$.

Величину $f(p)$ можно рассматривать как функцию вектора штрафов $p = (p_i \mid i = 1, \dots, n)$. Хелд и Карп предложили два метода нахождения штрафов p^* , минимизирующих $f(p)$. Один основан на технике преобразования столбцов, а второй является методом наискорейшего спуска. (Другой, несколько более общий последовательный подход, основанный на эвристическом поиске, был успешно применен Камерини и др.) Если метод штрафования вершин сходится, то для

минимального значения $f(\mathbf{p}^*) = 0$ так как $T \rightarrow H$ и $d_i^{T'} = 2$ для всех $i \neq 1, 2$. С другой стороны, как было отмечено выше, метод сходится не всегда, и при отсутствии сходимости минимальное значение $f(\mathbf{p}^*)$ величины $f(\mathbf{p})$ равно некоторому положительному числу. Совершенно очевидно, что в этом случае величина

$$F_{T'} + \sum_{i \neq 1, 2} p^*(i) (d_i^{T'} - 2) \quad (15)$$

дает нижнюю границу веса кратчайшей гамильтоновой цепи. В этом последнем случае получаемая граница является очень точной и может быть использована с большим эффектом в алгоритме поиска, применяющем дерево решений, — в духе того, как это делалось в разд. 7.6.2.1.

7.6.3.4. Стратегии наложения штрафов. Хотя, как уже отмечалось в разд. 7.6.3.3, существуют два метода для определения последовательности штрафов, вынуждающих кратчайший остов становиться кратчайшей гамильтоновой цепью (если только вообще сходимость возможна), оба эти метода являются слишком изысканными и требуют больших затрат времени. В настоящем разделе мы экспериментально исследуем влияние различных стратегий штрафования вершин на скорость сходимости метода. Для всех 33 случайно выбранных полных графов, использованных для проверки стратегий, по крайней мере одна стратегия приводила к кратчайшей гамильтоновой цепи. Это убеждает нас в том, что, хотя и существуют задачи, в которых алгоритм штрафования вершин не сходится (на самом деле любой связный граф без гамильтонова цикла приводит нас к такой ситуации), но все же в большинстве задач, при составлении которых не старались специально найти контрпримеры, будет иметь место сходимость.

(I) *Фиксированные штрафы*

(а) *Только положительные штрафы*

Это тот случай, когда все вершины x_i дерева T со степенями $d_i^T > 2$ штрафуются на величину $\gamma \cdot (d_i^T - 2)$, а все остальные вершины не штрафуются. Было найдено, что для малых γ число необходимых для достижения решения итераций (т. е. определения некоторого гамильтонова цикла) приблизительно обратно пропорционально величине γ , но для достаточно больших γ вполне возможно заикливание и решение получено не будет. Наилучшее значение γ зависит как от числа вершин, так и от распределения весов ребер.

(б) *Только отрицательные штрафы*

Это тот случай, когда вершины x_i (за исключением двух отмеченных концевых вершин) штрафуются на величину $-\gamma$, если $d_i^T = 1$, а

остальные не штрафуются. Результаты примерно те же, что и в случае (а).

(с) *Положительные и отрицательные штрафы*

В этом случае вершины x_i штрафуются в соответствии с (а) и (б), в зависимости от того, будет ли $d_i^T > 2$ или $d_i^T = 1$ соответственно. Было найдено, что значение γ , равное половине используемого в случаях (а) и (б), обеспечивает сходимость примерно за то же самое число итераций, что и в случаях (а) и (б). Этот метод в общем лучше методов (а) и (б).

(II) *Последовательно уменьшаемые штрафы*

При использовании фиксированных штрафов может возникнуть ситуация, когда вершина x_i с $d_i^T = 3$ (например), оштрафованная в некоторый момент на γ , превратится в вершину с $d_i^T = 1$ после следующей итерации, иными словами, вершина x_i «сверхоштрафована». Если используется смешанная стратегия (I (с)), то на следующей итерации вершина будет оштрафована на величину $-\gamma$, и это может привести к тому (не обязательно), что d_i^T снова станет равным 3. Из-за этих колебаний величины d_i^T может потребоваться неоправданно большое число итераций.

Было найдено, что в такой ситуации для большинства графов лучший метод состоит в уменьшении ранее наложенного штрафа, так что на следующей итерации для вышеприведенного примера следовало бы оштрафовать вершину x_i на $-\alpha\gamma$ вместо $-\gamma$, где $0 < \alpha < 1$.

Ситуация, аналогичная описанной, возникает и тогда, когда вершина со степенью $d_i^T = 1$ сверхштрафуются на отрицательную величину, в результате чего после очередной итерации она получает $d_i^T > 2$. В этом случае поступают как и в предыдущем.

В численных экспериментах, результаты которых приведены в табл. 2, было использовано значение $\alpha = 0,5$.

Вычислительные результаты † в методе штрафования вершин

Число вершин n	Стратегия штрафования													
	I(a)		I(b)		I(c)		II		III(a)		III(b)		III(c)	
	α	β	α	β	α	β	α	β	α	β	α	β	α	β
10	22	0,15	16	0,13	18	0,11	16	0,18	71	0,18	27	0,16	9	0,12
16	39	0,27	33	0,24	35	0,29	31	0,20	52	0,42	18	0,39	22	0,50
20	83	0,88	56	0,71	74	0,80	39	0,38	55	0,59	24	0,48	27	0,67
26	126	1,83	158	1,41	98	1,66	74	1,02	40	0,47	60	0,71	16	0,46
30	—	—	—	—	—	—	—	—	59	0,91	63	1,11	61	0,95
36									98	1,41	87	1,63	72	1,37
40									172	2,23	159	1,99	57	1,00
46									—	—	—	—	81	3,09
50													92	4,43
56													146	9,70
60													111	13,6

α Число итераций

β время вычисления (СВС 6600)

† Каждый элемент является средним значением для 3 графов одного и того же размера

(III) Вычисляемые штрафы

*

(а) Только положительные штрафы

В этом случае все вершины x_i с $d_i^T > 2$ штрафуются на $p(i)$, где $p(i)$ вычисляется как минимальный положительный штраф, который, будучи применен к вершине x_i (изолированно), уменьшает d_i^T на одну единицу после одной итерации. Штрафы $p(i)$ применяются тогда одновременно — каждый к своей вершине.

Величина $p(i)$ для заданной вершины x_i вычисляется следующим образом. Удалим из дерева $T = (X, A_T)$ только одно из ребер (x_i, x_r) , инцидентных вершине x_i . Это приведет к распадению T на два поддерева T_1 и T_2 . Найдем ребро наименьшего веса, соединяющее эти два поддерева, т. е. найдем ребро (x_j^r, x_k^r) с весом

$$c(x_j^r, x_k^r) = \min_{\substack{x_j \in T_1 \\ \neq x_i}} \min_{\substack{x_k \in T_2 \\ \neq x_i}} [c(x_j, x_k)], \quad (16)$$

где x_j^r и x_k^r — оптимальные значения x_j и x_k , соответственно, а T_1 и T_2 обозначают как деревья, так и соответствующие множества их вершин. Вес $c(x_j^r, x_k^r)$ является поэтому наименьшим весом «связи» поддеревьев T_1 и T_2 в одно дерево, когда удалено первоначальное ребро (x_i, x_r) . Таким образом, штраф $c(x_j^r, x_k^r) - c(x_i, x_r)$, налагаемый на вершину

x_i является наименьшим штрафом, который удалил бы ребро (x_i, x_r) из кратчайшего остова при следующей итерации. Поэтому если штраф выбран как

$$p(i) = \min_{(x_j, x_r) \in A_T} [c(x_j^r, x_k^r) - c(x_i, x_r)], \quad (17)$$

то штраф $p(i)$, наложенный только на вершину x_i , удалил бы (в предположении, что $p(i)$ является единственным) только одно ребро (x_i, x^*) из множества ребер, инцидентных x_i , т. е. уменьшил бы d_i^T на единицу. Ребро (x_i, x^*) является как раз тем ребром, которое минимизирует выражение (17).

Здесь следует отметить, что когда все штрафы $p(i)$, вычисленные по (17), накладываются одновременно каждый на свою вершину, степени некоторых вершин могут уменьшиться более чем на 1, в то время как степени других вершин могут вообще не уменьшиться (или могут даже увеличиться). Это объясняется совместным влиянием штрафов вершин на веса ребер при следующей итерации. Таким образом, хотя использование штрафов, найденных из (17), не гарантирует получение ответа после заданного числа итераций, но, как было обнаружено (см. табл. 2), этот метод штрафования лучше, чем методы (I) и (II).

(б) *Только отрицательные штрафы*

Это тот случай, когда все вершины x_i с $d_i^T = 1$ штрафуются на отрицательную величину $p(i)$, где $p(i)$ вычисляется как максимальный (наименьший отрицательный) штраф, который, будучи применен только к вершине x_i , сделает степень d_i^T равной 2 после одной итерации, т. е. добавит второе ребро к вершине x_i . Однако штрафы $p(i)$ применяются одновременно — каждый к своей вершине.

Значение $p(i)$ для данной вершины x_i вычисляется следующим образом. Добавим ребро (x_i, x_r) к дереву T . Это добавление приведет к замыканию цикла, составленного из ребра (x_i, x_r) вместе с ребрами дерева T , входящими в цепь от x_r до x_i . Пусть множество ребер дерева T в цепи от x_r до x_i (исключая последнее ребро, инцидентное x_i) будет S_{ri} . Если тогда добавить ребро (x_i, x_r) к ребрам T и удалить одно какое-либо ребро из S_{ri} , то получится другое дерево, в котором степень вершины x_i равна двум. Если, таким образом, накладывается единственный штраф $p(i)$ на вершину x_i , где

$$p(i) = \min_{x_r \in X} [c(x_i, x_r) - \min_{(x_j, x_k) \in S_{ri}} \{c(x_j, x_k)\}], \quad (18)$$

(т. е. $p(i)$ равен наименьшему дополнительному весу добавления ребра от x_i к какой-нибудь другой вершине x_r и удаления ребра наименьшего

веса в цепи от x_r к x_i), то степень вершины x_i станет равной 2 после единственной итерации.

Если все штрафы $p(i)$ накладываются одновременно, то, как и в случае (III (а)), взаимодействие этих штрафов делает невозможным предсказание изменения степеней вершин после одной итерации.

(с) *Положительные и отрицательные штрафы*

Эта стратегия заключается в наложении штрафов на вершины в соответствии с (III (а)) или (III (б)) в зависимости от того, будет ли $d_i^T > 2$ или $d_i^T = 1$.

В таблице 2 приводятся результаты и делается сравнение семи стратегий штрафования, описанных выше. Графы в этой таблице были получены случайным выбором n точек в квадрате при равномерном распределении, а в качестве весов ребер собралось евклидово расстояние между i и i . Таблица дает числа итераций и времена вычислений (ЭВМ CDC 6600, время в секундах), необходимые для получения ответа. Из этой таблицы видно, что действие всех стратегий штрафования зависит от задачи, хотя можно заметить, что наилучшей является стратегия (III (с)). Используя эту стратегию, можно найти кратчайшую гамильтонову цепь в графе примерно с 60 вершинами менее чем за 15 с.

На рис. 14 для некоторого графа с 24 вершинами показаны начальный кратчайший остов и окончательная кратчайшая гамильтонова цепь.

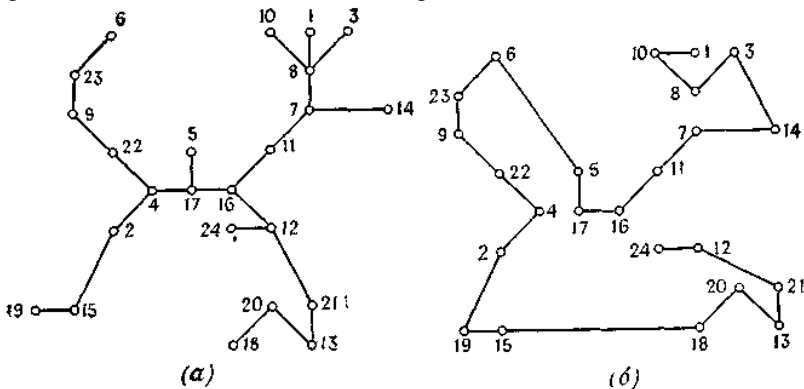


Рис. 14. Задача с 24 вершинами. (а) Начальный кратчайший остов. (б) Кратчайший гамильтонов цикл, полученный после 40 итерации с использованием штрафов III(с).

Первая и последняя пронумерованные вершины являются отмеченными концевыми вершинами.

7.6.4. Задачи, родственные задаче коммивояжера

До сих пор в разд. 7.6 мы имели дело с «открытой» задачей коммивояжера, т. е. с кратчайшей гамильтоновой цепью (а не с циклом). Вначале это было оправдано тем, что при этом мы имели дело с более прямой связью «открытой» задачи с кратчайшим остовом. С другой стороны, нужно сделать только очень небольшие изменения, чтобы подойти и к «замкнутой» задаче коммивояжера. Хелд и Карп, например, ввели понятие кратчайшего 1-дерева графа G , причем оно определяется как кратчайший остов порожденного подграфа графа G с удаленной вершиной 1 плюс два кратчайших ребра, исходящих из вершины 1 к двум другим вершинам дерева. Очевидно, что между понятием кратчайшего 1-дерева и «замкнутой» задачей коммивояжера существует та же самая связь, что и между понятием кратчайшего остова и «открытой» задачей. Таким образом, штрафование вершин изменяет относительные веса 1-деревьев, но оставляет инвариантным относительное упорядочение гамильтоновых циклов. Так же совершенно очевидно, что, как в «открытой» задаче, кратчайший остов со всеми вершинами степени 2 (за исключением двух концевых вершин) становится кратчайшей гамильтоновой цепью между этими концевыми вершинами, так и в «замкнутой» задаче кратчайшее 1-дерево, все вершины которого имеют степень 2, является кратчайшим гамильтоновым циклом графа. Алгоритм штрафования вершин, обсуждавшийся ранее в данном разделе, может быть использован поэтому фактически без изменений и для решения «замкнутой» задачи коммивояжера.

7.7. Задача коммивояжера и задача о назначениях

В разд. 7.5 было отмечено, что задача о назначениях, определяемая соотношениями (4), (5) и (6), может иметь решения, образованные некоторым числом непересекающихся циклов, и что соответствующий метод решения задачи коммивояжера требует наложения ограничений до тех пор, пока решение не будет состоять из единственного (гамильтонова) цикла. В настоящем разделе мы исследуем процедуры наложения этих ограничений в рамках алгоритма поиска, использующего дерево решений. Применяемая схема будет в основном такой же, как и в разд. 7.6.2 для алгоритма поиска, основанного на рассмотрении кратчайшего остова.

7.7.1. Алгоритм поиска, использующий дерево решений

Пусть решение задачи о назначениях с матрицей весов $[c_{ij}]$ (где $c_{ij} = \infty, \forall i$) образовано некоторым числом непересекающихся циклов. Так, например, если решение задачи с 8 вершинами имеет вид $\xi_{12} = \xi_{26} = \xi_{35} = \xi_{47} = \xi_{54} = \xi_{61} = \xi_{78} = \xi_{83} = 1$ и $\xi_{ij} = 0$ для всех остальных пар (i, j) , то решение, отвечающее двум циклам, изображено на рис. 15 (а).

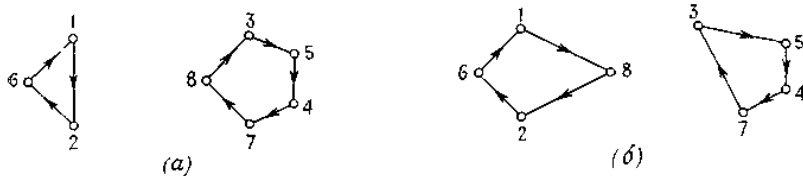


Рис. 15. Решение задач о назначениях. (а) Решение задачи P_o . (б) Решение задачи P_1 .

Теперь необходимо сделать следующее — исключить данное решение вместе со многими другими такими же решениями, не потеряв решение задачи коммивояжера с той же матрицей весов. Так как решение задачи коммивояжера является гамильтоновым циклом, то мы будем пытаться исключить любое решение, состоящее более чем из одного цикла.

(А) Правило простого ветвления. Пусть в общем случае решение задачи о назначениях содержит (не гамильтонов) цикл $(x_1, x_2, \dots, x_k, x_1)$ длины k . Удаление этого цикла (и всех решений, его содержащих) из дальнейшего рассмотрения может быть произведено путем наложения требования, что по крайней мере одна из дуг $(x_1, x_2), (x_2, x_3), \dots, (x_k, x_1)$ не должна входить в решение. Это можно сделать просто, если первоначальную задачу с матрицей весов $[c_{ij}]$ разбить на k подзадач P_1, P_2, \dots, P_k . В задаче P_1 $c(x_1, x_2)$ полагается равным ∞ , а все остальные c_{ij} остаются без изменения (т. е. такими же, как и в задаче P_o), в задаче P_2 $c(x_2, x_3) = \infty$ и вообще в задаче P_k $c(x_k, x_1) = \infty$. Очевидно, что решение задачи P_o , не содержащее цикла $(x_1, x_2, \dots, x_k, x_1)$, является решением по крайней мере одной из задач P_1, \dots, P_k и, следовательно, оптимальное решение задачи коммивояжера является решением одной или нескольких из этих подзадач.

Так, например, исключая на рис. 15 (а) цикл длины 3, получим дерево решений, изображенное на рис. 16, в котором задачи P_1, P_2 и P_3 представлены узлами дерева, растущего из начальной задачи P_o .

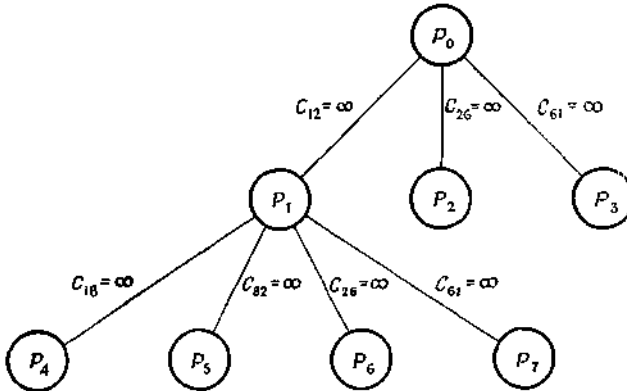


Рис. 16. Дерево решений с простым правилом ветвления А.

Пусть задачи P_1 , P_2 и P_3 решены как задачи о назначениях, и пусть соответствующие веса будут C_1 , C_2 и C_3 . Так как C_1 является нижней границей в задаче коммивояжера для P_1 , а также для P_2 и P_3 , то число $L = \min(C_1, C_2, C_3)$ является нижней границей для веса решения первоначальной задачи коммивояжера.

Пусть, скажем, $L = C_1$ (т. е. $C_1 \leq C_2$ и $C_1 \leq C_3$). Если решение задачи P_1 является гамильтоновым циклом, то оно будет решением первоначальной задачи коммивояжера. В противном случае пусть оно будет, скажем, таким, как на рис. 15 (б). Исключая цикл (1, 8, 2, 6, 1), мы снова составляем и решаем подзадачи P_4 , P_5 , P_6 и P_7 , показанные на рис. 16. Из этого рисунка видно, что P_4 отвечает задаче, для которой в матрице весов элемент c_{18} положен равным ∞ (в дополнение к тому, что элемент c_{12} положен равным ∞ уже ранее). То же самое относится и к P_5 , P_6 и P_7 . Нижняя граница теперь переопределяется как

$L = \min(C_2, C_3, C_4, C_5, C_6, C_7)$. Допустим, что задача P_3 соответствует весу L (т. е. $L = C_3$). Если решение задачи P_3 является гамильтоновым циклом, то оно будет решением первоначальной задачи коммивояжера. Дальнейшее ветвление в узле P_3 , с другой стороны, должно осуществляться точно так же, как это делалось в P_1 , и этот процесс продолжается до тех пор, пока задача с текущим значением веса L не

будет иметь в качестве решения гамильтонов цикл. Когда это произойдет, то полученный цикл будет окончательным решением, так как его вес (по определению числа L) не превосходит нижних границ весов любых других гамильтоновых циклов, которые могут появиться при дальнейшем ветвлении в оставшихся узлах дерева.

Из приведенного описания алгоритма становится очевидным, что примененный поиск с помощью дерева решений относится к типу с «приоритетом по ширине», как это объяснено в приложении 1.

(Б) Правило исключаящего ветвления. Как объяснено в приложении 1, для хорошего ветвления при разбиении задачи P_o на подзадачи P_1 , P_2 и P_3 требуется только, чтобы каждое возможное решение задачи P_o (за исключением удаляемых решений) было решением по крайней мере одной из подзадач. Однако, как отмечено в приложении, очень желательным свойством метода ветвления было бы разбиение возникающих подзадач на взаимно исключаящие, коль скоро речь идет о возможных решениях задачи P_o . Иными словами, каждое допустимое решение задачи P_k должно быть решением одной, и только одной из этих подзадач.

Ранее описанное правило ветвления основывалось на том факте, что цикл, такой, как $(x_1, x_2, \dots, x_k, x_1)$, может быть удален посредством исключения одной из его дуг. Это, однако, не приводит в процессе ветвления к взаимно исключаящим подзадачам. Так, в вышеприведенном примере решение задачи P_o , соответствующее циклам $(1, 3, 6, 1)$ и $(2, 5, 4, 7, 8, 2)$, является допустимым решением как подзадачи P_1 ($c_{12} = \infty$), так и P_2 ($c_{26} = \infty$).

Другим правилом ветвления, удаляющим цикл $(x_1, x_2, \dots, x_k, x_1)$ и приводящим к взаимно исключаящим подзадачам, является следующее:

для задачи P_1 положить $c(x_1, x_2) = \infty$;

для задачи P_2 положить $c(x_1, x_2) = -M$ и $c(x_2, x_3) = \infty$;

для задачи P_3 положить $c(x_1, x_2) = c(x_2, x_3) = -M$ и $c(x_3, x_k) = \infty$.

.....
 для задачи P_k положить $c(x_1, x_2) = c(x_2, x_3) = \dots =$
 $= c(x_{k-1}, x_k) = -M$ и $c(x_k, x_1) = \infty$,

где $-M$ — достаточно большое отрицательное число, гарантирующее, что дуга, вес которой равен $-M$, входит в оптимальное решение.

Это правило ветвления наверняка приводит к взаимно исключаящим подзадачам, так как для любых двух подзадач имеется по крайней мере одна дуга, исключенная из решения одной, но заведомо входящая в решение другой подзадачи. Легко также видеть, что ни одно возможное решение задачи P_o не будет потеряно, т. е. что любое решение первоначальной задачи P_o будет являться решением одной из

подзадач. Это очевидно в силу того, что любое решение задачи P_0 имеет некоторую последовательность дуг, выходящую из x_l , таких, как (x_l, x_a) , (x_a, x_b) и т. д., и первые r дуг должны совпадать с дугами цепи $(x_l, x_2, x_3, \dots, x_k)$ при некотором значении $r = 0, 1, \dots, k$. Значение $r = 0$ отвечает тому случаю, когда совпадения вообще нет, $r = 1$ — случаю, когда $x_a = x_2$, но $x_b \neq x_3$ и т. д.

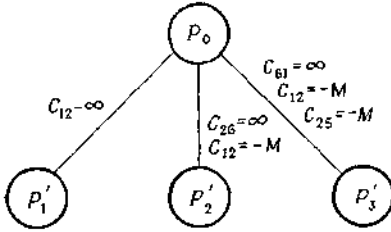


Рис. 17. Дерево решений с исключающим правилом ветвления Б.

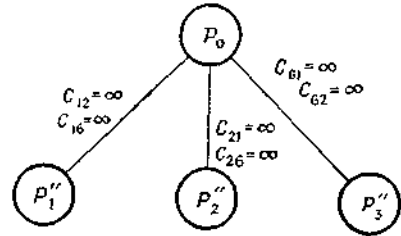


Рис. 18. Дерево решений с улучшенным правилом ветвления В.

В приведенном примере первоначальная задача P_0 разбивалась бы на три подзадачи, показанные на рис. 17 (ср. с разбиением на первом уровне из рис. 16, полученным по предыдущему правилу ветвления).

(В) Лучшее правило ветвления. Оба предыдущих правила ветвления исключали (при каждом отдельном ветвлении) все решения, содержащие заданный цикл, такой, как $(x_l, x_2, \dots, x_k, x_l)$. Очевидно, однако, что в решении задачи коммивояжера не только не должен существовать такой цикл, но должна быть по крайней мере одна дуга, ведущая из множества вершин $S = \{x_l, \dots, x_k\}$ в множество вершин $\bar{S} = X - S$. Действительно, существование какой-либо дуги, ведущей из S в \bar{S} , не только гарантирует исключение решений, содержащих цикл, принадлежащий множеству S , но также позволяет исключить решения, множества вершин которых лежат в S и которые состоят из нескольких циклов (а не только из одного). Таким образом, можно ожидать, что правило ветвления, основанное на требовании, чтобы существовала некоторая дуга из S в \bar{S} , будет равномерно лучше, чем два предыдущие правила ветвления 2.

Так как дуга из S в \bar{S} должна начинаться в некоторой вершине из S , то задачу можно расщепить на k подзадач. В подзадаче P_i мы будем требовать, чтобы начальной вершиной дуги являлась $x_i \in S$, а конечной вершиной была некоторая вершина в \bar{S} . Это можно сделать, положив

$c(x_i, x_j) = \infty$ для всех $x_i \in S$ и оставив без изменения все другие веса. В решении получившейся задачи о назначениях мы наверняка бы имели дугу из x_i , ведущую в \bar{S} , так как для всех других альтернатив их веса должны быть положены равными ∞ .

В примере, данном выше, начальная задача P_0 была бы в соответствии с настоящим правилом ветвления подразделена на три подзадачи, определенные, как показано на рис. 18. Дальнейшие ветвления происходят аналогично.

7.7.2. Пример

Найти решение задачи коммивояжера с матрицей весов

		1	2	3	4	5	6	7	8
$C_A =$	1	∞	76	43	38	51	42	19	80
	2	42	∞	49	26	78	52	39	87
	3	48	28	∞	36	53	44	68	61
	4	72	31	29	∞	42	49	50	38
	5	30	52	38	47	∞	64	75	82
	6	66	51	83	51	22	∞	37	71
	7	77	62	93	54	69	38	∞	26
	8	42	58	66	76	41	52	83	∞

Теперь мы используем только что описанный алгоритм исключения циклов, беря на каждом шаге циклы наименьшей длины и используя правило ветвления В.

Решение задачи о назначениях с матрицей C_A состоит из двух циклов (2, 4, 3) и (1, 7, 8, 6, 5) с весом 232. Это соответствует узлу А дерева решений, показанного на рис. 19.

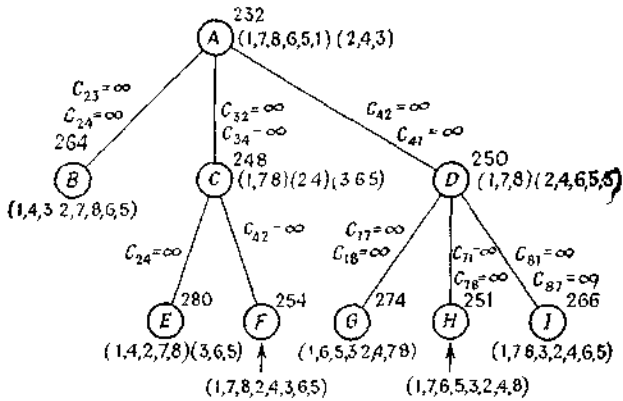


Рис. 19. Поиск, использующий дерево решений из примера 7.7.2.

Исключение цикла (2, 4, 3) по правилу ветвления В приводит к трем подзадачам, изображаемых на рис. 19 узлами B, C и D. Матрицы весов этих подзадач, решения задач о назначениях для них и веса этих решений равны соответственно

	1	2	3	4	5	6	7	8
1	∞	76	43	38	51	42	19	80
2	42	∞	∞	∞	78	52	39	87
3	48	28	∞	36	53	44	68	61
4	72	31	29	∞	42	49	50	38
5	30	52	38	47	∞	64	75	82
6	66	51	83	51	22	∞	37	71
7	77	62	93	54	69	38	∞	26
8	42	58	66	76	41	52	83	∞

$C_B =$

(1, 4, 3, 2, 7, 8, 6, 5)
264

	1	2	3	4	5	6	7	8
1	∞	76	43	38	51	42	19	80
2	42	∞	49	26	78	52	39	87
3	48	∞	∞	∞	53	44	68	61
4	72	31	29	∞	42	49	50	38
5	30	52	38	47	∞	64	75	82
6	66	51	83	51	22	∞	37	71
7	77	62	93	54	69	38	∞	26
8	42	58	66	76	41	52	83	∞

$C_C =$

(1, 7, 8), (2, 4), (3, 6, 5)
248

Решение подзадачи В является гамильтоновым циклом с весом 264. Однако нижние границы в обоих узлах С и D меньше этого значения, так что существует возможность получить лучшее решение, продолжая процесс ветвления.

	1	2	3	4	5	6	7	8
1	∞	76	43	38	51	42	19	80
2	42	∞	49	26	78	52	39	87
3	48	28	∞	36	53	44	68	61
4	72	∞	∞	∞	42	49	50	38
5	30	52	38	47	∞	64	75	82
6	66	51	83	51	22	∞	37	71
7	77	62	93	54	69	38	∞	26
8	42	58	66	76	41	52	83	∞

$C_D =$

(1, 7, 8), (2, 4, 6, 5, 3)
250

Наименьшая из нижних границ есть $L = \min(248, 250) = 248$, и она соответствует узлу C , поэтому процесс ветвления следует продолжать с этого узла, исключая цикл (2, 4). Теперь возникают две другие подзадачи, показанные узлами E и F на рис. 19. Их матрицы весов, решения задач о назначениях и веса таковы:

	1	2	3	4	5	6	7	8
1	∞	76	43	38	51	42	19	80
2	42	∞	49	∞	78	52	39	87
3	48	∞	∞	∞	53	44	68	61
4	72	31	29	∞	42	49	50	38
5	30	52	38	47	∞	64	75	82
6	66	51	83	51	22	∞	37	71
7	77	62	93	54	69	38	∞	26
8	42	58	66	76	41	52	83	∞

$C_E =$

(3, 6, 5), (1, 4, 2, 7, 8)
280

	1	2	3	4	5	6	7	8		
$C_F =$	1	∞	76	43	38	51	42	19	80	(1, 7, 8, 2 4 3, 6, 5) 254
	2	42	∞	49	26	78	52	39	87	
	3	48	∞	∞	∞	53	44	68	61	
	4	72	∞	29	∞	42	49	50	38	
	5	30	52	38	47	∞	64	75	82	
	6	66	51	83	51	22	∞	37	71	
	7	77	62	93	54	69	38	∞	26	
	8	42	58	66	76	41	52	83	∞	

Следует отметить, что на этом этапе решение подзадачи F является гамильтоновым циклом с весом 254. Этот вес меньше, чем вес предыдущего лучшего решения (с весом 264), и, следовательно, лучшим текущим решением будет решение подзадачи F . Два узла (E и D) все еще дают решения с негамильтоновыми циклами и являются поэтому кандидатами на продолжение процесса ветвления. Однако для узла E нижняя граница равна $280 > 254$ (текущее значение наилучшего решения), и, значит, ветвление в этом узле не может привести к

лучшему

результату.

	1	2	3	4	5	6	7	8
1	∞	76	43	38	51	42	∞	∞
2	42	∞	49	26	78	52	39	87
3	48	28	∞	36	53	44	68	61
4	72	∞	∞	∞	42	49	50	38
5	30	52	38	47	∞	64	75	82
6	66	51	83	51	22	∞	37	71
7	77	62	93	54	69	38	∞	26
8	42	58	66	76	41	52	83	∞

$C_0 =$

(1, 6, 5, 3, 2, 4, 7, 8)
274

Поэтому узел D с нижней границей $250 < 254$ является единственным узлом, ветвление в котором может привести к улучшению. Исключение цикла (1, 7, 8) из решения для узла D приводит к трем подзадачам, обозначенным на рис. 19 как G , H и I . Соответствующие этим подзадачам матрицы весов и решения задач о назначениях таковы:

s

	1	2	3	4	5	6	7	8
1	∞	76	43	38	51	42	19	80
2	42	∞	49	26	78	52	39	87
3	48	28	∞	36	53	44	68	61
4	72	∞	∞	∞	42	49	50	38
5	30	52	38	47	∞	64	75	82
6	66	51	83	51	22	∞	37	71
7	∞	62	93	54	69	38	∞	∞
8	42	58	66	76	41	52	83	∞

$C_H =$ (1,7,6,5,3,2,4,8)
251

	1	2	3	4	5	6	7	8
1	∞	76	43	38	51	42	19	80
2	42	∞	49	26	78	52	39	87
3	48	28	∞	36	53	44	68	61
4	72	∞	∞	∞	42	49	50	38
5	30	52	38	47	∞	64	75	82
6	66	51	83	51	22	∞	37	71
7	77	62	93	54	69	38	∞	26
8	∞	58	66	76	41	52	∞	∞

$C_I =$ (1,7,8,3,2,4,6,5)
266

На этом этапе мы замечаем, что решение подзадачи H является гамильтоновым циклом с весом 251, что меньше чем предыдущее. лучшее значение 254. Поэтому решение подзадачи H , т. е.

(1, 7, 6, 5, 3, 2, 4, 8), заменяет предыдущее лучшее решение. Кроме того, значение 251 меньше, чем нижняя граница в любом конечном узле дерева, и, следовательно, найдено оптимальное решение всей задачи. (Решения во всех концевых узлах — за исключением узла E ,

ветвление в котором было прекращено ранее,— являются на самом деле гамильтоновыми циклами, и здесь безотносительно к границам весов этих решений не нужно бы было производить дальнейшее ветвление.)

7.7.3. Вычислительные комментарии и характеристики

Только что описанный алгоритм поиска, использующий дерево решений и основанный на исключении циклов, зависит от нахождения решений задач о назначениях с очень незначительно измененными матрицами весов. Здесь следует отметить, что каждая из этих задач может быть решена очень просто, если хранить решение предыдущей задачи, из которой она возникла. В частности, модификации, производимые в случае всех трех ранее упомянутых правил ветвления, производятся с помощью приравнивания ∞ некоторого элемента $c(x_i, x_j)$ для дуги (x_i, x_j) , входящей в решение текущей задачи о назначениях.

В венгерском алгоритме решения задачи о назначениях элемент $c(x_i, x_j)$ в окончательной относительной матрице весов должен был бы иметь значение 0, а соответствующий маркер показывает, что в решении было назначение. Изменение значения элемента $c(x_i, x_j)$ привело бы, очевидно, к перераспределению этого назначения, но оставило бы без изменения другие $n - 1$ назначений. Таким образом, начиная с решения (назначений) задачи — до того как были произведены модификации матрицы $[c_{ij}]$ — и удаляя упомянутое назначение, можно получить решение новой модифицированной задачи, повторно применяя венгерский алгоритм на последнем шаге, так как единственный «прорыв», т. е. единственное увеличение числа ненулевых назначений с $n - 1$ до n , дал бы в действительности оптимальное решение новой задачи о назначениях.

Рабочие качества вышеописанного алгоритма поиска с исключением циклов по правилу ветвления В исследовали Беллмор и Мэлоун. На любой стадии ветвление в узле продолжалось так, чтобы исключить циклы наименьшей длины в решении задачи о назначениях, соответствующей этому узлу. Было показано, что два других правила ветвления обладают худшими характеристиками, особенно в задачах, графы которых содержат такие «скопления» вершин, что дуги в одном и том же скоплении имеют малые веса, а дуги между вершинами из разных скоплений имеют большие веса.)

Тем не менее вышеописанный алгоритм работает не слишком хорошо в симметричных задачах, так как в них решения задач о назначениях обычно состоят из большого числа циклов длины 2 и для их

исключения требуется очень много ветвлений. Еще хуже обстоит дело тогда, когда задача коммивояжера решается на графе, который не содержит гамильтоновых циклов конечного веса.

7.7.4. Лучшие границы для дерева поиска

Мы описали в разд. 7.7.1 алгоритм поиска, в котором в качестве нижней границы в узле берется значение из решения соответствующей задачи о назначениях. На самом-то деле совершенно очевидно, что метод исключения циклов остается неизменным независимо от того, какая нижняя граница используется для ограничения поиска. Но так как процесс ветвления в узле прекращается только тогда, когда или решение подзадачи, отвечающей этому узлу, является гамильтоновым циклом, или когда нижняя граница в узле превосходит значение полученного до этого наилучшего решения, то очевидно, что качество оценки границы оказывает заметное влияние на число ветвлений в дереве решений и, следовательно, на вычислительную эффективность метода. Целью настоящего раздела является описание метода получения нижней границы, вычисляемой по решению задачи о назначениях с небольшими дополнительными усилиями и являющейся более точной, чем ранее используемые границы. Пусть решение задачи о назначениях содержит некоторое число циклов, как это показано в качестве примера на рис. 20 (а).

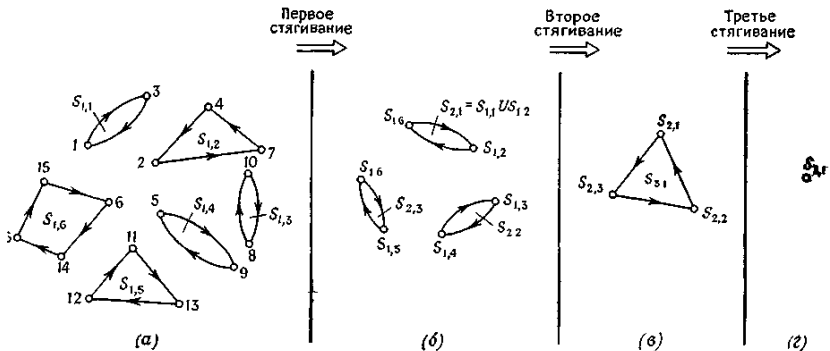


Рис. 20. Процесс стягивания. Первое стягивание, второе стягивание, третье стягивание.

Пусть i -й цикл обозначается $S_{1,i}$ и пусть число циклов равно n_1 . (Мы будем использовать один и тот же символ $S_{1,i}$ как для обозначения цикла, так и для обозначения множества его вершин.)

Определим *стягивание* как замену цикла единственной вершиной. Полученный граф содержит n_1 вершин $S_{1,i}$ ($i = 1, 2, \dots, n_1$). В качестве матрицы весов нового графа берется $(n_1 \times n_1)$ -матрица $C_1 \equiv [c_1(S_{1,i}, S_{1,j})]$ с элементами вида

$$c_1(S_{1,i}, S_{1,j}) = \min_{\substack{k_i \in S_{1,i} \\ k_j \in S_{1,j}}} [f_1(k_i, k_j)], \quad (19)$$

(где $F_1 = [f_1(k_i, k_j)]$ — окончательная матрица относительных весов, полученная в конце решения задачи венгерским, (например) методом.

На рис. 20 (а), например,

$$c_1(S_{1,5}, S_{1,6}) = \min_{\substack{k_5 \in \{11, 12, 13\} \\ k_6 \in \{6, 14, 15, 16\}}} [f_1(k_5, k_6)].$$

Теперь второе решение задачи о назначениях в случае задачи с матрицей C_1 все еще может содержать циклы, в качестве вершин которых выступают предыдущие циклы. Рис. 20 (б) показывает один из возможных способов образования новых циклов $S_{2,i}$ ($i = 1, 2, \dots, n_2$), где n_2 — их общее число (на рис. 20 (б) $n_2 = 3$). Эти циклы опять могут быть стянуты в вершины и дать новую задачу с новой матрицей весов $C_2 \equiv [c_2(S_{2,i}, S_{2,j})]$, вычисленной так же, как и в (19), т. е

$$c_2(S_{2,i}, S_{2,j}) = \min_{\substack{k_i \in S_{2,i} \\ k_j \in S_{2,j}}} [f_2(k_i, k_j)], \quad (20)$$

где множества $S_{2,i}$ являются объединениями всех множеств $S_{1,i}$ образующих «частные» циклы, и где $F_2 = [f_2(k_i, k_j)]$ — матрица относительных весов, полученная в конце решения второй задачи о назначениях

Решение задачи о назначениях для нового, дважды стянутого, графа все еще может содержать циклы, и итерационный процесс решения — стягивания можно продолжать до тех пор, пока задача не сведется к единственной вершине.

Определим *компрессию* как преобразование матрицы, не удовлетворяющей аксиоме треугольника в метрическом пространстве, в матрицу, удовлетворяющую этой аксиоме. Таким образом, для компрессии матрицы нужно заменить каждый элемент m_{ij} для которого

$$m_{ij} > m_{ik} + m_{kj} \quad (\text{при некотором } k),$$

элементом $\min_k [m_{ik} + m_{kj}]$ и продолжать эту замену до тех пор, пока

для всякого k не будут справедливы неравенства $m_{i,j} \leq m_{i,k} + m_{k,j}$.

Теорема 5. Сумма значений решений задач о назначениях, полученных во время процесса «решение — стягивание — компрессия» осуществляемого вплоть до момента, когда «стянутая» задача будет содержать единственную точку), является точной нижней границей для задачи коммивояжера.

Доказательство. Ранее было показано, что элемент (i,j) в матрицеотносительных весов, полученной в конце решения задачи о назначениях, дает нижнюю границу для дополнительного веса, который получается из-за включения дуги (i,j) в решение.

Рассмотрим цикл $S_{l,i}$, полученный в конце решения первой задачи о назначениях. Любой гамильтонов цикл, проходящий через все n вершин, будет иметь по крайней мере две дуги (одну направленную внутрь цикла $S_{l,i}$ другую — наружу), инцидентные вершинам, не лежащим в $S_{l,i}$ и «приходящие» в одну или более вершин из $S_{l,i}$. Число таких дуг будет, очевидно, четным.

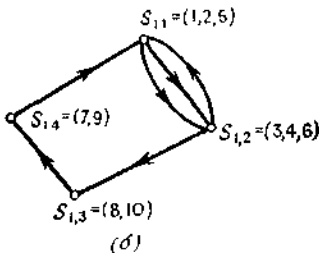
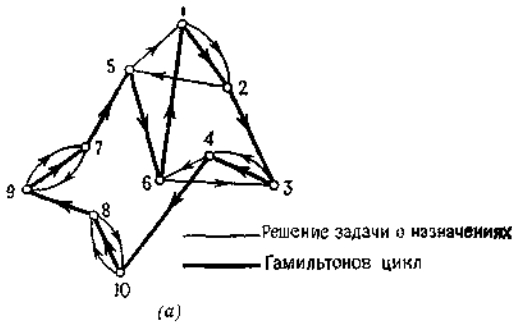


Рис. 21. Преобразование гамильтоновых циклов при стягивании.
(а) Решение задачи о назначениях. (б) Гамильтонов цикл.

На рис. 21 (а) гамильтонов цикл показан жирной линией, в то время как решение задачи о назначениях показано тонкой линией. После

стягивания получается граф, изображенный на рис. 21 (б), с двумя дугами, инцидентными $S_{1,3}$ и $S_{1,4}$, и четырьмя дугами, инцидентными $S_{1,1}$ и $S_{2,2}$.

Если теперь использовать неравенство треугольника, то получим

$$c_1(S_{1,4}, S_{1,3}) \leq c_1(S_{1,4}, S_{1,1}) + c_1(S_{1,1}, S_{1,2}) + c_1(S_{1,2}, S_{1,3}), \quad (21)$$

и, следовательно, назначение из рис. 22 (в котором дуги $(S_{1,4}, S_{1,1})$, $(S_{1,1}, S_{1,2})$, $(S_{1,2}, S_{1,3})$ заменены дугой $(S_{1,4}, S_{1,3})$) имеет значение, не превосходящее величины назначения из рис. 21 (б). Так как назначение на рис. 22 имеет по две дуги, инцидентных каждой вершине, и решение задачи о назначениях для стянутого графа имеет меньший вес назначения, то значение решения задачи о назначениях, скажем $V(AP_1)$, является нижней границей для величины назначения на рис. 21(б), т. е. для значения приращения веса, которое было бы необходимо для объединения вместе различных циклов.

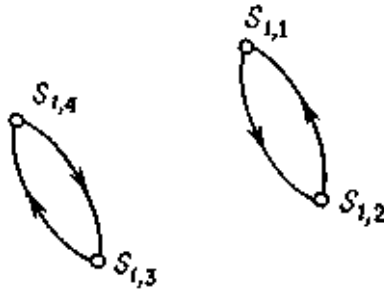


Рис. 22. Назначение, соответствующее рис. 21 (б), с двумя дугами в каждой вершине.

Очевидно, что поскольку граф назначения, полученный с помощью любого гамильтонова цикла, содержит после первого стягивания (т. е. в случае назначения, соответствующего рис. 21 (б)) эйлеров цикл, то его всегда можно преобразовать в граф с меньшим (или равным) весом, в котором каждой вершине инцидентны только две дуги. Это делается с помощью замены цепи, образованной дугами, ведущими из одной вершины в другую, единственной дугой между этими вершинами (как было продемонстрировано выше).

Если, с другой стороны, матрица относительных весов C_1 не удовлетворяет условию треугольника, то соотношение (21) может быть не выполнено. В этом случае матрица C_1 должна быть сначала подвергнута компрессии. Значение решения задачи о назначениях с компрессированной матрицей следует взять тогда в качестве нижней границы величины приращения веса, что необходимо при объединении

циклов. Это так, ибо компрессия матрицы может лишь уменьшить или оставить без изменения вес любого назначения для первоначальной матрицы.

Аналогично $V(AP_2)$ — вес решения задачи о назначениях после второго стягивания является нижней границей веса связываемых циклов, получаемых при этом стягивании, и так далее для третьего четвертого и остальных стягиваний. Поэтому величина

$$L = \sum_{i=0}^k V(AP_i). \tag{22}$$

где $V(AP_0)$ — значение начального решения задачи о назначениях с начальной матрицей C , а k — число стягиваний, необходимых для преобразования графа задачи в единственную вершину, равное весу решения задачи коммивояжера. Теорема доказана.

Вообще здесь следует отметить, что даже если начальная матрица весов удовлетворяет условию треугольника, то последующие матрицы относительных весов могут не удовлетворять этому условию и на каком-то этапе может потребоваться компрессия.

Пример вычисления границы

Рассмотрим задачу коммивояжера с 10 вершинами, матрица весов которой симметрична и приведена в табл. 3.

Таблица 3

Начальная матрица весов

2	32								
3	41	22							
4	22	50	63						
5	20	42	41	36					
6	57	51	30	78	45				
7	54	61	45	72	36	22			
8	32	20	10	54	32	32	41		
9	22	54	60	20	22	67	57	50	
10	45	51	36	64	28	20	10	32	50

Решение начальной задачи о назначениях дает величину $V(AP_0) = 184$.
 Результирующая матрица относительных весов дана в табл. 4, а
 решение дается графом на рис. 23.

Таблица 4

Матрица относительных весов

	1	2	3	4	5	6	7	8	9	10
1	∞	12	31	2	0	37	44	24	2	37
2	0	∞	0	18	10	19	39	0	22	31
3	19	0	∞	41	19	8	33	0	38	26
4	2	30	53	∞	16	58	62	46	0	56
5	0	22	31	16	∞	25	26	24	2	20
6	25	19	8	46	13	∞	0	12	35	0
7	32	39	33	50	14	0	∞	31	35	0
8	12	0	0	34	12	12	31	∞	30	24
9	2	34	50	0	2	47	47	42	∞	42
10	25	31	26	44	8	0	0	24	30	∞

↓

Решение задачи
о назначениях →

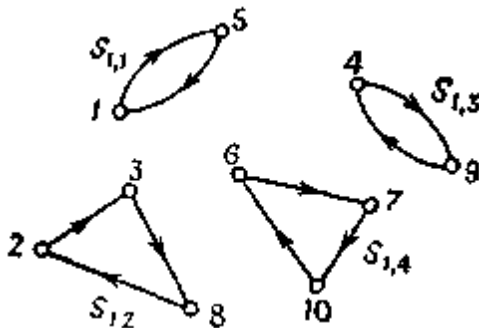


Рис. 23. Первое решение задачи о назначениях.

Стягивание графа из рис. 23 дает граф с 4 вершинами, матрица весов которого может быть вычислена по (19), и она дана в табл. 5 (а).

Таблица 10.5а

	1	2	3	4
1	∞	12	2	20
2	0	∞	2	8
3	2	14	∞	22
4	8	8	10	∞

Решение задачи
о назначениях



Таблица 10.6

	1	2	3	4
1	∞	10	0	10
2	0	∞	2	0
3	0	12	∞	12
4	0	0	2	∞

Матрица весов



Таблица 10.5б

	1	2	3	4
1	∞	12	2	20
2	0	∞	18	8
3	2	30	∞	42
4	8	8	30	∞

Таблица 10.7

	1	2
1	∞	0
2	10	∞



Эта матрица весов не удовлетворяет условию треугольника и поэтому подвергается компрессии. Результат дан в табл. 5 (б).

Решение задачи о назначениях с матрицей из табл. 5 (б) дает величину $V(AP_1) = 20$. Результирующая матрица относительных весов дана в табл/ 6, а решение задачи о назначениях дается графом на рис. 24.

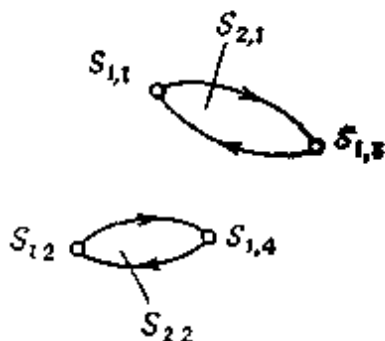


Рис. 24. Решение после первого стягивания.

Стягивание графа из рис. 24 дает граф с 2 вершинами, матрица весов которого (вычисленная по (20)) дана в табл. 7 (Эта тривиальная (2×2) -матрица не должна подвергаться компрессии, так как она удовлетворяет условию треугольника.)

Решение задачи о назначениях с матрицей из табл. 7 имеет значение $V(AP_2)=10$, а само решение дается графом на рис. 25, который переходит в единственную вершину после следующего сжатия.



Рис. 25. Решение после второго стягивания.

Таким образом, нижняя граница значения для задачи коммивояжера с матрицей из табл. 3 равна

$$L = V(AP_0) + V(AP_1) + V(AP_1) = 214$$

Сравнив ее со значением оптимального решения задачи коммивояжера, равного 216, получим ошибку в 0,93 %, в то время как использование в качестве нижней границы величины $V(AP_0)$ дает ошибку в 14,8%. Хотя настоящая нижняя граница и не является в общем случае такой близкой, как в приведенном примере, но результаты показывают, что эта граница в среднем значительно лучше, чем $V(AP_0)$

Время вычисления границы в задаче коммивояжера в среднем только на 9% больше времени, требуемого для решения задачи о назначениях

с той же самой матрицей весов. Время вычисления при решении задачи о назначениях с использованием венгерского метода оценивается как kn^3 , где k — некоторая постоянная, а n — порядок матрицы. Самый плохой случай при вычислении границы (что касается времени вычисления) возникает тогда, когда все циклы при каждом стягивании содержат только по две вершины. В этом случае общее время вычисления при решении задачи о назначениях таково:

$$kn^3 + k \left(\frac{n}{2}\right)^3 + k \left(\frac{n}{4}\right)^3 + \dots \approx \frac{8}{7} kn^3 \approx 1,143 kn^3. \quad (23)$$

Из (23) видно, что в худшем случае требуемое время для вычисления предложенной границы в задаче коммивояжера только на 14,3% больше, чем время, требуемое для решения задачи о назначениях того же самого размера. (Здесь следует заметить, что время, необходимое для выполнения той части процесса, которая связана с «стягиванием» и «компрессией», оценивается как n^2 .)

7.7.5. Пример из раздела

Рассмотрим еще раз пример из раздела 7.7.2, используя улучшенную границу, описанную выше (вместо использования в качестве такой границы просто значения решения задачи о назначениях). Граница, соответствующая начальной задаче с матрицей весов C_A (разд. 7.7.2), должна тогда быть $V(AP_0) + V(AP_1) = 232 + 13 = 245$. (Необходимо второе стягивание, и решение новой задачи о назначениях равно 13.) Подзадачи B , C и D получены, как и раньше, но новой границей для подзадачи C будет $V(AP_0) + V(AP_1) = 248 + 6 = 254$, вместо предыдущего значения 248. Новой границей для подзадачи D будет $V(AP_0) + V(AP_1) = 250 + 0 = 250$ — то же, что и раньше. (Граница для B останется, конечно, неизменной, так как решение задачи B является на самом деле гамильтоновым циклом. Теперь ветвление нужно делать в узле D (так как $250 < 254$), а не в C , как в предыдущем случае. Это ветвление приводит, как и прежде, к подзадачам G , H и I , и наилучшее решение (подзадача H) имеет значение 251. Дальнейшее ветвление в C (где решение не является гамильтоновым циклом) уже не будет необходимым, так как его нижняя граница, а именно 254, больше чем 251. Прежняя граница для C , равная 248 (меньше 251), была недостаточной для того, чтобы прекратить ветвление в этом узле. Таким образом, улучшенная граница позволяет сэкономить два узла в дереве решений поиска, как это показано на рис. 26.

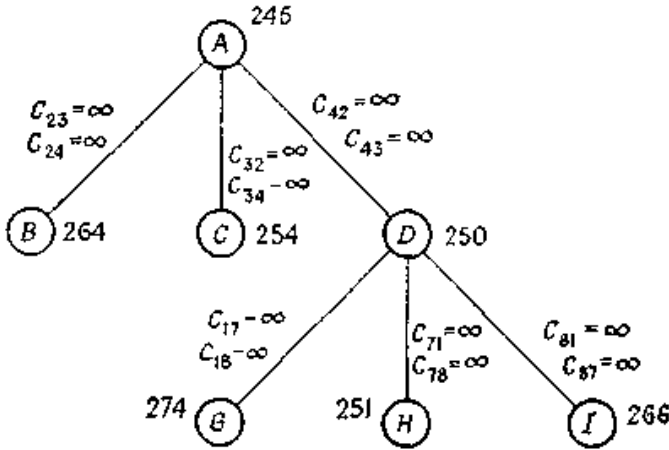


Рис. 26. Поиск с деревом решений из примера 7.7.2 с улучшенной границей для задачи о назначениях

Это дает экономию только на 2/9 для этой маленькой задачи, но для задач большого размера получается большой процент экономии узлов дерева, т. е. числа задач о $(n \text{ на } n)$ назначениях, которые следует решить

7.7.6. Приложение

Ниже излагается алгоритм вычисления верхней границы B минимального числа перенастроек аппаратуры, требующихся в задаче планирования из разд. 7.4, который можно использовать в качестве начального алгоритма этого раздела.

Шаг 0. Индекс $\leftarrow 0, B \leftarrow 0$.

Шаг 1. Взять метки $l(x_i) = 0, \forall x_i \in X$. Положить $p = 0$.

Шаг 2. $G = (X, A)$, выбрать любое $x_0 \in X$, положить $S = \{x_0\}$.

Шаг 3. Если индекс $= 0$, образовать $\bar{S} = S \cup \Gamma(S)$; в противном

случае $\bar{S} = S \cup \Gamma^{-1}(S)$. Здесь $\Gamma(x_i) = \{x_j \mid (x_i, x_j) \in A\}$ и $\Gamma(S) = \bigcup_{x_i \in S} \Gamma(x_i)$, а также

$$\Gamma^{-1}(x_i) = \{x_j | (x_j, x_i) \in A\} \text{ и } \Gamma^{-1}(S) = \bigcup_{x_i \in S} \Gamma^{-1}(x_i).$$

Шаг 4. Если $\bar{S} = S$, то перейти к шагу 5; в противном случае $p \leftarrow p + 1$, $l(x_i) \leftarrow p \forall x_i \in \bar{S} - S$, $\bar{S} \leftarrow S$ и вернуться к шагу 3.

Шаг 5. Найти $x \in \{x_i | l(x_i) = p\}$.

Шаг 6. Если индекс =0, найти $x' \in \{x_i | l(x_i) = p - 1 \text{ и } (x', x) \in A\}$; в противном случае найти $x' \in \{x_i | l(x_i) = p - 1 \text{ и } (x', x) \in A\}$.

Шаг 7. $X \leftarrow X - \{x\}$, $A \leftarrow A - \{(x, x_i) | x_i \in X\} - \{(x_i, x) | x_i \in X\}$. Если $x = \{x_0\}$, то $B \leftarrow B + 1$ и перейти к шагу 12; в противном случае перейти к шагу 8.

Шаг 8. $x \leftarrow x'$, $p \leftarrow p - 1$. Если $x' = x_0$, перейти к шагу 9; в противном случае перейти к шагу 5.

Шаг 9. Если индекс =0, индекса $\leftarrow -1$ и перейти к шагу 1; в противном случае перейти к шагу 10.

Шаг 10. Если $X = \{x_0\}$, то $B \leftarrow B + 1$ и перейти к шагу 12; в противном случае перейти к шагу 11.

Шаг 11. Индекс $\leftarrow 0$, $B \leftarrow B + 1$.

$$X \leftarrow X - \{x_0\}, A \leftarrow A - (\{(x_0, x_i) | x_i \in X\} - \{(x_i, x_0) | x_i \in X\}).$$

Шаг 12. Стоп. B является требуемой верхней границей.

Алгоритм требует некоторых пояснений. Если индекс =0, то прослеживаются *прямые* цепи, проходящие по вершинам графа G , начиная с вершины x_0 . Эти цепи прослеживаются с присвоением метки p тем вершинам из G , которые могут быть достигнуты из x_0 через p дуг (шаги 1, 2, 3 и 4). Если ни одна из этих цепей не может быть продолжена, то алгоритм переходит к шагам 5, 6 и 7, в которых самая длинная из этих цепей прослеживается назад к вершине x_0 . при этом убираются из графа вершины (и ассоциированные с ними дуги), лежащие на этой самой длинной цепи. Шаг 8 прекращает процесс удаления вершин. Шаг 9 возвращает к началу алгоритма со значением индекса =1, чтобы начать формирование *обратных* цепей, т. е. цепей, оканчивающихся в x_0 . Снова находится и удаляется самая длинная из этих цепей.

Самая длинная из прямых и обратных цепей (в или из x_0) может рассматриваться как одна длинная цепь, содержащая x_0 . Число B (необходимых последовательностей цепей) увеличивается тогда на единицу на шаге 11, и после выбора другой вершины x_0 из оставшегося графа процесс продолжается (формируются новые длиннейшие прямые и обратные цепи и т. д.), пока не будет исчерпан весь граф.

Конечное значение B , которое равно числу последовательностей цепей, необходимых для «стирания» всего графа (т. е. для покрытия всех вершин), является тогда, очевидно, верхней границей для требуемого минимального числа таких покрывающих последовательностей (т. е. минимального числа перенастроек аппаратуры).

8. Ациклические графы

8.1. Базисные графы.

Пусть G — ациклический граф с однократными ребрами. Тогда из $a > b$, $b > a$ следует $a = b$, и соответствующее графу G отношение включения $a > b$ является частичным упорядочением, граф G_i которого есть транзитивное замыкание G . Если $a > c > b$, то будем говорить, что c лежит между a и b , или что c есть промежуточная вершина для a и b .

Пусть $E = (a, b)$ — некоторое ребро. Если имеются промежуточные вершины между a и b , то ребро E оказывается излишним; если вершин между a и b нет, то ребро E существенное. В этом случае b непосредственно следует за a и a непосредственно предшествует b . Ориентированная цепь $P(a, b)$ в G максимальна, если все ее ребра существенные, т. е. если любая вершина из P непосредственно следует за предыдущей.

(Ориентированные) цепи между a и b конечны, если не существует ни бесконечно убывающей последовательности вершин

$$a > a_1 > a_2 > \dots > b,$$

ни бесконечно возрастающей последовательности

$$a > \dots > b_2 > b_1 > b.$$

В этом случае цепи между любой парой вершин $a' > b'$ также конечны при

$$a \geq a' > b' \geq b.$$

Пусть цепи между a и b конечны, $P(a, b)$ — любая цепь от a до b и $E_i = (a_i, a_{i+1})$ — несущественное ребро в P . Тогда можно увеличить P , заменяя E_i некоторой более длинной цепью $P_i(a_i, a_{i+1})$. Можно проверить, что, повторяя этот процесс конечное число раз любым способом, в конце концов получим максимальную цепь $Q(a, b)$. Обратно, легко видеть, что если цепи между a и b обладают этим свойством увеличения, то все цепи между ними конечны.

Теорема 1. *Ациклический граф G имеет не более одного базисного графа B , и если такой граф B существует, то он состоит из существенных ребер.*

Доказательство. Существенные ребра должны принадлежать каждому порождающему графу. С другой стороны, пусть $E_0 = (a_0, a_1)$ — ребро в базисном графе B . Если бы E_0 было излишним в G , то между a_0 и a_1 существовала бы некоторая вершина c . Так как B — базис, существуют цепи $P(a_0, c)$ и $P(c, a_1)$ в B . Но E_0 не может быть излишним в B , так что объединенная цепь

$$P(a_0, c) \cup P(c, a_1)$$

должна содержать E_0 . Но это приводит к ориентированному простому циклу, проходящему через a_0 или a_1 , в противоречии с тем, что G ациклический.

Теорема 2. *Для того чтобы ациклический граф имел базис, необходимо и достаточно, чтобы концы любого ребра $E_0 = (a_0, a_1)$ были связаны максимальной цепью.*

Доказательство. Если базисный граф существует, то условие выполняется. С другой стороны, если оно выполняется, то все существенные ребра определяют базис. Отсюда непосредственно следует

Теорема 3. *Ациклический граф имеет базис, если для каждого ребра $E_0 = (a_0, a_1)$ цепи, связывающие a_0 и a_1 конечны.*

Можно поставить следующую задачу. Пусть G — неориентированный граф. В каких случаях можно приписать его ребрам такие ориентации, чтобы он стал базисным графом частичного упорядочения? Легко видеть, что для треугольника, т. е. для простого цикла, состоящего из трех ребер, такая конструкция невозможна. Поэтому нужно потребовать, чтобы граф G имел простые циклы только длины $l \geq 4$. Однако этого условия недостаточно. Общая задача еще не решена.

8.2. Деформации цепей.

Выведем некоторые результаты для ациклических графов, которые содержат как частный случай хорошо известные теоремы типа теоремы Жордана — Гёльдера для групп и других алгебраических систем.

Предположим, что в ациклическом графе G две вершины a и b связаны максимальными ориентированными цепями

$$\begin{aligned} A(a, b) &= (a_0, a_1) \dots (a_{r-1}, a_r), & a_0 &= a, & a_r &= b, \\ B(a, b) &= (b_0, b_1) \dots (b_{l-1}, b_l), & b_0 &= a, & b_l &= b. \end{aligned} \quad (1)$$

Будем говорить, что цепи A и B в (1) *взаимно просты*, если у них общие только концы a и b и не существует такой вершины c , чтобы были максимальными цепи

$$P(a, c) \cup Q(c, a_i), \quad P(a, c) \cup R(c, b_j) \quad (2)$$

или цепи

$$Q_1(a_i, c) \cup P_1(c, b), \quad R_1(b_j, c) \cup P_1(c, b) \quad (3)$$

для какой-либо пары промежуточных вершин a_i и b_j (рис. 1).

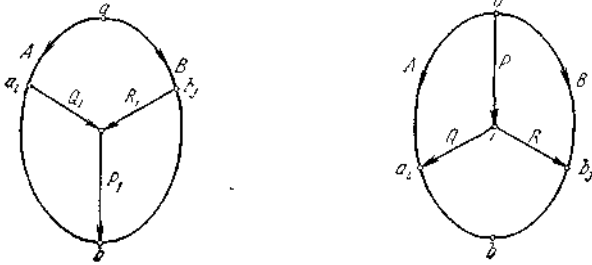


Рис. 1.

Если a_r и a_s — две вершины в максимальной цепи $A(a, b)$ в (1), то они должны быть связаны участком $A(a_r, a_s)$. Если существует другая максимальная цепь $B_1(a_r, a_s)$, то

$$A_1(a, b) = A(a, a_r) \cup B_1(a_r, a_s) \cup A(a_s, b)$$

также является максимальной цепью от a до b . Будем говорить, что A_1 получено из A *деформацией*. Деформация называется *простой*, если цепи

$$A(a_r, a_s), \quad B_1(a_r, a_s)$$

взаимно просты. Пара цепей (1) называется *деформационно эквивалентной*, если одна получается из другой конечным числом простых деформаций.

Теорема 4. Пусть в ациклическом графе G цепи между вершинами $a > b$ конечны. Тогда любая пара максимальных цепей (1), связывающих эти вершины, деформационно эквивалентна.

Доказательство. Допустим, что цепи $A(a, b)$ и $B(a, b)$ не являются деформационно эквивалентными. Если они имеют общие вершины, кроме a и b , то каждая из них должна разлагаться на участки A_i и B_i так, что или $A_i = B_i$, или они имеют только общие концы. Если все A_i и B_i взаимно просты, то цепи A и B будут деформационно эквивалентными. Поэтому можно предполагать, что A и B имеют только общие концы a и b и не деформационно эквивалентны. Тогда они не будут и взаимно

простыми, и найдется некоторая вершина c с цепями (2) или (3). В первом случае максимальные цепи

$$P(a, c) \cup Q(c, a_i) \cup A(a_i, b),$$

$$P(a, c) \cup R(c, b_j) \cup B(b_j, b)$$

также соединяют a и b . Так как A и B не деформационно эквивалентны, цепи хотя бы в одной из трех пар

$$A(a, a_i); \quad P(a, c) \cup Q(c, a_i),$$

$$Q(c, a_i) \cup A(a_i, b); \quad R(c, b_j) \cup B(b_j, b),$$

$$P(a, c) \cup R(c, b_j); \quad B(a, b_j)$$

не могут быть деформационно эквивалентными. Аналогичный результат получается, когда существуют цепи (3). Эти утверждения можно объединить следующим образом. Если цепи $A(a, b)$ и $B(a, b)$ не деформационно эквивалентны, то существуют такие вершины a' и b' с

$$a \geq a' > b' \geq b$$

(где оба равенства одновременно не допускаются), что a' и b' также связаны парой не эквивалентных цепей. Это рассуждение может быть повторено, и мы получим последовательности

$$a \geq a' \geq a'' \geq \dots \geq b'' \geq b' \geq b$$

пар вершин с тем же свойством. Так как по крайней мере одна из этих последовательностей бесконечна, мы приходим к противоречию.

Из теоремы 4 немедленно получается следствие:

Теорема 5. Пусть G — ациклический граф с конечными цепями между любыми вершинами $a > b$. Для того чтобы максимальные цепи $A(a, b)$ и $B(a, b)$ всегда имели одну и ту же длину, необходимо и достаточно, чтобы любая пара взаимно простых цепей обладала этим свойством.

Алгебраические приложения этой теоремы к теоремам Жордана — Гёльдера опираются на тот факт, что для соответствующих систем выполняется следующее специальное условие:

Условие четырехугольника. Если

$$a > a_1 > b, \quad a > b_1 > b,$$

где $a_1 \neq b_1$ непосредственно следуют за a , то существует вершина c между a и b , которая непосредственно следует как за a_1 , так и за b_1 (рис. 2).

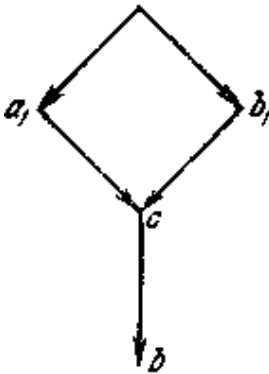


Рис. 2.

Если условие четырехугольника выполняется, то взаимно простые цепи имеют вид

$$(a, a_1) \cup (a_1, c); \quad (a, b_1) \cup (b_1, c),$$

и, следовательно, каждая из них имеет длину 2.

Чтобы получить обычную теорему Жордана — Гёльдера для *главных рядов* в группах, достаточно рассмотреть частично упорядоченное по включению множество всех нормальных делителей данной группы G_0 . Если в этой системе нормальный делитель A содержит два максимальных нормальных делителя A_1 и B_1 , то пересечение $C = A_1 \cap B_1$ также будет нормальным делителем в G_0 , и притом максимальным в A_1 и в B_1 согласно теореме об изоморфизме. Поэтому условие четырехугольника в данном случае выполнено.

Чтобы вывести теорему Жордана — Гёльдера для *композиционных рядов*, т. е. рядов подгрупп, в которых каждый член является максимальным нормальным делителем предыдущего, применяются те же рассуждения для частично упорядоченного множества всех композиционных подгрупп, т. е. подгрупп, содержащихся в композиционных рядах.

8.3. Графы воспроизведения.

Обратимся к одному применению теории графов в области биологии. Пусть V — популяция индивидуальных организмов некоторого вида, способных к размножению. Связи членов популяции V со своим потомством могут быть выражены бинарным отношением *непосредственного потомства* π , так что

$$a \pi b, \quad a, b \in V,$$

если b есть непосредственный потомок (дитя, отпрыск) индивидуума a . Отношению π соответствует ориентированный граф $G = G(\pi)$, который можно назвать *графом воспроизведения* для V . Он имеет однократные ребра и не имеет петель; в G существует ребро (a, b) тогда и только тогда, когда b является непосредственным потомком a . Если существует ориентированная цепь $P(c, d)$ между двумя вершинами, то, как и раньше, пишем $c > d$ и называем d *потомком* c , а c — *предком* d . Соответствующий *граф потомства* есть транзитивное замыкание графа G .

Множество вершин V должно распадаться по полу на два непересекающихся класса: отцов m и матерей f , так что

$$V = M \cup F, \quad m \in M, \quad f \in F. \quad (4)$$

При объединении двух индивидуумов из разных классов получается один или более отпрысков. Этот процесс воспроизведения имеет два свойства:

1. Каждый индивидуум имеет двух родителей.
2. Никакой индивидуум не является своим потомком.

В терминах графов это означает:

1. Число входящих ребер в каждой вершине $a \in V$ равно

$$\rho^*(a) = 2. \quad (5)$$
2. Граф является ациклическим.

Исследуем здесь графы с такими свойствами и поставим следующий вопрос. Пусть в графе выполняются указанные два условия. В каких случаях можно вершинам G приписать *пол*, т. е. когда существует такое непересекающееся разложение (4) множества вершин, что в каждой вершине имеется по одному ребру, приходящему в нее из каждого множества?

Рассмотрим сначала эту задачу в несколько более общей форме, опуская условие ацикличности. Кроме того, предположим, что вместо (5) выполняется только условие

$$\rho^*(a) \leq 2. \quad (6)$$

В этой связи для произвольного ориентированного графа G удобно использовать терминологию, что a является *родителем* b , если существует ребро (a, b) , направленное от a к b . Будем также говорить, что граф G имеет *дихотомию по полу*, если существует такое разложение (4) на два непересекающихся класса, что каждый индивидуум, имеющий хотя бы двух родителей, имеет их ровно два, по одному из каждого класса M и F .

Предположим теперь, что G удовлетворяет неравенству (6). Припишем характеристику пола t некоторой произвольной вершине a_0 . Если a_0 имеет непосредственного потомка a_1 , то a_1 может также быть

непосредственным потомком некоторого другого родителя a_2 . В этом случае a_2 нужно приписать пол f . Для a_1 поступаем таким же образом. Если a_2 имеет некоторого непосредственного потомка a_3 и a_3 имеет некоторого другого родителя a_4 , то a_4 должно иметь пол m . Продолжая этот процесс, мы припишем пол каждой из вершин a_{2n} , достижимых из a_0 (α, α^*)-цепями, т. е., каждой вершине множества

$$V(a_0; \alpha, \alpha^*). \quad (7)$$

Однако, чтобы так полученные характеристики m и f определялись однозначно, все цепи к a_{2n} должны давать одну и ту же характеристику. Легко видеть, что для этого необходимо и достаточно, чтобы длины всех (α, α^*) -циклов делились на 4. Если это условие удовлетворяется, то вершинам в каждом множестве (7) можно приписать характеристики с нужными свойствами. Отсюда получается

Теорема 6. *Для того чтобы ориентированный граф, удовлетворяющий (6), имел дихотомию по полу, необходимо и достаточно, чтобы длины всех чередующихся (α, α^*) -циклов делились на 4.*

При подходящей переформулировке условие (6) может быть в теореме опущено. Будем говорить, что чередующийся (α, α^*) -маршрут *правильный*, если ни за каким ребром $A = (a, b)$ непосредственно не следует обратное ребро A^* . Тогда мы имеем теорему:

Теорема 7. *Для того чтобы ориентированный граф имел дихотомию по полу, необходимо и достаточно, чтобы длины всех чередующихся правильных (α, α^*) -маршрутов делились на 4.*

Доказательство. Условие, очевидно, необходимо. С другой стороны, из него следует (6). Если бы в некоторой вершине было 3 и более входящих ребер (рис. 3), то имелся бы циклический правильный (α, α^*) -маршрут длины 6.

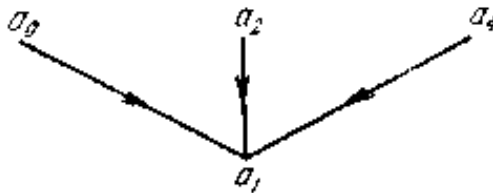


Рис. 3

Теорему 6 можно сформулировать в терминах графа чередующейся композиции, (α, α^*) -маршруту в G соответствует единственный маршрут в $G \cdot G^*$. Петли в $G \cdot G^*$ определяются 2-маршрутами, состоящими из ребра и его обращения. Они исключаются, когда мы

ограничиваемся правильными (α, α^*) -маршрутами. Если циклические правильные (α, α^*) -маршруты в G имеют длины, делящиеся на 4, то циклы в $G \bullet G^*$ имеют четные длины; следовательно, $G \bullet G^*$ есть двудольный граф, если пренебречь петлями, и наоборот. Кроме того, множества (7) являются связными компонентами графа $G \bullet G^*$. В каждой из них можно приписать характеристику t или f дайной вершине a_0 , и характеристики остальных вершин определяются однозначно. Таким образом, теорему 7 можно сформулировать иначе.

Теорема 8. *Для того чтобы ориентированный граф G имел дихотомию по полу, необходимо и достаточно, чтобы граф чередующейся композиции $G \bullet G^*$ без петель был двудольным. Количество способов, которыми можно приписать характеристики, равно 2^k , где k есть число связных компонент графа $G \bullet G^*$.*

Вернемся к графам воспроизведения. Можно построить такой граф последовательно, полагая, что от пары вершин a и b происходит новая вершина c , соединенная с ними ребрами (a, c) и (b, c) . Если в таком ациклическом графе каждый индивидуум имеет двух родителей, то граф должен быть бесконечным. В генетических экспериментах или в их описаниях обычно исходят из некоторой данной ограниченной популяции, или порождающего множества B , от которого происходят последующие поколения. Предки этих данных индивидуумов неизвестны, и для каждого b из B полагают $\rho^*(b) = 0$. Можно также начинать с некоторого порождающего множества, в котором семейные отношения известны только не полностью. Поэтому приходится рассматривать условие (6) как характеризацию общих графов воспроизведения. При этом определении из теоремы 7 следует

Теорема 9. *Ориентированный граф G является графом воспроизведения тогда и только тогда, когда он ациклический и его граф чередующейся композиции $G \bullet G^*$ двудольный.*

В графе воспроизведения можно ввести обычные понятия семейных отношений: отец, мать, брат, сестра, сводный брат, сводная сестра, дядя, тетя, двоюродные брат и сестра, племянницы и племянники и т. д. Большинство обычных запретов в человеческом обществе могут быть представлены как простые ограничения на граф воспроизведения. Например, смешение с потомками не разрешается. В терминах графов это означает, что если существует ориентированная цепь $P(a_0, a_n)$, то не может существовать ребра (a_0, a_n) , иными словами, граф воспроизведения должен быть базисным. Читатель может описать на языке графов и другие запреты.

Приложение 1

МЕТОДЫ ПОИСКА, ИСПОЛЬЗУЮЩИЕ ДЕРЕВО РЕШЕНИЙ

1. Принцип поиска, использующий дерево решений

Основной принцип, на котором базируются методы поиска с деревом решений, состоит в разбиении начальной задачи P_0 на некоторое число подзадач P_1, P_2, \dots, P_k (в целом представляющих всю задачу P_0) с последующей попыткой *разрешить* каждую из этих подзадач. Выражение *разрешить* мы понимаем так:

либо (i) найти оптимальное решение,

либо (ii) показать, что значение оптимального решения хуже, чем для полученного до этого наилучшего решения,

либо (iii) показать, что подзадача не является допустимой.

Это разбиение описывается *деревом* на рис. П.1, причем вершины изображают подзадачи.

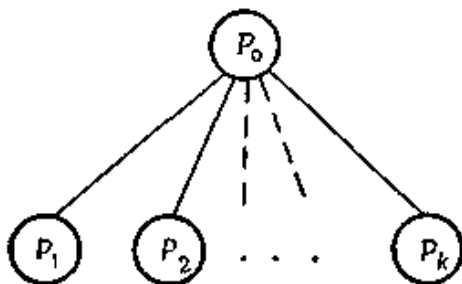


Рис. П.1. Разбиение задачи P_0 на подзадачи.

Смысл разбиения задачи P_0 на некоторое число подзадач состоит в том, что или эти подзадачи проще разрешить, или они имеют меньший размер, или обладают структурой, не присущей первоначальной задаче P_0 . Но, вообще говоря, все еще может оказаться, что подзадачу P_i нельзя разрешить, и эта подзадача сама разбивается на новые подзадачи $P_{i_1}, P_{i_2}, \dots, P_{i_r}$, как это показано на рис. П.2.

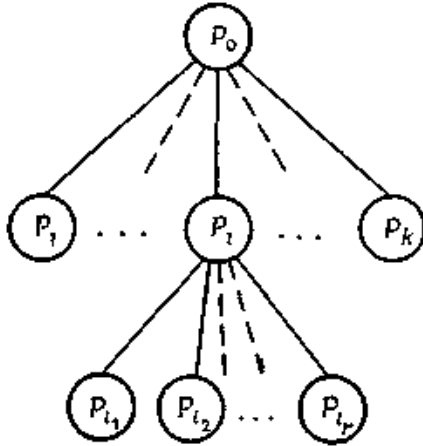


Рис. П.2. Дерево после ветвления в вершине P_i .

Это разбиение (называемое также *ветвлением*), повторяется для каждой подзадачи, которая не может быть разрешена.

На любом этапе полное множество подзадач, требующих разрешения, представляется множеством концевых вершин (т. е. вершин степени 1) всех цепей, исходящих из корня дерева решений. (Корень этого дерева изображает начальную задачу P_0 .) Эти вершины называются *висячими* вершинами, и на рис. П.2 это

$$P_1, \dots, P_{i-1}, P_{i_1}, \dots, P_{i_r}, P_{i+1}, \dots, P_k$$

Если поиск исчерпан, то очевидно, что множество подзадач, на которые разбита задача, должно представлять всю задачу. Таким образом, если задача P_i разбита на r подзадач $P_{i_1}, P_{i_2}, \dots, P_{i_r}$, то

$$\{P_{i_1}\} \cup \{P_{i_2}\} \cup \dots \cup \{P_{i_r}\} = \{P_i\}, \quad (\text{П.1})$$

где $\{P\}$ обозначает множество всех допустимых решений задачи P .

Так как соотношение (П.1) должно быть применено к каждому разбиению, то

$$\{P_0\} = \cup \{P(j) \mid P(j) \text{ — висячая вершина дерева}\}. \quad (\text{П.2})$$

В случаях когда требуется перебрать все решения задачи P_0 (а не только найти оптимальное решение), желательно уметь перебирать решения с помощью вышеприведенного разбиения задачи на подзадачи и перебирать решения каждой из этих подзадач. В этом

случае нужно избежать дублирования построенных решений, т. е. нужно разбивать задачу P_i на подзадачи $P_{i_1}, P_{i_2}, \dots, P_{i_q}$ так, чтобы

$$\{P_{i_s}\} \cap \{P_{i_q}\} = \emptyset \quad (\text{П.3})$$

для любых двух подзадач P_{i_s} и P_{i_q} , для которых $s \neq q$.

Соотношение (П.3) определяет собственное разбиение задачи P_i . Хотя условие (П.3) не является необходимым для полноценного поиска с деревом решений, оно тем не менее имеет большие выгоды с вычислительной точки зрения, так как

(а) для задачи оптимизации P_o оптимальное решение является решением одной и только одной подзадачи, представляемой висячей вершиной;

(б) для задачи полного перебора объединение множеств решений подзадач, представляемых висячими вершинами, дает множество всех решений задачи P_o без дублирования.

2. Некоторые примеры ветвления

Рассмотрим задачу P_i с n переменными, в которой некоторая переменная ξ может принимать только четыре возможных значения, скажем, a, b, c и d .

(а) Возможно разбиение P_i на четыре подзадачи $P_{i_1}, P_{i_2}, P_{i_3}$ и P_{i_4} , причем для подзадачи P_{i_1} мы полагаем $\xi = a$, для P_{i_2} полагаем $\xi = b$, для P_{i_3} — $\xi = c$ и для P_{i_4} — $\xi = d$. Каждая из подзадач P_{i_1}, \dots, P_{i_4} содержит $n - 1$ переменных и, следовательно, допускает более простое решение, чем задача P_i (рис. П.3 (а)).

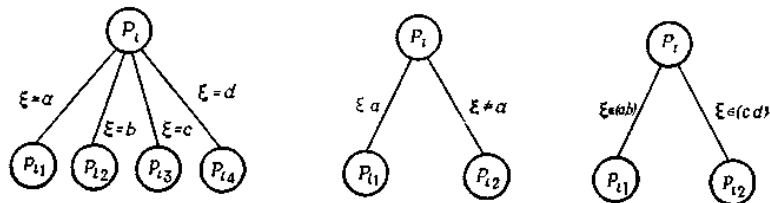


Рис. П.3. Три возможных способа ветвления в вершине P_i .

(б) Возможно другое разбиение P_i на две подзадачи P_{i_1} и P_{i_2} ,

где для P_{i_1} мы полагаем $\xi = a$, а для P_{i_2} полагаем $\xi \neq a$, т. е. ξ равно b , c или d (рис. П.3 (б)).

(в) Еще одно возможное разбиение P_i на две подзадачи P_{i_1} и P_{i_2} , где для P_{i_1} $\xi = a$ или b , а для P_{i_2} $\xi = c$ или d (рис. П.3 (в)).

Все три ветвления являются допустимыми и удовлетворяют условию (П.3). Какому из них отдать предпочтение — зависит от природы решаемой задачи, причем возможности типа (а) или (б) используются чаще остальных.

3. Типы поиска, использующего дерево решений

Из вышесказанного видно, что любая подзадача, представляемая висячей вершиной и не поддающаяся разрешению, может быть в любой момент разбита на меньшие подзадачи. Но существует только два основных типа поиска в зависимости от того, как выбирается следующая висячая вершина для продолжения процесса ветвления.

3.1. Поиск по глубине

При этом типе поиска ветвление осуществляется в последней полученной подзадаче до тех пор, пока не будет порождена подзадача, которую можно разрешить. В этом месте делается шаг *возвращения*, т. е. берется предпоследняя порожденная подзадача и ветвление продолжается в соответствующей вершине. При этом типе поиска задачи, получаемые на каждом этапе, хранятся в *стеке* вместе с самой верхней задачей, выбранной для исследования (разрешения_ или разбиения). Вновь получаемые задачи помещаются в верх этого стека, а когда подзадача разрешена, она удаляется от стека.



Рис. П.4а. Дерево поиска по глубине.

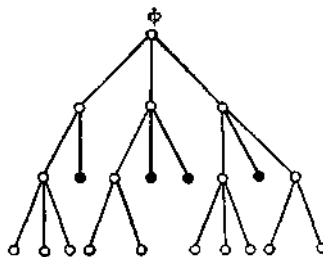


Рис. П.4б. Дерево поиска по ширине.

• Задача решена.

Вид дерева решений при этом типе поиска, когда разрешается первая подзадача, показан на рис. П.4а, где порядок приоритета исследования получаемых подзадач показан нумерацией.

3.2. Поиск по ширине

При поиске по ширине ветвление происходит от уровня к уровню, так что если на уровне 1 начальная задача P_0 разбивается на подзадачи P_1, P_2, \dots, P_k , то каждая из этих подзадач исследуется раньше, чем задачи уровня 2. Задачи уровня 1, которые не могут быть разрешены, разбиваются на подзадачи уровня 2, и опять все они исследуются до исследования какой-либо подзадачи, могущей возникнуть на уровне 3, и т. д. Вид дерева решений при этом типе поиска показан на рис. П.4б.

4. Применение границ

Если задача P_0 подлежит решению как задача оптимизации, то безотносительно к типу поиска этот поиск завершается только тогда, когда будут разрешены все подзадачи, представляемые всеякими вершинами. Для ускорения процесса разрешения для каждой из всеяких вершин вычисляются нижние или верхние границы (для случаев минимизации или максимизации соответственно). Эти

границы дают наименьшее (или наибольшее) возможное значение оптимального решения той подзадачи, которая соответствует рассматриваемой висячей вершине. Таким образом (для задачи минимизации), если окажется, что нижняя граница для вершины, соответствующей задаче P_i , больше, чем величина наилучшего ответа, полученного ранее при поиске, то в P_i нет необходимости производить дальнейшее ветвление, так как в $\{P_i\}$ нет решения, лучшего, чем текущий наилучший ответ. В соответствии с толкованием (II) термина «разрешение задачи» из разд. 1 подзадача P_i окажется автоматически разрешенной.

5. Функции ветвления

Как при поиске по глубине, так и при поиске по ширине выбор очередной вершины для ветвления не был полностью определен.

При поиске по глубине, когда после ветвления задача P_i разбивается на подзадачи $P_{i_1}, P_{i_2}, \dots, P_{i_r}$, очередное ветвление, как уже говорилось, производится в одной из этих только что порожденных подзадач. Но мы не указали, в какой именно, и любая из них может рассматриваться как «последняя порожденная». При поиске по ширине, как уже было сказано, все подзадачи данного уровня должны исследоваться до исследования задач следующего уровня, но не был указан порядок их исследования.

Функция ветвления — это функция, которая позволяет «вычислить», какая из *допустимых* вершин должна использоваться при следующем ветвлении. Для вершины, соответствующей подзадаче P_j , эта функция является некоторой мерой вероятности того, что оптимальное решение всей задачи P_o является решением для P_j . Совершенно очевидно, что вершина, соответствующая подзадаче с большими шансами на оптимальное решение, должна пользоваться правом преимущественного выбора при очередном ветвлении. Можно указать несколько эвристических мер этой вероятности, причем одна из полезных мер связана просто с вычислением для вершин нижних или верхних границ. Для такой меры вершина с более низкой нижней границей (для случая минимизации) считается имеющей большую вероятность.

После введения понятия функции ветвления сразу же возникает мысль о другом типе поиска с деревом решений (в дополнение к описанным ранее поискам по глубине и ширине). Можно использовать функцию ветвления и так, чтобы она *полностью* определяла выбор следующей для ветвления вершины. Например, если значениями функции

ветвления являются границы вершин (нижние и верхние), как упоминалось выше, то всегда можно производить ветвление в той висячей вершине, нижняя граница которой наименьшая. Этот тип поиска является, вообще говоря, гибридом поисков по глубине и ширине, хотя в литературе он часто называется поиском по ширине.

Литература

1. Оре О. Теория графов. – М.: Наука, 1980
2. Харари Ф. Теория графов. – М.: Мир, 1973
3. Харари Ф., Палмер Э. Перечисление графов. – М.: Мир, 1977
4. Кристофидес Н. Теория графов. – М.: Наука, 1978
5. Горбатов В.А. Основы дискретной математики. – М.: Высшая школа, 1986
6. Сигорский В.П. Математический аппарат инженера. – К.: Техніка, 1975
7. Берж К. Теория графов и ее применение. – М.: Наука, 1962
8. Баскер Р., Саати Т. Конечные графы и сети. – М.: Наука, 74
9. Оре О. Графы и их применение. – М.: Мир, 1965
10. Коршунов Ю.М. Математические основы кибернетики. – М.: Энергия, 1980
11. Кузнецов О.П., Адельсон-Вельский Г.М. Дискретная математика для инженера. – М.: Энергия, 1980
12. Кук Д., Бэйз Г. Компьютерная математика. – М.: Наука, 1990

.□