Лекция 8: Алгоритмы для задач о паросочетаниях

Б.М.Верников, А.М.Шур

Уральский федеральный университет,
Институт математики и компьютерных наук,
кафедра алгебры и дискретной математики

В этой лекции мы опишем алгоритм решения задачи о назначениях, а также более сложного ее варианта — задачи об оптимальных назначениях. Напомним формулировку задачи.

Задача о назначениях

В данном двудольном графе найти максимальное паросочетание.

Нам потребуются два понятия для работы с паросочетаниями. Пусть дан двудольный граф G и паросочетание P в нем.

- Простая цепь в G называется *чередующейся относительно* P, если из любых двух соседних ребер этой цепи ровно одно лежит в P.
- Чередующаяся относительно P (u, v)-цепь называется *увеличивающей относительно* P, если вершины u и v свободны относительно P.

Если из контекста понятно, о каком паросочетании идет речь, мы опускаем слова «относительно P», ребра из паросочетания называем *темными*, остальные ребра графа — *светлыми*, а изменение статуса ребра (добавление в паросочетание или удаление из паросочетания) — *перекраской ребра*.

Лемма об увеличивающей цепи

Если в двудольном графе G есть увеличивающая относительно паросочетания P цепь, то при перекраске всех ребер этой цепи получится паросочетание, содержащее на одно ребро больше, чем P.

Доказательство. Так как крайние вершины увеличивающей цепи являются свободными, в такой цепи светлых ребер на единицу больше, чем темных, а значит, после перекраски всех ребер цепи число темных ребер в графе вырастет на единицу.

Проверим, что ни одной вершине не инцидентны два темных ребра. Так как длина увеличивающей цепи нечетна, ее крайние вершины лежат в разных долях, т. е. не совпадают. Значит, в результате перекраски каждая вершина цепи получила ровно одно инцидентное темное ребро. При этом крайние вершины цепи инцидентных темных ребер ранее не имели, а у остальных вершин такое ребро было одно (так как P — паросочетание), и оно стало светлым при перекраске. Следовательно, после перекраски любой вершине цепи инцидентно ровно одно темное ребро, а для остальных вершин ничего не изменилось. Итак, после перекраски темные ребра действительно образуют паросочетание, и в нем на одно ребро больше, чем в P. □

- Данная лемма объясняет термин «увеличивающая цепь».

На рис. 1 слева изображен граф G, в котором жирными линиями отмечено паросочетание P. Цепь $x_1 \to y_2 \to x_3 \to y_4 \to x_5 \to y_3$ является увеличивающей относительно P; в результате перекраски всех ее ребер получим паросочетание P', изображенное на рис. 1 справа.

Рис. 1

Построение увеличивающих цепей — основной ингредиент *венгерского алгоритма*, решающего задачу о назначениях. Перед изложением алгоритма мы докажем теорему, из которой следует его корректность. Нам потребуется следующее несложное

Замечание

Если в связном графе степень любой вершины равна 1 *или* 2, *то этот граф является либо простой цепью, либо простым циклом.* □

Теорема Бержа

Теорема Бержа

Паросочетание P в двудольном графе G является максимальным тогда и только тогда, когда не существует увеличивающей относительно P цепи.

Доказательство. Необходимость следует из леммы об увеличивающей цепи: если такая цепь относительно P существует, то P не максимально. *Достаточность.* Пусть увеличивающей относительно P цепи не существует и пусть в G есть максимальное паросочетание P', отличное от P (если такого паросочетания нет, то P — максимальное паросочетание, к тому же единственное в G, и доказывать нечего). Рассмотрим подграф G' в G, состоящий из всех ребер, принадлежащих *ровно одному* из паросочетаний P, P', и всех вершин, инцидентных этим ребрам. На рис. 2 слева приведен пример паросочетаний P (жирные линии) и P' (красные линии), а справа — соответствующий подграф G'.

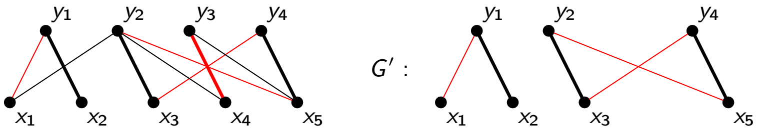


Рис. 2

Теорема Бержа: окончание доказательства

Любая вершина в подграфе G' имеет степень не ниже 1 и не выше 2 (вершина может быть инцидентна максимум одному ребру из P и максимум одному ребру из P'). Согласно сделанному выше замечанию, каждая компонента связности графа G' является простой цепью либо простым циклом.

Любая цепь в G' является чередующейся (как относительно P, так и относительно P'), поскольку ребра из одного паросочетания не имеют общих вершин. В то же время, в G (а значит, и в его подграфе G') нет увеличивающих цепей ни относительно P — по условию, ни относительно P' — по доказательству необходимости в данной теореме. Следовательно, крайние ребра любой цепи в G' принадлежат разным паросочетаниям, а значит, длина такой цепи четна. Циклы имеют четную длину по критерию двудольности (лекция 7). Но тогда в любой компоненте связности графа G' ребер из P столько же, сколько и ребер из P', т. е. получаем $|P \setminus P'| = |P' \setminus P|$, откуда $|P| = |P'|$. Таким образом, паросочетание P максимально. $\square$

Венгерский алгоритм

Вход: двудольный граф G с долями X и Y, $|Y| \leqslant |X|$.
Выход: список ребер максимального паросочетания P в графе G.

1. Выбрать произвольное ребро (x, y), положить $P = \{(x, y)\}$.
2. Положить $F = Y \setminus \{y\}$.
3. Если F пусто, закончить работу, иначе пометить все вершины из F как непросмотренные.
4. Выбрать непросмотренную вершину v из F; если существует увеличивающая цепь с началом в v, перекрасить все ребра этой цепи, удалить v из F и вернуться на шаг 3.
5. Пометить вершину v как просмотренную; если в F остались непросмотренные вершины, вернуться на шаг 4.

Легко проверить, что венгерский алгоритм корректен: он всегда останавливается, строит паросочетание (по лемме об увеличивающей цепи), и это паросочетание максимально (если остановка произошла на шаге 3, то текущее паросочетание насыщает все вершины из Y, а если алгоритм вышел из цикла после шага 5, то в графе нет увеличивающих относительно P цепей и применима теорема Бержа).

Единственная техническая сложность венгерского алгоритма — поиск увеличивающих цепей. Для этого используются деревья поиска, уже упоминавшиеся в лекции 6. Мы продемонстрируем их построение на примере, найдя максимальное паросочетание в графе, изображенном на рис. 1 (он воспроизведен на рис. 3 слева).

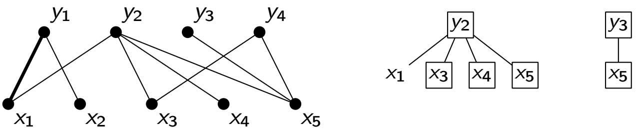


Рис. 3

Пусть на шаге 1 алгоритма выбрано ребро (x_1, y_1), шаги 2 и 3 произведены. Далее, пусть на шаге 4 строится увеличивающая цепь из вершины y_2. Для этого строится дерево с корнем y_2; сыновьями корня становятся все смежные с y_2 вершины. Если среди этих вершин есть свободная (на рис. 3 справа свободные вершины помещены в квадратики), то увеличивающая цепь найдена. Перекраска цепи в данном случае — это занесение ребра (x_3, y_2) в P. После этого вершина y_2 удаляется из F. Следующая итерация выглядит аналогично: в P добавится ребро (x_5, y_3), вершина y_3 будет удалена из F.

После двух итераций паросочетание P выглядит как на рис. 4 слева. На третьей итерации поиск увеличивающей цепи будет производиться из вершины y_4 — единственной оставшейся свободной верхней вершины. При этом на первом шаге построения дерева свободная нижняя вершина найдена не будет; продолжим поиск, сначала дописав в сыновья к каждой из нижних вершин смежную с ней в P верхнюю вершину (цепь должна быть чередующейся), а затем дописав в сыновья эти верхним вершинам все смежные с ними нижние вершины, еще не попавшие в дерево (см. рис. 4 посередине). В этот момент мы найдем свободную вершину x_4 и увеличивающую цепь — единственную (y_4, x_4)-цепь в построенном дереве.

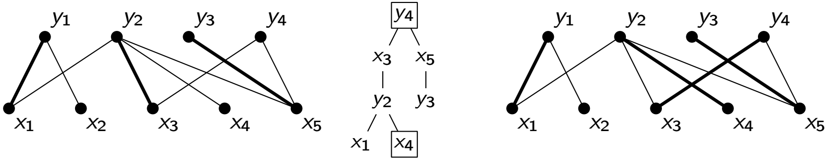


Рис. 4

Перекраска цепи состоит в удалении из P ребра (x_3, y_2) и добавлении в P ребер (x_3, y_4) и (x_4, y_2). Итоговое паросочетание приведено на рис. 4 справа. На следующей итерации алгоритм завершит работу, поскольку свободных верхних вершин не осталось.

Рассмотрим усложненный вариант задачи о назначениях, в котором присутствует функция «выгодности» назначения.

Задача об оптимальном назначении (неформальная постановка)

Имеется n вакантных должностей и n претендентов на них. Каждый из претендентов может быть назначен на любую должность; выгода от назначения i-го претендента на j-ю должность выражается числом w_{ij}. Требуется произвести назначения так, чтобы добиться максимально возможной суммарной выгоды.

Чтобы перевести эту задачу на язык теории графов, нам понадобится

Определение

Двудольный граф, в котором любые две вершины из разных долей соединены ровно одним ребром, называется *полным двудольным*. Через $K_{n,m}$ обозначается полный двудольный граф, доли которого содержат n и m вершин.

Задача об оптимальном назначении: графовая постановка

Построим двудольный граф, в котором, как и при формализации задачи о назначении, нижние вершины $\{x_1, \dots, x_n\}$ будут соответствовать претендентам, а верхние $\{y_1, \dots, y_n\}$ — должностям. Поскольку теперь каждый претендент может быть назначен на любую должность и число претендентов равно числу должностей, мы получим полный двудольный граф $K_{n,n}$. Для каждых $i, j = 1, \dots, n$ ребру (x_i, y_j) припишем вес w_{ij}, т. е. выгоду от соответствующего назначения. Получим взвешенный граф $(K_{n,n}, w)$ (где n — число претендентов, w — весовая функция, задаваемая равенствами $w(x_i, y_j) = w_{ij}$).

Определение

Совершенное паросочетание в графе $(K_{n,n}, w)$, имеющее максимальный вес среди всех совершенных паросочетаний этого графа, называется *оптимальным*.

Задача об оптимальном назначении (графовая постановка)

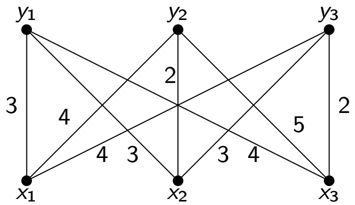
Найти оптимальное паросочетание в заданном взвешенном графе $(K_{n,n}, w)$.

Алгоритм решения этой задачи активно использует венгерский алгоритм.

Определение

Пусть $(K_{n,n}, w)$ — взвешенный граф, $w(x_i, y_j) = w_{ij}$ для всех $i, j = 1, 2, \ldots, n$. Квадратная матрица $W = (w_{ij})$ порядка n называется *матрицей весов* этого графа. Вместо $(K_{n,n}, w)$ мы будем писать $K_{n,n}[W]$.

Так, матрица на рис. 5 справа задает взвешенный граф $(K_{3,3}, w)$ на рис. 5 слева.



$$W = \begin{pmatrix} 3 & 4 & 4 \\ 3 & 2 & 3 \\ 4 & 5 & 2 \end{pmatrix}$$

Рис. 5. Взвешенный двудольный граф и его матрица весов

Определение

Допустимой разметкой взвешенного графа $K_{n,n}[W]$ называется функция, сопоставляющая каждой вершине v графа действительное число $f(v)$ так, что $f(x_i) + f(y_j) \geqslant w_{ij}$ для всех $i, j = 1, 2, \ldots, n$. Число $f(v)$ называется *меткой* вершины v. Граф $K_{n,n}[W]$ с допустимой вершинной разметкой f обозначается через $K_{n,n}[W; f]$.

Замечание

Допустимая вершинная разметка существует в каждом взвешенном графе $K_{n,n}[W]$. Ее можно получить, например, положив

$$f(x_i) = \max_{1 \leqslant k \leqslant n} \{w_{ik}\} \text{ и } f(y_i) = 0 \text{ для всех } i = 1, 2, \ldots, n. \tag{1}$$

Определение

Пусть $K_{n,n}[W; f]$ — взвешенный граф с допустимой разметкой. *Подграфом равенств* графа $K_{n,n}[W; f]$ называется его суграф, определяемый следующим образом: ребро (x_i, y_j) графа $K_{n,n}[W; f]$ является ребром подграфа равенств тогда и только тогда, когда $f(x_i) + f(y_j) = w_{ij}$. Подграф равенств графа $K_{n,n}[W; f]$ мы будем обозначать через $G_{W,f}$.

На рис. 6 слева представлен граф, изображеный на рис. 5, с допустимой разметкой, задаваемой формулами (1) (метки вершин указаны в скобках), а справа — подграф равенств этого графа.

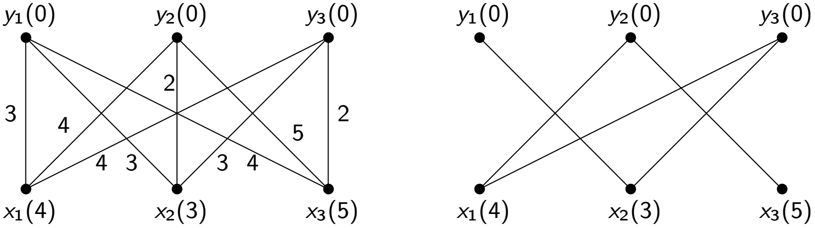


Рис. 6. Граф с допустимой вершинной разметкой и его подграф равенств

Лемма о подграфе равенств

Пусть f — допустимая разметка взвешенного графа $K_{n,n}[W]$. Если подграф равенств графа $K_{n,n}[W;f]$ содержит совершенное паросочетание P, то P является оптимальным паросочетанием в графе $K_{n,n}[W]$.

Доказательство. Пусть P — совершенное паросочетание в графе $G_{W,f}$ (а значит, и в графе $K_{n,n}[W]$), Q — другое совершенное паросочетание в $K_{n,n}[W]$. Напомним, что вес графа — это сумма весов входящих в него ребер. Используя определение допустимой разметки и тот факт, что в совершенном паросочетании каждая вершина графа инцидентна ровно одному ребру, получаем

$$w(Q) = \sum_{(x_i,y_j)\in Q} w(x_i,y_j) \leqslant \sum_{(x_i,y_j)\in Q} (f(x_i)+f(y_j)) = \sum_{(x_i,y_j)\in P} w(x_i,y_j) = w(P),$$

откуда по определению следует максимальность P. $\square$

Алгоритм Куна–Манкреса

Вход: взвешенный граф $K_{n,n}[W]$ с долями X и Y.
Выход: множество ребер оптимального паросочетания P в данном графе.

1. Задать в $K_{n,n}[W]$ произвольную допустимую разметку f и найти подграф равенств $G_{W,f}$.
2. Венгерским алгоритмом найти максимальное паросочетание P в графе $G_{W,f}$ и множество F свободных относительно P вершин доли X.
3. Если $F = \varnothing$, закончить работу.
4. Найти все чередующиеся цепи в графе $G_{W,f}$, начинающиеся в F, положить S и T равными множеству всех вершин доли X (соответственно, доли Y), встретившихся в этих цепях.
5. Если в T нет свободных вершин, положить

$$\Delta = \min_{x_i \in S, y_j \in Y \setminus T} \{f(x_i) + f(y_j) - w_{ij}\}, \tag{2}$$

$f(x) = f(x) - \Delta$ для всех $x \in S$, $f(y) = f(y) + \Delta$ для всех $y \in T$, найти новый граф $G_{W,f}$ и вернуться на шаг 4.
6. Увеличить P, перекрасив найденную увеличивающую цепь, и вернуться на шаг 3.

В соответствии с леммой о подграфе равенств, алгоритм Куна–Манкреса перестраивает исходную разметку так, чтобы найти совершенное паросочетание в подграфе равенств. Следующие замечания обеспечивают корректность алгоритма:

- величина Δ всегда строго положительна (по построению S и T, вершины из S смежны в $G_{W,f}$ только вершинам из T; значит, в (2) для любого ребра (x_i, y_j) имеем $f(x_i) + f(y_j) - w_{ij} > 0$);
- измененная на шаге 5 функция f остается допустимой разметкой (сумма меток вершин, инцидентных ребру, уменьшилась только для ребер, соединяющих вершины из S и $Y \setminus T$, причем величина Δ определена так, чтобы эта сумма осталась не меньше веса ребра);
- ребра, вошедшие в максимальное паросочетание в текущем графе равенств, остаются в графе равенств после изменения разметки (метки увеличились только для вершин из T, но по построению они связаны темными ребрами только с вершинами из S, у которых метки соответственно уменьшились);
- после изменения графа $G_{W,f}$ на шаге 5 множество S не меняется, а во множество T добавляется хотя бы одна вершина (та, на которой достигается минимум в (2); это обеспечивает конечность числа итераций шагов 4–5 между увеличениями паросочетания, а значит, тот факт, что алгоритм остановится).

Алгоритм Куна–Манкреса: пример

Найдем оптимальное паросочетание в графе $K_{5,5}$ с долями $X = \{x_1, x_2, x_3, x_4, x_5\}$ и $Y = \{y_1, y_2, y_3, y_4, y_5\}$ и матрицей весов

$$W = \begin{pmatrix} 1 & 4 & 1 & 1 & 1 \\ 2 & 4 & 1 & 1 & 2 \\ 3 & 5 & 1 & 5 & 2 \\ 1 & 1 & 1 & 4 & 1 \\ 2 & 1 & 2 & 5 & 2 \end{pmatrix}.$$

Зададим допустимую разметку в соответствии с формулами (1): $f(x_1) = f(x_2) = f(x_4) = 4$, $f(x_3) = f(x_5) = 5$, $f(y_1) = \cdots = f(y_5) = 0$. Найдем в графе $K_{n,n}[W; f]$ подграф равенств $G_{W,f}$ (см. рис. 7; возле каждой вершины в скобках указана ее метка).

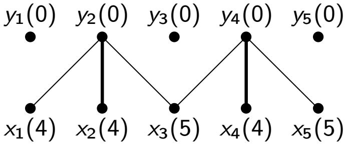


Рис. 7

Пусть P — максимальное паросочетание в графе $G_{W,f}$, выделенное на рис. 7 жирными линиями.

Если $P = \{(x_2, y_2), (x_4, y_4)\}$, то $F = \{x_1, x_3, x_5\}$. При построении чередующихся цепей, начиная с F, мы «захватим» еще вершины y_2, y_4 и x_2, x_4. Таким образом, $S = X$, $T = \{y_2, y_4\}$.

В T нет свободных вершин — по множествам $S = X$ и $Y \setminus T = \{y_1, y_3, y_5\}$ вычисляем $\Delta = 2$. Обновляем метки вершин; новый граф $G_{W,f}$ с текущим паросочетанием выглядит как на рис. 8.

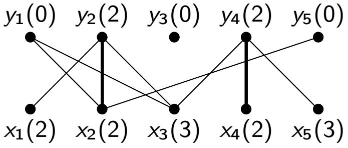


Рис. 8

По-прежнему $F = \{x_1, x_3, x_5\}$. При построении чередующихся цепей немедленно находим свободную вершину y_1, перекрашиваем цепь $x_3 \to y_1$ из одного ребра и возвращаемся на шаг 3.

Теперь $F = \{x_1, x_5\}$; при построении чередующихся цепей находим увеличивающую цепь $x_1 \to y_2 \to x_2 \to y_5$, перекрашиваем ее и возвращаемся на шаг 3, имея $F = \{x_5\}$. Текущее паросочетание изображено на рис. 9 (см. следующий слайд).

Алгоритм Куна–Манкреса: пример (окончание)

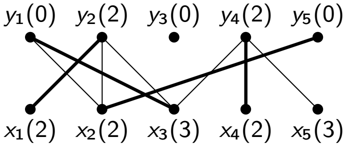


Рис. 9

Имеем $F = \{x_5\}$ и строим $S = X$ и $T = \{y_1, y_2, y_4, y_5\}$. В T нет свободных вершин; согласно (2), находим $\Delta = 1$ и изменяем метки. В новом графе $G_{W,f}$ (см. рис. 10) находим увеличивающую цепь $x_5 \to y_3$ длины 1 (красное ребро на рис. 10) и, перекрасив ее, получаем совершенное паросочетание в текущем подграфе равенств, т. е. оптимальное паросочетание в исходном взвешенном графе.

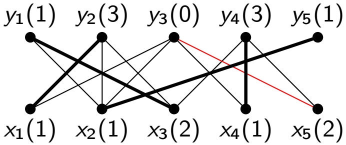


Рис. 10

Представляет интерес интерпретация найденного нами оптимального паросочетания P с точки зрения неформальной постановки задачи об оптимальном назначении. В P входит ребро (x_2, y_5). Это значит, что мы назначили претендента x_2 на должность y_5. В соответствии с матрицей весов

$$W = \begin{pmatrix} 1 & 4 & 1 & 1 & 1 \\ 2 & 4 & 1 & 1 & 2 \\ 3 & 5 & 1 & 5 & 2 \\ 1 & 1 & 1 & 4 & 1 \\ 2 & 1 & 2 & 5 & 2 \end{pmatrix},$$

выгода от этого назначения равна 2. Между тем претендент x_2 мог бы с большей выгодой, равной 4, быть назначен на должность y_2. Аналогично, на должность y_4 назначен претендент x_4. Выгода от этого назначения равна 4, в то время как на должность y_4 можно было бы с выгодой 5 назначить претендента x_3. Вообще, только два претендента (x_1 и x_4) назначены на «наиболее подходящие» для них должности, и только на три должности (y_1, y_3 и y_5) назначены «наилучшие» для них претенденты. Однако суммарная выгода от сделанных назначений, равная 15, является максимально возможной.