

УДК 519.161

ОПТИМИЗАЦИЯ ТРАНСПОРТНЫХ ПЕРЕВОЗОК НА ОСНОВЕ ПОИСКА ЗВЕЗДНЫХ ПОКРЫТИЙ

О.А. Подоляка, доцент, к.т.н., ХНАДУ,
А.Н. Подоляка, ст. преподаватель, НАУ «ХАИ»

Аннотация. Задача назначения пар водителей автобусов на длинные рейсы рассмотрена как задача поиска наибольшего покрытия двудольного графа звездами второй степени. Приведены доказательства ключевых теорем нормализационного метода решения данной задачи.

Ключевые слова: двудольный граф, реберное покрытие, звездное покрытие, паросочетание.

ОПТИМІЗАЦІЯ ТРАНСПОРТНИХ ПЕРЕВЕЗЕНЬ НА ОСНОВІ ПОШУКУ ЗІРКОВИХ ПОКРИТТІВ

О.О. Подоляка, доцент, к.т.н., ХНАДУ,
О.М. Подоляка, ст. викладач, НАУ «ХАИ»

Анотація. Задача призначення пар водіїв автобусів на довгі рейси, розглянута як задача пошуку найбільшого покриття дводольного графа зірками другого ступеня. Наведено докази ключових теорем нормалізаційного методу розв'язання даної задачі.

Ключові слова: дводольний граф, реберне покриття, зоряне покриття, паросполучення.

TRANSPORTATION OPTIMIZATION BASED ON STAR COVERING

O. Podolyaka, assistant professor, cand. eng. sc., KhNAHU,
A. Podolyaka, senior lecturer, NAU «KhAI»

Abstract. Assignment problem of bus drivers pairs on long trips, are considered as a task of searching maximal second degree star covering of a bipartite graph. The base normalizing method theorems of solving this problem are proved.

Key words: bipartite graph, edge covering, star covering, matching.

Введение

В теории графов известны задачи поиска независимых множеств вершин и ребер, а именно, множеств несмежных ребер и вершин. Обычно для задачи поиска реберного покрытия существует эффективный алгоритм решения, а задача поиска вершинного покрытия, в общем случае, является трудно решаемой.

Рассмотренная в работе задача поиска наибольшего звездного покрытия двудольно-

го графа (НЗП) относится к классу задач поиска реберных покрытий графа. В статье показано, что эта задача эффективно решается $O(n^4)$ нормализационным методом. Для начала поясним, что звезда это дерево единичной высоты, его корень называется вершиной, а число его листьев определяет степень звезды h . Наибольшее звездное покрытие графа (НЗП) - это покрытие наибольшего числа вершин графа деревьями заданной степени. Теперь рассмотрим связь звездных покрытий с классическими задачами поиска реберных покрытий.

Независимое множество ребер обычно называют паросочетанием. Его поиск эффективно осуществляется при помощи различных модификаций метода поиска максимального потока. Множество прикладных задач распределения работ и ресурсов в системах, представленных множеством независимых объектов сводится к поиску паросочетаний [1]. Следует отметить, что поиск паросочетания эквивалентен поиску покрытия графа звездами первой степени. Следовательно, задача поиска независимого множества ребер – это вырожденный случай задачи поиска НЗП. Таким образом, звездные покрытия обобщают класс задач, сводимых к поиску паросочетаний. А именно: задача о назначениях (ЗН) и задача о назначениях на узкое место, поиска системы различных представителей (трансверсаль), поиска цепи и антицепи, поиска наименьшего вершинного покрытия двудольных графов. Теория паросочетаний достаточно хорошо исследована и представлена множеством книг и работ. Огромный вклад в развитие теории паросочетаний внесли такие всемирно известные ученые как: Кёниг Д. (Kernig D.), Эгервари Э. (Egerváry E), Кун Харольд. (Kuhn H.), Татт Уильям. (Tutte W.T.), Бёрж Клауд. (Berge C.), Форд Л. (Ford L. R.), Фалкерсон Д. (Fulkerson D.R.), Рабин М. (Rabin M. O.) и Вазирани В. (Vazirani V. V.), Эдмондс Джек (Edmonds J.) и др.

Необходимо отметить, что научный интерес представляет случай поиска покрытия графа звездами, степень которых больше единицы. Поскольку эту задачу при $h > 1$ невозможно непосредственно решить методом поиска потока [2], т.к. степень звезды означает разномножение потока в вершине графа сети. Закон сохранения потока соблюдается только в задаче поиска звездного покрытия первой степени. Важно отметить что, классические потоки обобщают паросочетания, но не обобщают звездные покрытия.

Следует сказать, что задача поиска НЗП эквивалентна транспортной задаче, в которой степени звезд определяют потребности потребителей, а ресурсы всех поставщиков считаются равными единице.

Прикладной интерпретацией звезды является производительность машины в единицах обрабатываемого ресурса. В работе [3] отмечено, что задача поиска покрытия звездами

разной степени (машин различной производительности) сводится к задаче поиска НЗП (звездами одинаковой степени). Необходимо также отметить, что математическая модель задачи поиска НЗП характеризуется типизацией звезд по степени. Например, в задаче покрытия графа звездами разной степени – каждой звезде соответствует машина определенной производительности. Следовательно, звездные покрытия развивают теорию моделей систем, представленных независимыми машинами различной производительности.

Следует отметить, что каждая звезда покрытия при $h > 1$ представляет собой «зависимое» множество ребер, т.к. все ребра звезды имеют одну общую вершину. Тем не менее, все звезды покрытия независимы по отношению друг к другу, поскольку не имеют общих вершин. Важно отметить, что задача такого вида в дискретной математике пока строго не классифицирована.

Задача поиска регулярного подграфа также может быть сведена к поиску звездного покрытия, поскольку множество ребер звезды независимо. Регулярный граф с вершинами степени k называют также k -фактором. В этом смысле регулярный граф первой степени – это паросочетание, второй – это цикл, и т.п. Метод ветвей и границ поиска гамильтонова цикла может быть основан на поиске 2-фактора, поскольку фактор определяет нижнюю границу веса гамильтонова цикла. В этой связи заслуживает внимания теорема Нэша-Вильямса, которая связывает существование гамильтонова цикла с наличием k -фактора у графа с $2k+1$ вершинами. Регулярные графы также определяют реберную раскраску графа, т.к. определение раскраски эквивалентно разложению графа на k непересекающиеся паросочетаний. Данное разложение для двудольных графов отражено в теореме Кенига. Связь k -факторов и дважды стохастических матриц прослеживается в теореме Биркгофа.

Таким образом, исследование звездных покрытий представляет огромный научный и практический интерес. Они обобщают целые классы естественнонаучных задач теории графов, теории множеств и дискретной математики. Множество прикладных задач оптимизации функционирования систем, представленных независимыми машинами раз-

личной производительности может быть сведено к математической модели задачи поиска НЗП.

Данная статья является продолжением цикла публикаций, в которых рассматривается метод решения оптимизационной задачи поиска наибольшего звездного покрытия (НЗП) двудольного графа и ряда задач, сводимых поиску НЗП [1]. Следует сказать, что в работе [3] представлена общая схема метода поиска НЗП и приведены ключевые теоремы – теоремы запрещений, нормализации и лемма останова. По названию основной теоремы метод поиска звездного покрытия был назван нормализационным. Следует отметить, что в сети есть публикации, в которых задача поиска НЗП ошибочно относится к классу NP. Поэтому основная цель настоящей статьи – привести доказательство теорем запрещений и нормализации. Наличие доказанных теорем дает право считать нормализационный алгоритм корректным и полиномиальным. Другая задача публикации – дать максимально простое объяснение работы этого алгоритма на примере решения задачи малой размерности. Т.е. изложить принципы работы алгоритма на уровне ручного счета.

Перейдем к рассмотрению, задачи поиска покрытия взвешенного двудольного графа $G(V_1, V_2, E)$ звездами одинаковой степени.

Математическая модель задачи поиска НЗП

h-звезда – это связный двудольный граф $G_{1,h}$, одна из долей которого имеет степень равную h и называется вершиной звезды, а каждая из вершин другой доли – единичную степень. Звезду можно также рассматривать как дерево, у которого количества ребер и листьев равны степени звезды. В графе $G(V_1, V_2, E)$ вершины V_1 – это вершины звезд, а V_2 – листья. *Звездным покрытием* (ЗП) графа назовем множество $E_h \in E$ ребер графа, инцидентных звездам, степень которых не превосходит h . Вес звездного покрытия определяется как сумма весов всех его ребер.

$$w(E_h) = \sum_{e \in E_h} w(e) \quad (1)$$

Наибольшим h -звездным покрытием (НЗП)

графа назовем h -звездное покрытие $E'_h \in E$ максимальной мощности, т.е. для любого $E'_h, |E'_h| \geq |E_h|$. НЗП, покрывающее все вершины графа назовем совершенным ЗП E_h^* . Множество всех возможных наибольших h -звездных покрытий графа обозначим E_h^{all} . В рассматриваемой задаче первой доли исходного графа соответствуют вершины звезд, а вершины второй доли – это листья звезд. Для удобства будем считать, что количества вершин второй V_2 и первой доли V_1 кратны степени звезды, т.е.

$$|V_2| \geq |V_1|, |V_2| = h \cdot |V_1|, h \in N. \quad (2)$$

Выполнение этого условия всегда можно обеспечить путем добавления недостающего количества фиктивных вершин и ребер. Пусть оптимальным считается звездное покрытие сумма весов ребер которого минимальна.

$$w(E_h^*) = \min_{E_h \in E_h^{all}} [w(E_h)] \quad (3)$$

Это функционал задачи поиска в двудольном графе НЗП минимального веса.

Нужно отметить, что прикладной интерпретацией звезды может быть машина (автомобиль, станок, процессор). Например, степень звезды может означать число посадочных мест (задач, процессов и т.п.). Поэтому задача поиска НЗП находит множество практических приложений. А именно, в транспортной логистике, задачах распределения: работ, задач, ресурсов и т.п.

Необходимо также отметить, что математическая модель задачи поиска характеризуется типизацией звезд или вершин первой доли по мощности. Например, в задаче покрытия графа звездами разной степени – каждой звезде соответствует определенный тип машины [2,3]. Следовательно, звездные покрытия развивают теорию моделей систем, представленных параллельно функционирующими объектами различной производительности.

Теперь рассмотрим постановку прикладной оптимизационной задачи поиска НЗП.

Практическая задача

Автопредприятию для организации перевозки пассажиров по пяти международным маршрутам необходимо назначить пары водителей автобусов. Для каждого водителя известна требуемая заработная плата за выполнение каждого рейса. Цель руководства - назначить наибольшее число водителей на маршруты и минимизировать затраты предприятия по заработной плате.

Исходные данные задачи представлены матрицей двудольного графа β размера (5×10) , в которой строки I означают автобусы, а столбцы J представляют водителей. Элемент матрицы означает зарплату соответствующего водителя за рейс. Ребра оптимального НЗП $\{(1,5), (1,9), (2,3), (2,10), (3,6), (3,7), (4,2), (4,4), (5,1), (5,8)\}$ выделены зеленым, их нам предстоит найти.

I/J	1	2	3	4	5	6	7	8	9	10
1	29	38	48	39	23	21	21	27	17	41
2	46	19	7	49	30	34	22	37	23	14
3	40	23	38	32	24	8	12	37	9	37
4	14	17	31	26	27	16	28	36	19	33
5	10	20	24	45	32	27	17	21	37	39

Рис. 1. Исходная матрица β

Вот двудольный граф задачи, в котором выделены ребра оптимального НЗП.

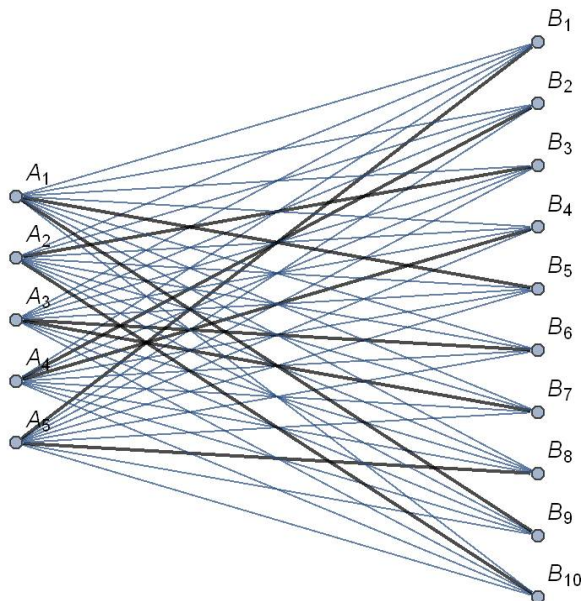


Рис. 2. Двудольный граф задачи.

Замечания

1. Единственному решению в матрице данной задачи соответствует: по два элемента (пара водителей) в каждой строке и по одному в каждом столбце.

2. Если водитель не может выполнить рейс, то ему назначается бесконечная зарплата. В программе такой зарплате соответствует большое число.

3. Выполнение второго пункта означает, что граф задачи всегда будет полным, и поэтому его НЗП является совершенным. Однако, данное совершенное покрытие может содержать фиктивные ребра большого (бесконечного) веса. В исходной задаче, такое покрытие не будет совершенным. Следовательно, могут существовать оптимальные решения, в которых не все водители назначены на рейсы. Т.е. возможны пустые рейсы или рейсы с одним водителем.

4. Каждое оптимальное назначение (НЗП) состоит из наибольшего числа водителей, т.к. содержит минимальное число тяжелых фиктивных ребер.

5. Если оптимальное покрытие не является совершенным в исходном графе, то в некоторой мере теряется цель оптимизации, поскольку, автобусы с одним водителем при длительных маршрутах потенциально опасны для пассажиров.

6. Требование полноты звезд оптимального НЗП, делает задачу NP-полной [2]. Полнота звезд означает, что предприятию необходимо перевести наибольшее число пассажиров минимальным числом автобусов. Этому случаю соответствует один из видов другой сложной задачи, которая называется задача о покрытии множеств (у Гери и Джонсона ТП-3).

7. Может показаться, что эта задача поиска НЗП похожа на задачу о назначениях. Это впечатление глубоко ошибочно. Модель данной задачи существенно сложнее ЗН [4]. Матрица задачи поиска НЗП прямоугольная размера $(n, n \cdot h)$, а у ЗН квадратная. Модель НЗП-задачи более точная т.к. в ней есть оценка каждого отдельного водителя, а в ЗН – целой пары. Если просуммировать зарплаты всевозможных сочетаний пар водителей, а затем представить сочетания пар в виде матрицы - получится ЗН. Проблема в том, что эта матрица будет экспоненциального размера.

Перейдем к рассмотрению метода решения этой задачи.

Нормализационный метод

Нормализация - процедура уменьшения весов ребер графа не принадлежащих оптимальному НЗП. Поскольку выполняется минимизация - эти ребра называются тяжелыми.

Пусть, *линия* – означает строку или столбец матрицы смежности графа.

В ходе работы алгоритма полиномиальное число раз выполняется процедура нормализации, в которой вычисляется разность пары линий матрицы смежности графа. Уменьшаемая *линия* называется *клиентской*, а вычитаемая – *серверной*. В программе линия представлена одномерным массивом или вектором, в математике линия является мультимножеством, т.к. может содержать повторяющиеся элементы. Разность линий названа *мультимножеством линейных пар* (МЛП). Каждую *Линейную пару* (ЛП) мультимножества образуют три поля: элемент клиентской строки и элемент серверной строки и индекс столбца линии этих элементов в матрице. Вес ЛП – это разность клиентского и серверного значений.

Равенство линий

Линии называются *линейно равными* или *эквивалентными*, если равны веса всех линейных пар мультимножества этих линий. Линейно равные линии образуют класс эквивалентности, его определяет множество индексов эквивалентных линий матрицы смежности графа [3,4]. Мощность класса эквивалентности серверной линии $|S_{=}|$ и степень звезды h - определяют грань МЛП, которая называется нормализатор. Процедура нормализации уменьшает клиентские значения ЛП вес которых больше нормализатора. Указанные клиентские значения - это тяжелые ребра графа, которые не могут формировать оптимальное НЗП. После нормализации все максимальные элементы МЛП будут равны весу нормализатора.

Необходимо отметить, что в процессе нормализации образуются эквивалентные линии, т.к. выравниваются веса линейных пар. Следовательно, в ходе работы алгоритма происходит объединение классов эквивалентных линий. Когда все линии становятся эквивалентными работа алгоритма заканчивается

(см. лемму останова [3,4]).

Рассмотрим линейное равенство на примере некоторой эквивалентной подматрицы $L(3,6)$ матрицы смежности двудольного графа $G(V_1, V_2, E), |V_2| = 6$.

0	8	5	9	6	7	5
-2	10	7	11	8	9	7
4	4	1	5	2	3	1

Рис. 3. Подматрица $L(3,6)$

В левом столбце указаны линейные коэффициенты, при вычитании которых из элементов первой строки можно вычислить элементы всех эквивалентных строк. Нужно отметить, что в программной реализации алгоритма рационально хранить только одну строку, множество индексов ее эквивалентных линий и множество линейных коэффициентов.

Замечание: Мощность класса эквивалентности отдельной линии равна единице.

Нейтрализация эквивалентных линий – приведение подматрицы эквивалентных линий к нулевой форме. Она осуществляется в два этапа: первый – вычитание любой равной строки из всех строк матрицы графа, и второй - вычитание из матрицы произвольного столбца. Нейтрализация изменяет вес оптимального НЗП, но не изменяет множество его ребер.

Нейтральная матрица $L_0(3,5)$ матрицы $L(3,5)$, $L[i,j] = -(V[i]+U[j])$:

$V \setminus U$	-4	-1	-5	-2	-3	-1
-4	0	0	0	0	0	0
-6	0	0	0	0	0	0
0	0	0	0	0	0	0

Рис. 4. Нейтральная матрица $L_0(3,5)$

Следует отметить, что коэффициенты $V[i]$ и $U[j]$ в разделе исследование операций называют *потенциалами*. Вспомним, например, широко известный метод потенциалов решения транспортной задачи или венгерский алгоритм решения ЗН.

Если все линии равны, то матрица смежно-

сти двудольного графа эквивалентна нулевой матрице. Вес любого оптимального НЗП нейтральной матрицы можно определить, как

$$w(E_h^*) = - \left[\sum_{i \in I} U_i + h \cdot \sum_{j \in J} V_j \right].$$

Нормализация недопустимых ребер при помощи МЛП

МЛП используется для проверки совместности пар ребер, которые связывают заданное подмножество вершин одной доли со всеми вершинами другой доли графа. Очевидно, чтобы построить НЗП меньшего веса следует минимизировать веса парных сочетаний его ребер.

Рассмотрим пример МЛП L^{cs} клиентской строки c и серверной строки s матрицы смежности β некоторого двудольного графа $G_{3,6}$, $h = 2$.

J	1	2	3	4	5	6
c	3	8	15	8	5	7
s	8	5	9	6	7	5
$L^{cs} = \beta_c - \beta_s$	-5	3	6	2	-2	2

Рис. 5. МЛП строки c и серверной строки s матрицы β

Для определения допустимых пар ребер НЗП используется следующая лемма.

Лемма о допустимой паре ребер МЛП

Допустимую пару для ребра β_{ca} клиентской линии L^{cs} формируют ребра β_{sb} , $b \neq a$ серверной линии u которых $L_b^{cs} \geq L_a^{cs}$, $a \in J$, $b \in J$.

$$\text{Т.к.: } L_b^{cs} \geq L_a^{cs} \leftrightarrow \beta_{cb} - \beta_{sb} \geq \beta_{ca} - \beta_{sa} \leftrightarrow \beta_{cb} + \beta_{sa} \geq \beta_{ca} + \beta_{sb}$$

Ребра, у которых $L_b^{cs} = L_a^{cs}$ формируют эквивалентные пары. Следовательно, для ребра $\beta[c,4]=8$ допустимую пару могут составить ребра линии s , ЛП которых $L_j^{cs} \geq L_4^{cs}$ т.е. $L_j^{cs} \geq 2$. Поэтому ребра $\beta[c,4]=8$ и $\beta[s,5]=7$ являются не совместимыми т.к. вершины (c, s) и вершины $(4, 5)$ можно соединить парой

ребер меньшего веса: $\beta[c,5]=5$ и $\beta[s,4]=6$. Следует отметить, что ребра данных пар формируют замкнутую чередующуюся цепь $\{(c,4), (4,s), (s,5), (5,c)\}$, четные ребра которой определяют выигрышную пару. Величину выигрыша можно также определить как расстояние между линейными парами: $d_{3,4} = L_4^{cs} - L_5^{cs} = 2 - (-2) = 4$.

Обратите внимание, что для клиентского ребра $\beta[c,3]=15$, которому соответствуют максимальная ЛП $L_3^{cs} = 6$, вообще, не существует допустимой пары в серверной линии s (т.к. $L_j^{cs} \geq L_3^{cs}$, $L_j^{cs} = \emptyset$, $j = \emptyset$). Поэтому, это ребро не может входить в оптимальное НЗП и его можно удалить из графа. Недопустимые ребра также называются тяжелыми.

Ребро $\beta[c,2]=8$, которому соответствует вторая максимальная ЛП $L_2^{cs} = 3$ также можно удалить из графа. Т.к. для него есть только одно допустимое ребро $\beta[s,3]=9$ в серверной линии s , а необходимо – два (т.к. граф $G_{3,6}$ нужно покрыть звездами второй степени). Очевидно, что верхняя граница веса недопустимого ребра – это бесконечность. Возникает вопрос – какова нижняя граница веса недопустимого ребра, которая гарантирует сохранение веса оптимального НЗП в исходном графе?

Прежде чем дать ответ на этот вопрос, перечислим ряд важных моментов, которые следует учитывать при вычислении нижней границы веса недопустимых ребер.

Во-первых, назначение тяжелым ребрам нового меньшего веса, не должно допускать их удаление с помощью данного МЛП. Н-р, если ребру $\beta[c,3]$ назначить вес 13, то оно снова образует максимальную ЛП в данном множестве и значит станет недопустимым. Эту идею можно использовать для доказательства теоремы нормализации.

Во-вторых, уменьшение веса тяжелых ребер должно означать допустимость этих ребер для формирования ЗП. Т.е. для каждого нормализованного ребра должно быть достаточное число ребер в серверной линии для формирования, совершенного ЗП.

И в-третьих – после назначения тяжелым ре-

брам меньшего веса, они должны формировать в МЛП максимальные линейные пары, в противном случае может уменьшиться вес оптимального ЗП в заданном графе.

На основании вышесказанного можно сделать важный вывод – недопустимые в данном МЛП ребра $\beta[c,3]=15$ и $\beta[c,2]=8$ должны формировать ЛП равные третьей по величине паре мультимножества (т.е. $L_4^{cs} = L_6^{cs} = 2$). Эта линейная пара называется нормализатором. Следовательно, веса ребер должны равняться: $\beta[c,2]=7$ и $\beta[c,3]=11$.

МЛП вершин c и s примет вид.

J	1	2	3	4	5	6
c	3	7	11	8	5	7
s	8	5	9	6	7	5
$L^{cs} = \beta_c - \beta_s$	-5	2	2	2	-2	2

Рис. 6. МЛП после нормализации

Обратите внимание, что отрицательные элементы мультимножества являются двумя максимальными в МЛП разности строк s и c . После уменьшения весов ребер строки s их значения будут равны: $\beta[s,1]=1$ и $\beta[s,5]=3$, и, значит, МЛП будет формировать две линейно равные или эквивалентные строки.

Теперь мы наглядно убедились, что в ходе нормализации образуются эквивалентные линии. Однако, в этом примере нам сразу удалось сформировать две эквивалентные линии, потому что мы рассмотрели малый граф. Для произвольного графа большой размерности, рассмотренная нормализация недостаточна для построения эквивалентных линий, поскольку для этого нужно уменьшать веса большего числа ребер. Теперь пришло время дать ответы на ряд взаимосвязанных вопросов:

1. Зачем, вообще, нужно строить эквивалентные линии?
2. Можно ли нормализовать веса более h ребер в МЛП клиентской строки матрицы графа?
3. Можно ли строить МЛП столбцов и уменьшать на их основе веса ребер?

Сначала, дадим ответы на первые два вопроса. Во-первых, ранее отмечалось, что эквивалентные линии нужно определять, чтобы знать, когда остановить алгоритм. Во-вторых, мощность класса эквивалентных ли-

ний определяет число ребер, которые можно нормализовать. Забегая вперед, отметим, что число тяжелых ребер, которые можно нормализовать в клиентской строке равно $h \cdot |S_{\equiv}|$, где h – степень звезды, а $|S_{\equiv}|$ – мощность класса эквивалентности серверной строки. Поскольку, в рассмотренном примере $|S_{\equiv}|=1$, мы могли нормализовать два ребра.

Теперь ответим на третий вопрос – для завершения работы нормализационного алгоритма необходимо строить МЛП как строк, так и столбцов, иначе возможно зацикливание. Оно происходит, когда тяжелые ребра формируют НЗП. В данной ситуации поиск новых тяжелых ребер будет невозможен. Переход от МЛП строк к МЛП столбцов решает проблему зацикливания, т.к. в ходе построения МЛП столбцов будут найдены эквивалентные столбцы. Кроме того, при ортогональном повороте – переходе от строк к столбцам и наоборот, осуществляется нейтрализация пар тяжелых и легких ребер текущего мультимножества (т.к. $\max_{c_i} - \max_{c_j} \approx 0$ и $\min_{s_i} - \min_{s_j} \approx 0$). Ортогональный поворот можно применить для построения эффективного алгоритма поиска паросочетания.

После наглядного примера перейдем к формализации процесса удаления и нормализации недопустимых ребер. Сначала сформулируем условия существования совершенного звездного покрытия степени h в невзвешенном двудольном графе.

Теорема 1. Обобщенная теорема Холла существования совершенного ЗП

Совершенное ЗП существует, тогда и только тогда когда для каждого подмножества вершин $\forall X \subset V_1$ графа $G_{n,h*n}(V_1, V_2, E)$ выполняется неравенство $|\Gamma(X)| \geq h \cdot |X|$, где $\Gamma(X) \subset V_2$ – множество смежных вершин для вершин множества X .

Замечание. В классической теореме Холла $h=1$.

Удаление тяжелых ребер

Теперь объединим обобщенную теорему Холла, эквивалентные линии и лемму о допустимой паре ребер МЛП, которая накладывает ограничения на парные сочетания ребер взвешенного звездного покрытия. И в

процессе данного объединения докажем теорему запрещений и нормализации. Сформулируем эти теоремы.

Пусть:

L^{cs} – МЛП клиентской линии c и серверной линии s матрицы β двудольного графа $G_{n,h*n}(V_1, V_2, E)$; $L_j^{cs} = \beta_{cj} - \beta_{sj}$ – линейная пара вершины j , $L_j^{cs} \in L^{cs}$; $|S_{\pm}|$ – мощность класса эквивалентности серверной линии L^{cs} , $s \in S_{\pm}$; t – количество тяжелых ребер клиентской линии c МЛП L^{cs} , которые можно удалить из графа.

$$t = \begin{cases} h \cdot |S_{\pm}|, & \text{если } s \text{ – строка} \\ \left\lfloor \frac{|S_{\pm}|}{h} \right\rfloor, & \text{если } s \text{ – столбец} \end{cases} \quad (5)$$

$\max(L^{cs}, n)$ – функция определения значения n -го по величине элемента мультимножества L^{cs} . Начальным (максимальным) элементом считается первый. $L_{norm}^{cs} = \max(L^{cs}, t+1)$ – нормализатор, $L_{norm}^{cs} \in L^{cs}$ (это $(t+1)$ -я по весу линейная пара в отсортированном по убыванию МЛП).

Теорема 2. Строгая теорема запрещений

Удаление из графа G ребра $(c, j) \in E$, у которого $L_j^{cs} > L_{norm}^{cs}$ не изменяет оптимальное НЗП.

Приведем компактную формулировку этой теоремы.

$$\forall j, L_j^{cs} > L_{norm}^{cs} \rightarrow \beta_{cj} = \infty$$

Уменьшение веса тяжелых ребер графа выполняется на основе теоремы нормализации. Она является обобщением теоремы запрещений.

Теорема 3. Нестрогой нормализации

$$\begin{aligned} \forall j, L_j^{cs} > L_{norm}^{cs} \rightarrow \\ \rightarrow \beta_{cj} = \beta_{sj} + \max(L^{cs}, t+1) = \beta_{sj} + L_{norm}^{cs} \end{aligned} \quad (6)$$

Формулу (6) назовем нормализационным оператором.

В МЛП может быть несколько равных по весу нормализаторов. Таким образом, нормализатор – это грань, которая делит мультимножество на два подмультимножества: пер-

вое мощности $|S_{\pm}|$ состоит из больших или равных нормализатору линейных пар, и второе – мощности $|L^{cs}| - |S_{\pm}|$, состоящее из пар меньших или равных нормализатору.

Замечание: Поскольку вместо $|S_{\pm}|$ одинаковых МЛП L^{cs} , $s \in S_{\pm}$ рационально вычислять только одно, вычислительную сложность нормализационного оператора для каждой отдельной серверной строки s можно считать линейной. Также можно считать, что $\max(L^{cs}, t+1)$ эквивалентен по сложности $|S_{\pm}|+1$ поисков $\max(L^{cs}, 1)$.

Докажем теорему 2 (запрещений).

Будем считать данный взвешенный двудольный граф $G_{n,h*n}(V_1, V_2, E)$ полным. Если исходный граф не полный, то отсутствующие в нем ребра всегда можно заменить на очень тяжелые ребра. Очевидно, что полный граф всегда имеет совершенное звездное покрытие.

Произвольной линии матрицы смежности соответствует вершина графа G и ребра, соединяющие смежные ей вершины. Поскольку линия $c \in C_{\pm}$ или $s \in S_{\pm}$ МЛП может быть как строкой, так и столбцом матрицы смежности, обозначим J – множество линий (вершин) противоположной доли графа ($|J| = |V_1| \vee |V_2|$).

Минимальное число вершин t противоположной доли для $|S_{\pm}|$ серверных линий (вершин), которые можно покрыть полными звездами степени h , согласно обобщенной теоремы Холла, определим по формуле (5).

Выделим в клиентской линии $c \in C_{\pm}$ МЛП L^{cs} подмножество E' , $|E'| \leq t$ тяжелых ребер, вес линейных пар $L_{E'}^{cs}$, которых больше значения $(t+1)$ -ой пары L_{norm}^{cs} (то есть, $L_j^{cs} > L_{norm}^{cs}$, $(c, j) \in E'$). Обозначим J' , $j \in J'$, $J' \subset J$ множество вершин противоположной доли тяжелых ребер E' .

Проверим существование совершенного звездного покрытия по теореме 1 для вершин

J' и $(t+1) = |c \cup S|$ линий (вершин) данного МЛП.

Согласно лемме о допустимой паре: для каждого тяжелого ребра $(c, a) \in E'$ клиентской линии $c \in C_{=}$ пару могут составить ребра серверной линии (s, b) ведущие в вершины $b \in J'$, у которых $L_b^{cs} \geq L_a^{cs}, b \neq a, \forall s \in S_{=}$, $(c, a) \in E', (c, b) \in E'$. Очевидно, что в МЛП L^{cs} каждой серверной линии $s \in S_{=}$ множество J' одинаково, т.к. линии эквивалентны. Необходимо отметить, что для подмножества вершин $J', |J'| \leq t$ тяжелых ребер E' МЛП L^{cs} не соблюдается условие обобщенной теоремы Холла. Т.к. это МЛП, образуют $|S_{=}|$ серверных линий и одна дополнительная клиентская линия c . Значит, невозможно построить совершенное звездное покрытие, используя ребра E' , которое связывает $(|S_{=}|+1)$ вершину c $|J'| \leq t$ вершинами противоположной доли. Следовательно, тяжелые ребра не могут формировать совершенное ЗП и их можно удалить из графа.

Доказательство теоремы 3.

После нормализации тяжелых ребер (c, j) , $\forall j, L_j^{cs} > L_{norm}^{cs}$, по формуле (6), в МЛП L^{cs} будет как минимум $(t+1)$ максимальная ЛП равная по весу нормализатору, т.е. $\forall j, L_j^{cs} = L_{norm}^{cs}$, $(\beta_{cj} - \beta_{sj}) = L_{norm}^{cs}$. Значит после изменения весов ребер по формуле (6), запрещение других тяжелых ребер $(c, j) \in E'$ станет невозможным т.к. неравенство $L_j^{cs} > L_{norm}^{cs}$ теоремы 3 строгое. Отметим, что если вес любого нормализованного ребра увеличить на бесконечно малое величину ε , то это ребро можно будет снова удалить и уменьшить его вес на ε по теореме 3. Значит формула (6) находит правильную нижнюю оценку веса запрещенного ребра.

Замечания:

1. Следует отметить, что с точки зрения теории алгоритмов процедуру запрещения тяжелых ребер можно рассматривать как независимую подзадачу, т.к. в МЛП задействовано ограниченное подмножество вершин гра-

фа. Поэтому можно утверждать, что поиск НЗП осуществляется на основе объединения решений независимых подзадач.

2. Обратите внимание, что в программных массивах индексация начинается с нуля, а в теоремах 2 и 3 с единицы, поскольку в теоремах мы имеем дело со счетными величинами, а не адресами компьютера.

3. Теорема 2 запрещений названа строгой поскольку ее неравенство строгое. Несложно сформулировать и доказать нестрогую теорему запрещений ($L_j^{cs} \geq L_{norm}^{cs}$), которая позволяет удалить ровно t тяжелых ребер (т.е. удалять недостающие ребра равных ЛП). Нестрогая теорема запрещений, будет полезна, если необходимо найти только одно оптимальное решение.

4. При вычислении веса тяжелых ребер на основании строгой нормализации нужно учитывать бесконечно малые ε в формуле (6). Анализ сочетаний бесконечно малых, или запрещений ребер в ЛП мультимножеств, дает возможность получить бонусные теоремы нормализации. Они позволяют нормализовать более t тяжелых ребер, т.е. ускорить сходимость алгоритма. Теорема 3 является не строгой т.к. в ней не учитываются значения ε .

Пример нормализации

Рассмотрим пример МЛП L^{cs} клиентской строки c и серверной строки $s \in S_{=}$, $|S_{=}| = 2$ матрицы β некоторого двудольного графа $G_{4,8}$, $h = 2$.

J	1	2	3	4	5	6	7	8
c	-4	3	-2	7	3	-5	6	3
s	0	0	0	0	0	0	0	0
s	0	0	0	0	0	0	0	0
x								

Рис. 7. Пример нормализации МЛП

Поскольку две серверные строки нейтральны, строка c - это МЛП L^{cs} . Число тяжелых ребер L^{cs} равно $t = h \cdot |S_{=}|$ т.е. $t = 2 \cdot 2 = 4$.

Обратите внимание, что в нашем примере: $|E'| = |J'| = 2$, $J' = \{4, 7\}$, $J' \subset J$ а $t = 4$, $L_{norm}^{cs} = \{L_2^{cs}, L_5^{cs}, L_8^{cs}\}$. Поэтому, по строгой теореме можно уменьшить вес двух, а не четырех ребер, т.к. в МЛП L^{cs} имеется три равных по

весу нормализатора.

Важно отметить, что нормализация тяжелых ребер строки c на основании L^{cs} не зависит от весов элементов строки x .

Теперь можно перейти к рассмотрению алгоритма.

Схема нормализационного алгоритма

a) while (все линии не эквивалентны)

b) Рациональный перебор пар параллельных линий $\{c,s\}$ матрицы смежности графа β

for($\{c,s\} \in \beta$)

1. Вычисление разности линий c и s , формирование мультимножества L^{cs}

$$L^{cs} = \beta_c - \beta_s, \quad s \in S_-, L_j^{cs} = \beta_{cj} - \beta_{sj}$$

2. Определение числа удаляемых ребер клиентской линии c

$$t = \begin{cases} h \cdot |S_-|, & \text{если } s - \text{строка} \\ \left\lfloor \frac{|S_-|}{h} \right\rfloor, & \text{если } s - \text{столбец} \end{cases}$$

3. Поиск нормализатора.

$$L_{norm}^{cs} = \max(L^{cs}, t+1)$$

4. Нормализация тяжёлых ребер $(c, j) \in \beta, \forall j, L_j^{cs} > L_{norm}^{cs}$

$$\beta_{cj} = \beta_{sj} + L_{norm}^{cs}$$

5. Проверка равенства линий МЛП c и s . Объединение равных линий в множестве S_- линии s

$$\text{if}(c == s) \text{ then } S_- = C_- \cup S_-$$

Пример решения задачи назначения пар водителей на рейсы в MS Excel

Исходная матрица двудольного графа β нашей задачи представлена на рисунке 8.

I/J	0	1	2	3	4	5	6	7	8	9	V	Norm
1	0	0	0	0	0	0	0	0	0	0	0	0
2	17	-19	-41	10	7	13	1	10	6	-27	0	10
3	11	-15	-10	-7	1	-13	-9	10	-8	-4	0	1
4	-15	-21	-17	-13	4	-5	7	9	2	-8	0	4
5	-19	-18	-24	6	9	6	-4	-6	20	-2	0	6
U	-29	-38	-48	-39	-23	-21	-21	-27	-17	-41		

Рис. 9. Вычитание строки 1 из матрицы β

U, V – линейные коэффициенты (потенциалы) строк и столбцов $\beta_{ij} = \beta_{ij} + U_j + V_i$. В конце алгоритма мы должны получить нейтральную (нулевую) матрицу и массивы коэффициентов U и V . Они являются решением задачи. Если сумма коэффициентов U и V элемента нейтральной матрицы меньше соответствующего элемента исходной матрицы, то этот элемент не принадлежит оптимальному НЗП.

I/J	1	2	3	4	5	6	7	8	9	10	V
1	29	38	48	39	23	21	21	27	17	41	0
2	46	19	7	49	30	34	22	37	23	14	0
3	40	23	38	32	24	8	12	37	9	37	0
4	14	17	31	26	27	16	28	36	19	33	0
5	10	20	24	45	32	27	17	21	37	39	0
U	0	0	0	0	0	0	0	0	0	0	0

Рис. 8. Исходная матрица β

В ходе алгоритма будем вычитать выбранную серверную линию из всех линий матрицы, а также советующих линейных коэффициентов. Вычитание массивов (линий) в Excel реализуется при помощи пометки серверных строк и столбцов символом \$. После вычитания серверная линия и равные ей линии станут нейтральными, а все остальные линии матрицы будут мультимножествами этих равных серверных линий. В каждой линии определим нормализатор Norm и уменьшим веса всех элементов линии, которые больше Norm. Для нахождения нормализатора нужно воспользоваться функцией Excel НАИБОЛЬШИЙ(диапазон(линия);k), которая определяет k-ое наибольшее значение линии. В нашей задаче $k=t+1$ - это число равных серверных линий МЛП плюс единица.

Вычтем строку 1 из всех строк β , $\beta_i = L^{i,1} = \beta_i - \beta_1, i \in I$.

В крайнем правом столбце Norm указаны значения нормализаторов, соответствующих мультимножеств. Нормализатором i -го МЛП $L^{i,1}$ является 3-й наибольший элемент строки матрицы графа, т.к. $k = h \cdot |S_-| + 1$ $k = 2 \cdot 1 + 1 = 3$. Все нормализаторы в матрице графа выделены зеленым, а тяжелые ребра –

красным. Поскольку строка 1 нейтральна, то вес тяжелых ребер по теореме 3 равен нормализатору, т.к.

$$\beta_{cj} = \beta_{sj} + \max(L^{cs}, t + 1) = 0 + L_{norm}^{cs}.$$

После нормализации ребер получим следующую матрицу.

I/J	0	1	2	3	4	5	6	7	8	9	V
1	0	0	0	0	0	0	0	0	0	0	0
2	10	-19	-41	10	7	10	1	10	6	-27	0
3	1	-15	-10	-7	1	-13	-9	1	-8	-4	0
4	-15	-21	-17	-13	4	-5	4	4	2	-8	0
5	-19	-18	-24	6	6	6	-4	-6	6	-2	0
U	-29	-38	-48	-39	-23	-21	-21	-27	-17	-41	

Рис. 10. Вид β после нормализации

I/J	0	1	2	3	4	5	6	7	8	9	V	Norm
1	-10	19	41	-10	-7	-10	-1	-10	-6	27	0	19
2	0	0	0	0	0	0	0	0	0	0	0	0
3	-9	4	31	-17	-6	-23	-10	-9	-14	23	0	4
4	-25	-2	24	-23	-3	-15	3	-6	-4	19	0	3
5	-29	1	17	-4	-1	-4	-5	-16	0	25	0	1
U	-39	-19	-7	-49	-30	-31	-22	-37	-23	-14		

Рис. 11. Вычитание строки 2 из полученной матрицы

I/J	1	2	3	4	5	6	7	8	9	10	V
1	-10	19	19	-10	-7	-10	-1	-10	-6	19	0
2	0	0	0	0	0	0	0	0	0	0	0
3	-9	4	4	-17	-6	-23	-10	-9	-14	4	0
4	-25	-2	3	-23	-3	-15	3	-6	-4	3	0
5	-29	1	1	-4	-1	-4	-5	-16	0	1	0
U	-39	-19	-7	-49	-30	-31	-22	-37	-23	-14	

Рис. 12. Нормализация тяжелых ребер

I/J	1	2	3	4	5	6	7	8	9	10	V	Norm
1	-1	15	15	7	-1	13	9	-1	8	15	0	15
2	9	-4	-4	17	6	23	10	9	14	-4	0	14
3	0	0	0	0	0	0	0	0	0	0	0	0
4	-16	-6	-1	-6	3	8	13	3	10	-1	0	8
5	-20	-3	-3	13	5	19	5	-7	14	-3	0	13
U	-30	-23	-11	-32	-24	-8	-12	-28	-9	-18		

Рис. 13. Вычитание строки 3 из полученной матрицы

Обратите внимание – в первой строке три равных максимальных нормализатора. Следовательно, нет ребер, веса которых можно уменьшить по теореме 3. Чтобы не выполнять подобных бесполезных операций нужно

рационально выбирать пары линий МЛП. Если же перебирать все подряд пары линий в двух циклах for, то сложность алгоритма составит $O(n^4)$.

I/J	1	2	3	4	5	6	7	8	9	10	V
1	-1	15	15	7	-1	13	9	-1	8	15	0
2	9	-4	-4	14	6	14	10	9	14	-4	0
3	0	0	0	0	0	0	0	0	0	0	0
4	-16	-6	-1	-6	3	8	8	3	8	-1	0
5	-20	-3	-3	13	5	13	5	-7	13	-3	0
U	-30	-23	-11	-32	-24	-8	-12	-28	-9	-18	

Рис.14. Нормализация тяжелых ребер

I/J	1	2	3	4	5	6	7	8	9	10	V	Norm
1	15	21	16	13	-4	5	1	-4	0	16	0	16
2	25	2	-3	20	3	6	2	6	6	-3	0	6
3	16	6	1	6	-3	-8	-8	-3	-8	1	0	6
4	0	0	0	0	0	0	0	0	0	0	0	0
5	-4	3	-2	19	2	5	-3	-10	5	-2	0	5
U	-14	-17	-10	-26	-27	-16	-20	-31	-17	-17		

Рис.15. Вычитание строки 4 из полученной матрицы

I/J	1	2	3	4	5	6	7	8	9	10	V
1	15	16	16	13	-4	5	1	-4	0	16	0
2	6	2	-3	6	3	6	2	6	6	-3	0
3	6	6	1	6	-3	-8	-8	-3	-8	1	0
4	0	0	0	0	0	0	0	0	0	0	0
5	-4	3	-2	5	2	5	-3	-10	5	-2	0
U	-14	-17	-10	-26	-27	-16	-20	-31	-17	-17	

Рис.16. Нормализация тяжелых ребер

I/J	1	2	3	4	5	6	7	8	9	10	V	Norm
1	19	13	18	8	-6	0	4	6	-5	18	0	18
2	10	-1	-1	1	1	1	5	16	1	-1	0	5
3	10	3	3	1	-5	-13	-5	7	-13	3	0	3
4	4	-3	2	-5	-2	-5	3	10	-5	2	0	3
5	0	0	0	0	0	0	0	0	0	0	0	0
U	-10	-20	-8	-31	-29	-21	-17	-21	-22	-15		

Рис.17. Вычитание строки 5 из полученной матрицы

I/J	1	2	3	4	5	6	7	8	9	10	V
1	18	13	18	8	-6	0	4	6	-5	18	0
2	5	-1	-1	1	1	1	5	5	1	-1	0
3	3	3	3	1	-5	-13	-5	3	-13	3	0
4	3	-3	2	-5	-2	-5	3	3	-5	2	0
5	0	0	0	0	0	0	0	0	0	0	0
U	-10	-20	-8	-31	-29	-21	-17	-21	-22	-15	

Рис.18. Нормализация тяжелых ребер

После нормализации строк мы получили два равных столбца (3,10).

Выполним их нейтрализацию.

I/J	1	2	3	4	5	6	7	8	9	10	V
1	0	-5	0	-10	-24	-18	-14	-12	-23	0	-18
2	6	0	0	2	2	2	6	6	2	0	1
3	0	0	0	-2	-8	-16	-8	0	-16	0	-3
4	1	-5	0	-7	-4	-7	1	1	-7	0	-2
5	0	0	0	0	0	0	0	0	0	0	0
U	-10	-20	-8	-31	-29	-21	-17	-21	-22	-15	

Рис. 19. Вид матрицы после нормализации

Перейдем к нормализации ребер по столбцам. Определим порядок нормализатора 3-го

столбца $k = \left\lfloor \frac{|S_{-}|}{h} \right\rfloor + 1, k = \left\lfloor \frac{2}{2} \right\rfloor + 1 = 2$.

Нормализация по другим серверным столб-

цам бесполезна, т.к. для них число тяжелых

ребер $t = \left\lfloor \frac{1}{2} \right\rfloor = 0$.

Norm	1	0	0	0	0	0	1	1	0	0	
I/J	1	2	3	4	5	6	7	8	9	10	V
1	0	-5	0	-10	-24	-18	-14	-12	-23	0	-18
2	1	0	0	0	0	0	1	1	0	0	1
3	0	0	0	-2	-8	-16	-8	0	-16	0	-3
4	1	-5	0	-7	-4	-7	1	1	-7	0	-2
5	0	0	0	0	0	0	0	0	0	0	0
U	-10	-20	-8	-31	-29	-21	-17	-21	-22	-15	

Рис. 20. Нормализация ребер по столбцам

В начальной строке матрицы Norm приведены нормализаторы мультимножеств 3-го столбца. Мы видим, что нормализация по столбцам пока невозможна, т.к. в каждом столбце есть по два или три равных максимума. Значит нужно продолжить нормализацию по строкам.

Продолжаем вычитать строки и уменьшать веса недопустимых ребер. В результате получим матрицу, в которой есть пять эквивалентных столбцов (5,6,7,8,9) и две эквивалентных строки (1,3).

Norm	-5	-2	-5	-2	0	0	0	0	0	-5	
I/J	1	2	3	4	5	6	7	8	9	10	V
1	0	0	0	0	0	0	0	0	0	0	-18
2	-5	-3	-6	-3	0	0	0	0	0	-6	-12
3	0	0	0	0	0	0	0	0	0	0	-10
4	-5	-5	-5	-5	0	0	0	0	0	-5	-14
5	-5	-2	-5	-2	0	0	0	0	0	-5	-12
U	-3	-8	-1	-17	-5	2	-2	-9	1	-8	

Рис. 21. Дальнейшая нормализация по строкам

I/J	1	2	3	4	5	6	7	8	9	10	V
1	-5	-2	-5	-2	0	0	0	0	0	-5	-18
2	-5	-3	-6	-3	0	0	0	0	0	-6	-12
3	-5	-2	-5	-2	0	0	0	0	0	-5	-10
4	-5	-5	-5	-5	0	0	0	0	0	-5	-14
5	-5	-2	-5	-2	0	0	0	0	0	-5	-12
U	-3	-8	-1	-17	-5	2	-2	-9	1	-8	

Рис. 22. Нормализация тяжелых ребер в МЛП пятого серверного столбца.

После нормализации элементов клиентских строк (1,3) мы получили три равных строки (1,3,5).

I/J	1	2	3	4	5	6	7	8	9	10	V	Norm
1	0	0	0	0	0	0	0	0	0	0	-18	0
2	0	-1	-1	-1	0	0	0	0	0	-1	-12	-1
3	0	0	0	0	0	0	0	0	0	0	-10	0
4	0	-3	0	-3	0	0	0	0	0	0	-14	0
5	0	0	0	0	0	0	0	0	0	0	-12	0
U	2	-6	4	-15	-5	2	-2	-9	1	-3		

Рис. 23. Вычитание строки 1 из полученной матрицы

I/J	1	2	3	4	5	6	7	8	9	10	V
1	0	0	0	0	0	0	0	0	0	0	-18
2	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-12
3	0	0	0	0	0	0	0	0	0	0	-10
4	0	-3	0	-3	0	0	0	0	0	0	-14
5	0	0	0	0	0	0	0	0	0	0	-12
U	2	-3	4	-12	-5	2	-2	-9	1	-3	

Рис. 23. Нормализация тяжелых ребер

I/J	1	2	3	4	5	6	7	8	9	10	V	Norm
1	0	0	0	0	0	0	0	0	0	0	-18	0
2	0	0	0	0	0	0	0	0	0	0	-11	0
3	0	0	0	0	0	0	0	0	0	0	-10	0
4	0	-3	0	-3	0	0	0	0	0	0	-14	-3
5	0	0	0	0	0	0	0	0	0	0	-12	0
U	2	-3	4	-12	-5	2	-2	-9	1	-3		

Рис. 24. Вычитание строки 1 и столбца 1 из полученной матрицы.

Обратите внимание, что нормализатор -3 является 9-м по весу элементом строки 4, т.к. $k=t+1=4*2+1$.

I/J	1	2	3	4	5	6	7	8	9	10	V
1	0	0	0	0	0	0	0	0	0	0	-18
2	0	0	0	0	0	0	0	0	0	0	-11
3	0	0	0	0	0	0	0	0	0	0	-10
4	-3	-3	-3	-3	-3	-3	-3	-3	-3	-3	-14
5	0	0	0	0	0	0	0	0	0	0	-12
U	2	-6	4	-15	-5	2	-2	-9	1	-3	

Рис. 25. Нормализатор -3

Вычитание столбца 1, получение линейных коэффициентов эквивалентной матрицы β^0 .

I/J	1	2	3	4	5	6	7	8	9	10	V
1	0	0	0	0	0	0	0	0	0	0	-18
2	0	0	0	0	0	0	0	0	0	0	-11
3	0	0	0	0	0	0	0	0	0	0	-10
4	0	0	0	0	0	0	0	0	0	0	-11
5	0	0	0	0	0	0	0	0	0	0	-12
U	2	-6	4	-15	-5	2	-2	-9	1	-3	

Рис. 26. Нейтральная матрица

Получена нейтральная матрица – конец алгоритма. Вес оптимального НЗП:

$$w(E_h^*) = - \left[\sum_{i \in I} U_i + h \cdot \sum_{j \in J} V_j \right] = 155.$$

Следует отметить, что существует бесконечное множество допустимых комбинаций

коэффициентов U и V. Конкретные значения коэффициентов зависят от порядка перебора пар строк и столбцов при нормализации.

Теперь выполним восстановление весов ребер $B_{ij}^- = -(U_i + V_j)$ по линейным коэффициентам.

I/J	1	2	3	4	5	6	7	8	9	10	V
1	16	24	14	33	23	16	20	27	17	21	0
2	9	17	7	26	16	9	13	20	10	14	0
3	8	16	6	25	15	8	12	19	9	13	0
4	9	17	7	26	16	9	13	20	10	14	0
5	10	18	8	27	17	10	14	21	11	15	0
U	0	0	0	0	0	0	0	0	0	0	

Рис. 27 Матрица с восстановленными весами ребер

Зеленым выделены элементы оптимального НЗП $w(E_h^*) = 155$.

Вычислим матрицу D разности весов начальной матрицы графа и β^- , вычисленной алгоритмом (Рис. 28).

Эта матрица показывает, на какую величину нормализационный алгоритм уменьшил вес тяжелых ребер графа. Нули матрицы D определяют ребра оптимального НЗП. Необходимо отметить, что в этой матрице могут быть лишние нули, ребра которых не принадлежат оптимальному НЗП. Эти лишние

ребра называются *анкерными*. Лишние ребра могут быть также элементами другого эквивалентного оптимального решения. Чтобы найти одно решение нужно знать, как удалять лишние ребра. В данном случае это сделать просто. Для этого нужно выделить строки, в которых есть ровно по два h зеленых элемента (2,4,5) или столбцы в которых присутствует только один элемент (1-7,10) и затем исключить лишние ребра. В общем случае, задача удаления лишних ребер не является простой. Эта задача эквивалентна нахождению подмножеств заданного множества, она решается через поиск паросочетания.

I/J	1	2	3	4	5	6	7	8	9	10
1	13	14	34	6	0	5	1	0	0	20
2	37	2	0	23	14	25	9	17	13	0
3	32	7	32	7	9	0	0	18	0	24
4	5	0	24	0	11	7	15	16	9	19
5	0	2	16	18	15	17	3	0	26	24

Рис. 28 Матрица D

Лишние ребра также можно удалить нормализационным алгоритмом. Для этого можно внести малые возмущения, т.е. изменить вес разрешенных ребер на разную величину и удалить лишние в ходе нормализации. Можно также последовательно перебирать ребра и проверять существование НЗП.

Следует также отметить, что ребра эквивалентных оптимальных решений формируют циклы. Поэтому множество всех оптимальных решений можно представить в виде системы вложенных циклов.

Заключение

В данной публикации впервые приведены доказательства ключевых теорем нормализационного метода. Теперь, мы имеем все основания считать задачу поиска наибольшего звездного покрытия графа эффективно решаемой.

Во второй части статьи подробно рассмотрен пример решения задачи поиска НЗП. Поэтому, настоящую работу можно считать практическим пособием по методу решения задачи. Следует отметить, что для упрощения понимания нормализационного метода некоторые на первый взгляд очевидные моменты его реализации не рассматривались, а именно – структуры данных и алгоритмы формирования множеств эквивалентных линий. При разработке программы на эти моменты следует обратить внимание.

В прикладном аспекте результаты работы могут быть использованы при решении задач

транспортной логистики, оптимизации параллельных производственных процессов, а также при решении прикладных задач распределения работ в системах, представленных независимыми машинами различной производительности.

Литература

1. Ловас Л. и Пламмер М. Прикладные задачи теории графов. Теория паросочетаний в математике, физике, химии. М.: Мир, 1998. – 653 с.
2. Подоляка А.Н. Эффективное решение задачи покрытия двудольного графа звездами и некоторых ее обобщений / А.Н. Подоляка, О.А. Подоляка, Е.В. Скакалина // Вісник Чернігівського державного технологічного ун-ту, 2012. – №4(61). – С. 172 – 179.
3. Подоляка О.А. Поиск наибольшего покрытия двудольного графа звездами заданной степени / О.А. Подоляка, А.Н. Подоляка // Автомобіль і електроніка. Сучасні технології, 2015. – Вип. 7. – С. 126 – 132.
4. Ніконов О.Я. Математичні методи розв'язання багатокритеріальної задачі про призначення / О.Я. Ніконов, О.О. Подоляка, О.М. Подоляка, О.В. Скакаліна // Вісник ХНАДУ, 2011. – №55. – С. 23-33.

Рецензент: О.Я. Никонов, профессор, д.т.н., ХНАДУ.