Предмет «Экономико-математические методы»

группы ЭЗБ-190-1с,2с

План:

Общие сведения о графах:

1. Определение графа
2. Матрица смежности и матрица инцидентности
3. Цикломатическое число
4. Хроматическое число
5. Метрические характеристики графа

Экономические приложения:
1. Алгорим Декстры;
2. Траспортные задачи

О предмете:

В последнее время наблюдается неуклонное вторжение математических методов в различные отрасли науки и техники. Процесс математизации затронул и экономическую науку, в которой возникло экономико-математическое направление. Это направление появилось сравнительно недавно, но быстро и успешно развивается.

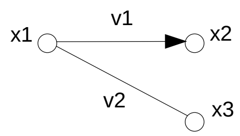
Теория графов изучает графы как абстрактные математические образования, независимо от их конкретных истолкований, а полученные общие результаты затем прилагаются к самым различным дисциплинам.

Как математическая дисциплина теория графов сформировалась в середине XX в., хотя отдельные задачи о графах имеют 200-летнюю давность. Термин «граф» приобрел право гражданства и вошел в математический язык в 1936 г., после выхода в свет монографии Кёнига, в которой впервые графы изучаются как самостоятельные математические объекты независимо от их содержания. За последние 25 лет теория графов развивается весьма быстро. В связи с интенсивным развитием экономико-математических исследований возникает много задач и методов связанных с графами.

Общие сведения о графах

Граф – совокупность объектов (X, V), где X – множество вершин графа, а V – множество связей графа, символически обозначаемый G (X,У).

Если связь между вершинами не направлена, то ее называют *ребром***.** Если связь между вершинами направлена, то ее называют *дугой.*



где связь v1 является дугой, а v2 - ребром.

Если в графе имеется ребро (дуга) (x_i, x_j) (причем для дуги x_i - вершина, из которой она исходит, x_j - в которую заходит), то говорят, что вершины x_i и x_j *смежные***,** и *не смежные* в противном случае.

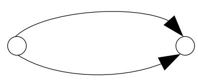
Вершины, определяющие связь, называются *концевыми* вершинами этой связи.

Два ребра (дуги) называются смежными**,** если они имеют общую концевую вершину.

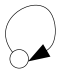
Ребро (дуга) и ее концевые вершины являются *инцидентными* друг другу.

Смежность есть отношение между однородными элементами графа, тогда как инцидентность является отношением между разнородными элементами.

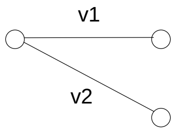
Все ребра (дуги) с одинаковыми концевыми вершинами (дуги должны иметь одинаковое направление) называются *параллельными или кратными*.



Ребро (дуга) с совпадающими концевыми вершинами называется *петлей.*



Граф называется *простым*, если он не содержит петель и кратных *ребер* (дуг).



Простой граф называется *полным*, если в нем каждая пара вершин является смежной.

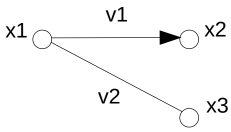
Граф, содержащий кратные связи, называет *мультаграфом*.

Если в графе допускаются и петли, и кратные связи, то граф называется *псевдографом*.

Граф, не имеющий связей, называется *пустым*.

Если в графе все связи направлены (дуги), то такой граф называется *ориентированным* (орграфом), в противном случае *неориентированным*.

Если в графе присутствуют ребра и дуги, то такой граф является *смешанным*.



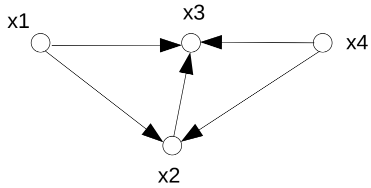
Степенью вершины $P(x_i)$ называется число ребер графа G инцидентных вершине x_i.

Вершина графа, имеющая степень 0 называется *изолированной*, а степень 1 *висячей*.

Для орграфа вводятся характеристики *полустепень исхода* $P^{-}(x_i)$ - количество дуг исходящих из вершины x_i и *полустепень захода* $P^{+}(x_i)$ - количество дуг заходящих в вершину x_i.

Для ориентированного графа справедливо равенство $P(x_i) = P^{+}(x_i) + P^{-}(x_i)$.

Пр.



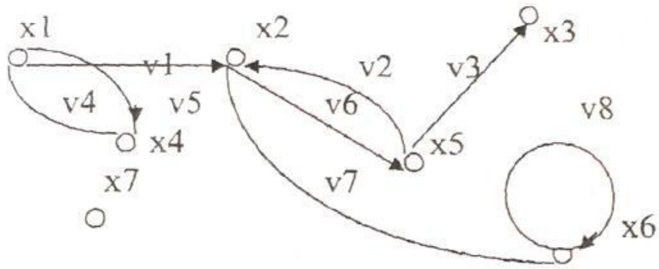
$$P(x_i) = P^+(x_i) + P^-(x_i)$$
$$P(x_1) = 0 + 2 = 2$$
$$P(x_2) = 2 + 1 = 3$$
$$P(x_3) = 3 + 0 = 3$$
$$P(x_4) = 0 + 2 = 2$$

Способы задания графов

Существуют четыре способа задания графов: графический, перечислительный, с помощью матриц смежности и инцидентности.

1. *Графический способ*

Граф задается множеством вершин и связей:



(1)

2. *Перечислительный способ*

Граф задается перечислением множества вершин, множества дуг и ребер, списком изолированных вершин.

Граф, изображенный на рис., можно задать следующим описанием:

множество вершин: $X = \{x_1, x_2, x_3, x_3, x_5, x_6, x_7\}$;

множество дуг: $V = \{(x_1, x_2), (x_5, x_2), (x_5, x_3), (x_1, x_4), (x_2, x_5), (x_6, x_6)\}$;

множество ребер: $V = \{(x_1, x_4), (x_2, x_6)\}$;

список изолированных вершин: $\{X7\}$.

3. *Матрица смежности*

Матрица смежности является квадратной матрицей размером n x n, где n – количество вершин заданного графа $A(G) = \|a_{ij}\|$.

Для неориентированного графа a_{ij}, принимает следующие значения:

$$a_{ij} = \begin{cases} k, \text{где } k - кол-во \text{ связей между } \mathrm{x_i} \text{ и } \mathrm{x_j} \text{ вершинами} \\ 0, \text{е}сли \text{ связи между } \mathrm{x_i} \text{ и } \mathrm{x_j} \text{ отсутствуют} \end{cases}$$

Для ориентированного графа a_{ij}, принимает следующие значения:

$$a_{ij} = \begin{cases} k, \text{где } k - кол-во \text{ дуг исходящих из } \mathrm{x_i} \text{ и заходящих в } \mathrm{x_j} \\ 0, \text{е}сли \text{ связи между } \mathrm{x_i} \text{ и } \mathrm{x_j} \text{ отсутствуют} \end{cases}$$

По матрице смежности можно определить:

1. полустепень исхода вершины $\mathrm{x_i}$ как сумму чисел в i-строке;
2. полустепень захода вершины $\mathrm{x_i}$ как сумму чисел в i-столбце;
3. петли как элемнты главной диагонали матрицы;
4. изолированные вершины при условии, что соответствующие строки и столбцы содержат только нулевые элементы.

Пример:

Построим матрицу смежности для графа (1):

x_i / x_j	x_1	x_2	x_3	x_4	x_5	x_6	x_7
x_1	0	1	0	2	0	0	0
x_2	0	0	0	0	1	1	0
x_3	0	0	0	0	0	0	0
x_4	1	0	0	0	0	0	0
x_5	0	1	1	0	0	0	0
x_6	0	1	0	0	0	1	0
x_7	0	0	0	0	0	0	0

4. *Матрица инцидентности*

Матрица инцидентности – это матрица размером nxm, где n – количество вершин графа, а m - количество связей графа $B(G) = \|b_{ij}\|$.

Для неориентированного графа b_{ij}, принимает следующие значения:

$$b_{ij}=\begin{cases}1,\ \text{если вершина } x_i \text{ инцидентна ребру } v_j\\ 0,\ \text{если вершина } x_i \text{ не инцидентна ребру } v_j\end{cases}$$

Для ориентированного графа b_{ij}, принимает следующие значения:

$$b_{ij}=\begin{cases}1,\ \text{если вершина } x_i \text{ начало дуги } v_j\\ -1,\ \text{если вершина } x_i \text{ конец дуги } v_j\\ 0,\ \text{если вершина } x_i \text{ не инцидентна дуге } v_j\end{cases}$$

По матрице инцидентности можно определить:

1. если в столбце 1-а единица, то столбец петля;
2. если единицы имеют противоположные знаки, то граф ориентирован.

Пример:

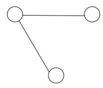
Построим матрицу инцидентности для графа (1):

x_i / v_j	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
x_1	1	0	0	1	1	0	0	0
x_2	-1	-1	0	0	0	1	1	0
x_3	0	0	-1	0	0	0	0	0
x_4	0	0	0	1	-1	0	0	0
x_5	0	1	1	0	0	-1	0	0
x_6	0	0	0	0	0	0	1	1
x_7	0	0	0	0	0	0	0	0

Связность. Компоненты связности

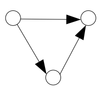
Вершина x_1 графа G достижима из вершины x_2, если $x_1 = x_2$, либо существует маршрут соединяющий вершины x_1, x_2.

Граф (орграф) называется связным (сильно связным), если для любых его вершин x_i, x_j существует маршрут, соединяющий x_i и x_j.



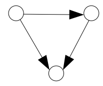
k=1

Граф (орграф) называется односторонне связным, если для любых двух его вершин хотя бы одна вершина достижима из другой.



k=1

Орграф называется слобосвязным, если полученный из него не ориентированный граф, путем замены дуг на ребра является связным.



k=1

Максимальный по включению вершин связный (сильно связный) подграф графа называется его компонентой связности (k).

Общее число компонент связности графа называется степенью связности графа.

Матрица связности графа G называется квадратная матрица $S(G)=\left\|s_{ij}\right\|$

$$s_{ij}=\begin{cases}1, \text{если } i=j \text{ или существует маршрут соединяющий две вершины}\\ 0, \text{иначе}\end{cases}$$

Матрица сильной связности графа G называется квадратная матрица $S(G)=\left\|s_{ij}\right\|$

$$s_{ij}=\begin{cases}1, \text{если одна вершина достижима из другой и обратно}\\ 0, \text{иначе}\end{cases}$$

Алгоритм определения компоненты связности:

1.Составим две матрицы:

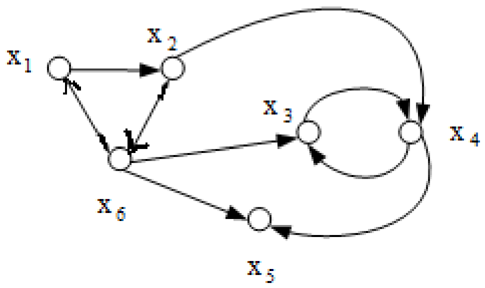
- Матрицу смежности;
- Матрицу связности.

2. В матрице $S(G)$ в 1-ой строке находим единицы и заносим в 1-ую компоненту связности.

   Для того, что бы определить как будут связаны полученные вершины нужно из матрицы $A(G)$ выбрать те строки и столбцы, которые соответствуют взятым вершинам.

3. Вычеркиваем из обеих матриц $S(G)$ и $A(G)$ строки и столбцы, которые соответствуют найденным вершинам в п.2.
4. Повторяем п.2 и п.3 до тех пор пока не вычеркнутая все строки и столбцы.

Пример:



1. Построим матрицу смежности для графа:

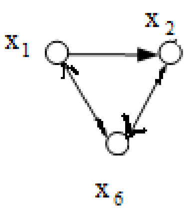
$A(G)$	x_1	x_2	x_3	x_4	x_5	x_6
x_1	0	1	0	0	0	0
x_2	0	0	0	1	0	1
x_3	0	0	0	1	0	0
x_4	0	0	1	0	1	0
x_5	0	0	0	0	0	0
x_6	1	0	1	0	1	0

2. Построим матрицу связности для графа:

$S(G)$	x_1	x_2	x_3	x_4	x_5	x_6
x_1	1	1	0	0	0	1
x_2	1	1	0	0	0	1
x_3	0	0	1	1	0	0
x_4	0	0	1	1	0	0
x_5	0	0	0	0	1	0
x_6	1	1	0	0	0	1

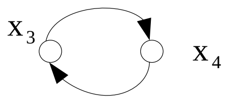
3. В матрице $S(G)$ в 1-ой строке находим единицы и заносим в 1-ую компоненту связности: $k_1=\{x_1, x_2, x_6\}$

Для того, что бы определить как будут связаны полученные вершины нужно из матрицы $A(G)$ выбрать те строки и столбцы, которые соответствуют взятым вершинам.



$A(G)$	x_1	x_2	x_6
x_1	0	1	0
x_2	0	0	1
x_6	1	0	0

4. В матрице $S(G)$ в 3-ей строке находим единицы и заносим во 2-ую компоненту связности: $k_2=\{x_3, x_4\}$.



$A(G)$	x_3	x_4
x_3	0	1
x_4	1	0

5. В матрице $S(G)$ в 5-ой строке находим единицы и заносим в 3-ую компоненту связности: $k_3=\{x_5\}$.

Ответ: K=3

Цикломатическое число вычисляется по формуле: $\nu(G)=m-n+k$, где m – число связей; n - число вершин; k – коэффициент связности.

Метрические характеристики графа

Длина кратчайшего пути маршрута называется расстоянием между вершинами $d(x_i, x_j)$.

Диаметром графа G называется max $(d(x_i, x_j))$.

Пусть x_i - произвольная вершина из X. Величина $r(x)=\max d(x_i, x_j)$ называется максимальным удалением (эксцентриситет).

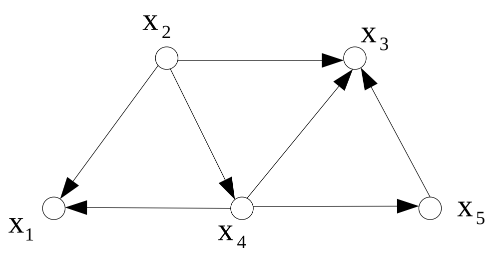
Радиусом графа G называется величина $r(G)=\min (r(x_i))$.

Любая вершина $x_i \in X$ такая, что $r(x_i)= r(G)$ называется центром графа G (центральные вершины).

Задача нахождения центральных вершин графа постоянно используется в практической деятельности людей.

Пусть, например, граф представляет сеть дорог, т.е. вершины его соответствуют отдельным населенным пунктам, а ребра – дорогам между ними. Требуется оптимально разместить магазины, пункты обслуживания, АЗС. В подобных ситуациях критерий оптимальности часто заключается в оптимальности «наихудшего» случая, т.е. в минимальном расстоянии от места обслуживания до наиболее удаленного пункта. Следовательно, местами размещения должны быть центральные вершины.

Пример:



Решение:

1. Расстояние между вершинами:

$d(x_1, x_2)=1$

$d(x_1, x_3)=2$ $d(x_1, x_3)=2$

$d(x_1, x_4)=1$ $d(x_2, x_4)=1$ $d(x_3, x_4)=1$

$d(x_1, x_5)=2$ $d(x_2, x_5)=2$ $d(x_3, x_5)=1$ $d(x_4, x_5)=1$

2. Диаметром графа: $d(G)=\max(d(x_i, x_j))=2$
3. Максимальным удалением (эксцентриситет):

$r(x_1)=2$

$r(x_2)=2$

$r(x_3)=2$

$r(x_4)=1$

$r(x_5)=2$

4. Радиусом графа: $r(G)=\min(r(x_i))=1$
5. Центральная вершина: x_4

Экономические приложения

Предположим, что необходимо посетить каждый из нескольких городов (магазинов, складов и т.д.) и возвратиться к своему первоначальному положению. Учитывая сеть дорог, соединяющих различные города (магазины и т.д.) на маршруте, проблема состоит в том, чтобы найти маршрут, который минимизирует полное расстояние путешествия.

Сеть дорог может быть представлена взвешенным графом (т.е. графом каждому ребру, которого присвоено определенное число – вес ребра) следующим образом. Каждый интересующий нас объект представляется вершиной, а каждая дорога или маршрут – ребром, соединяющим соответствующие две вершины, причем вес ребра равен длине данной дороги. Путешествие, при котором посещается каждый объект и которое заканчивается в исходном (стартовом) положении, представляется замкнутой последовательностью ребер графа, у которого последовательность вершин содержит все вершины. В терминологии теории графов цикл графов, проходящий через все вершины графа ровно по одному разу называется гамильтоновым, при этом необязательно проходить через все ребра графа. Вес гамильтонова цикла – сумма весов ребер, содержащихся в нем.

Как правило, оказывается целесообразным несколько видоизменить исходную проблему следующим образом. Каждому ребру полного графа (*граф, в котором все вершины попарно соединены между собой*) приписывается вес, равный наикратчайшему расстоянию между соответствующими объектами, в соответствии с используемой сетью дорог. Эти самые короткие расстояния могут быть найдены с помощью алгоритмов Декстра, Форда-Беллмана.

Алгоритм Дейкстры

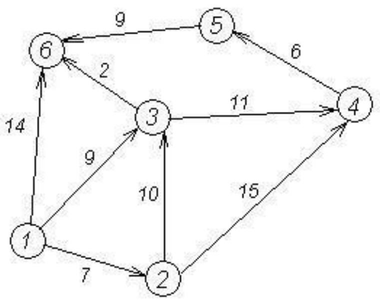
Метка самой вершины a полагается равной 0, метки остальных вершин — бесконечности. Это отражает то, что расстояния от a до других вершин пока неизвестны.

Все вершины графа помечаются как «непосещенные».

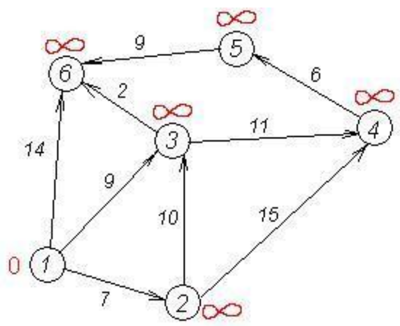
Если все вершины посещены, алгоритм завершается. В противном случае из еще не посещенных вершин выбирается вершина *u*, имеющая минимальную метку. Мы рассматриваем всевозможные маршруты, в которых *u* является предпоследним пунктом. Вершины, соединенные с вершиной *u* ребрами, назовем соседями этой вершины. Для каждого соседа рассмотрим новую длину пути, равную сумме текущей метки *u* и длины ребра, соединяющего *u* с этим соседом. Если полученная длина меньше метки соседа, заменим метку этой длиной. Рассмотрев всех соседей, пометим вершину *u* как посещенную и повторим шаг.

<u>Пример</u>

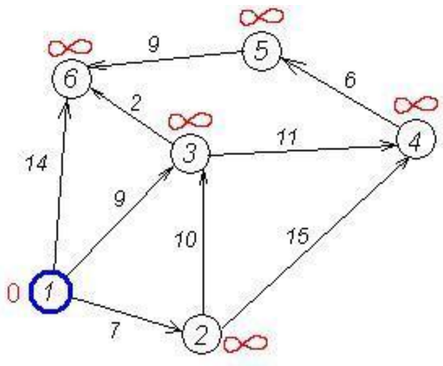
Рассмотрим работу алгоритма на примере графа, показанного на рисунке. Пусть требуется найти расстояния от 1-й вершины до всех остальных.



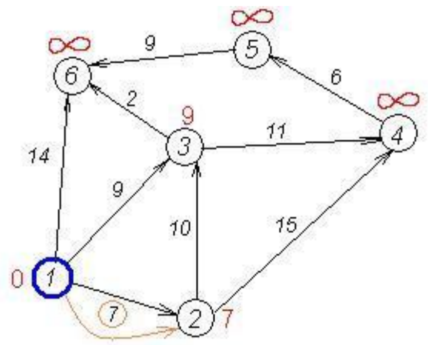
Кружками обозначены вершины, линиями — пути между ними (ребра графа). В кружках обозначены номера вершин, над ребрами обозначена их «цена» — длина пути. Рядом с каждой вершиной красным обозначена метка — длина кратчайшего пути в эту вершину из вершины 1.



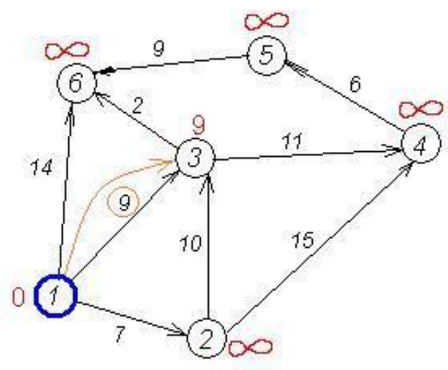
<u>Первый шаг:</u> Рассмотрим шаг алгоритма Дейкстры для нашего примера. Минимальную метку имеет вершина 1. Ее соседями являются вершины 2, 3 и 6.

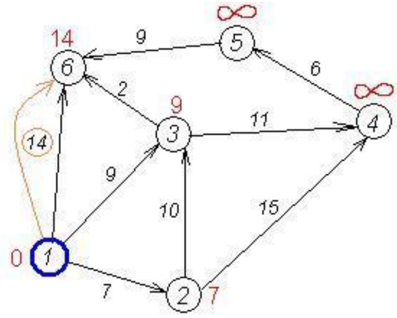


Первый по очереди сосед «вершины 1» — «вершина 2», потому что длина пути до нее минимальна. Длина пути в нее через «вершину 1» равна кратчайшему расстоянию до «вершины 1» + длина ребра, идущего из 1 в 2, то есть 0 + 7 = 7. Это меньше текущей метки вершины 2, поэтому новая метка 2-й вершины равна 7.

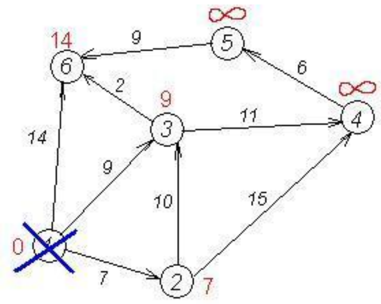


Аналогичную операцию проделываем с двумя другими соседями 1-й вершины — 3-й и 6-й.

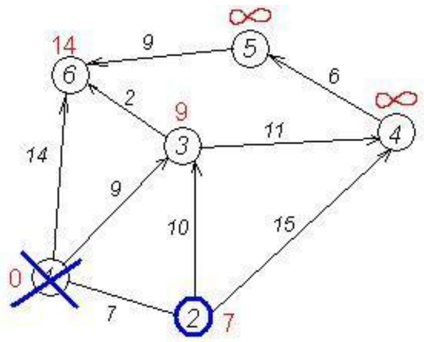




Все соседи вершины 1 проверены. Текущее минимальное расстояние до вершины 1 считается окончательным и пересмотру не подлежит. Вычеркнем её из графа, чтобы отметить, что эта вершина посещена.



<u>Второй шаг:</u> Шаг алгоритма повторяется. Снова находим минимальную метку из непосещенных вершин. Это вершина 2 с меткой 7.

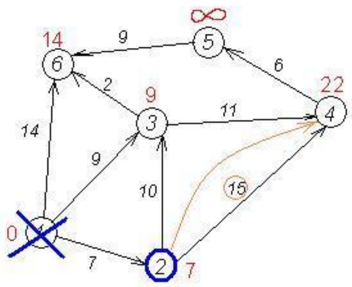


Снова пытаемся уменьшить метки соседей выбранной вершины, пытаясь пройти в них через 2-ю. Соседями вершины 2 являются 1, 3, 4.

Первый (по порядку) сосед вершины 2 — вершина 1. Но она уже посещена, поэтому с 1-й вершиной ничего не делаем.

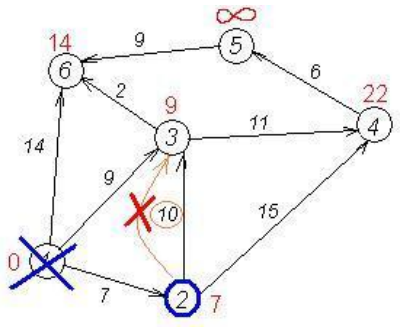
Следующий сосед вершины 2 — вершина 4.

Длина пути будет = кратчайшее расстояние до 2 + расстояние между вершинами 2 и 4 = 7 + 15 = 22. Т.к. 22<∞, устанавливаем метку вершины 4 равной 22.



Ещё один сосед вершины 2 — вершина 3.

Длина такого пути будет = 7 + 10 = 17. Но текущая метка третьей вершины равна 9<17, поэтому метка не меняется.



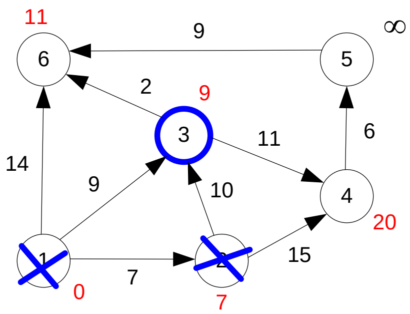
Все соседи вершины 2 просмотрены, замораживаем расстояние до неё и помечаем ее как посещенную.

<u>Третий шаг:</u> Снова находим минимальную метку из непосещенных вершин (вершина 3).

Соседями вершины 3 являются 6 и 4.

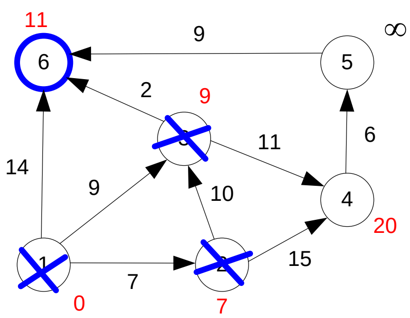
вершина 4: длина пути будет = 9 + 11 = 20. Текущая метка вершины 4 равна 22, т.к. 20<22 меняем метку на 20.

вершина 6: длина пути будет = 9 + 2 = 11. Текущая метка вершины 6 равна 14, т.к. 11<14 меняем метку на 11.



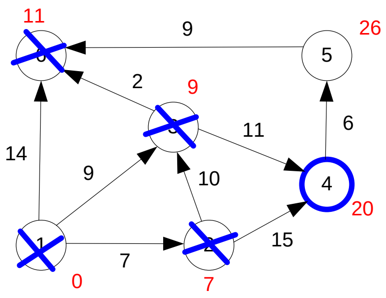
Все соседи вершины 3 просмотрены, замораживаем расстояние до неё и помечаем ее как посещенную.

<u>Четвертый шаг:</u> Снова находим минимальную метку из непосещенных вершин (вершина 6). У данной вершины соседей нет, соответственно помечаем ее как посещенную.



Пятый шаг: Снова находим минимальную метку из непосещенных вершин (вершина 4). Соседом вершины 4 являются 5.

вершина 5: длина пути будет = 20 + 6 = 26. Т.к. 26<∞, устанавливаем метку вершины 5 равной 26.



Все соседи вершины 4 просмотрены, замораживаем расстояние до неё и помечаем ее как посещенную.

Завершение выполнения алгоритма: Алгоритм заканчивает работу, когда вычеркнуты все вершины. Результат его работы: кратчайший путь от вершины 1 до 2-й составляет 7, до 3-й — 9, до 4-й — 20, до 5-й — 26, до 6-й — 11.

Транспортные задачи

Транспортная задача является одной из наиболее распространенных специальных задач линейного программирования. Частные постановки задачи рассмотрены рядом специалистов по транспорту, например О. Н. Толстым. Первая строгая постановка транспортной задачи принадлежит Ф. Хичкоку, поэтому в зарубежной литературе ее называют проблемой Хичкока. Первый точный метод решения Т-задачи разработан Л. В. Канторовичем и М. К. Гавуриным. Под названием “транспортная задача” объединяется широкий круг задач с единой математической моделью.

Постановка задачи

Под термином "транспортные задачи" понимается широкий круг задач не только транспортного характера. Общим для них является, как правило, распределение ресурсов, находящихся у *m* производителей (поставщиков), по

n потребителям этих ресурсов. В качестве критериев в транспортных задачах используют: *критерий стоимости* (план перевозок оптимален, если достигнут минимум затрат на его реализацию), *критерий времени* (план оптимален, если на его реализацию затрачивается минимум времени) и другие.

Наиболее часто встречаются следующие задачи, относящиеся к транспортным:

- прикрепление потребителей ресурса к производителям;
- привязка пунктов отправления к пунктам назначения;
- взаимная привязка грузопотоков прямого и обратного направлений;
- отдельные задачи оптимальной загрузки промышленного оборудования;
- оптимальное распределение объемов выпуска промышленной продукции между заводами-изготовителями и др.

Рассмотрим экономико-математическую модель прикрепления пунктов отправления к пунктам назначения. Имеются m пунктов отправления груза и объемы отправления по каждому пункту $a_1, a_2, ..., a_m$. Известна потребность в грузах $b_1, b_2, ..., b_n$ по каждому из n пунктов назначения. Задана матрица стоимостей доставки по каждому варианту c_{ij}, $i=\overline{1,m}$ $j=\overline{1,n}$. Необходимо рассчитать оптимальный план перевозок, т.е. определить, сколько груза должно быть отправлено из каждого i-го пункта отправления (от поставщика) в каждый j-ый пункт назначения (до потребителя) x_{ij} с минимальными транспортными издержками.

В общем виде исходные данные представлены в табл. 3.1. Строки транспортной таблицы соответствуют пунктам отправления (в последней клетке каждой строки указан объем запаса продукта a_i), а столбцы - пунктам назначения (последняя клетка каждого столбца содержит значение потребности b_j). Все клетки таблицы (кроме тех, которые расположены в нижней строке и правом столбце) содержат информацию о перевозке из i-го пункта в j-й: в правом верхнем углу находится цена перевозки единицы продукта, а в левом нижнем - значение объема перевозимого груза для данных пунктов.

Таблица 1. Исходные данные

Потребители / Поставщики	B_1	B_2	…	B_n	Запасы (объемы отправления)
A_1	c_{11} x_{11}	c_{12} x_{12}	…	c_{1n} x_{1n}	a_1
A_2	c_{21} x_{21}	c_{22} x_{22}	…	c_{2n} x_{2n}	a_2
…	…	…	…	…	…
A_m	c_{m1} x_{m1}	c_{m2} x_{m2}	…	c_{mn} x_{mn}	a_m
Потребность	b_1	b_2	…	b_n	

Транспортная задача называется закрытой, если суммарный объем отправляемых грузов $\sum_{i=1}^{m} a_i$.равен суммарному объему потребности в этих грузах по пунктам назначения $\sum_{j=1}^{n} b_i$

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_i \qquad 3.1)$$

Если такого равенства нет (потребности выше запасов или наоборот), запасу называют открытой, т.е.:

$$\sum_{i=1}^{m} a_i \neq \sum_{j=1}^{n} b_i \qquad (3.2)$$

Транспортным задачам присущи следующие особенности:

- распределению подлежат однородные ресурсы;
- условия задачи описываются только уравнениями;
- все переменные выражаются в одинаковых единицах измерения;
- во всех уравнениях коэффициенты при неизвестных равны единице;
- каждая неизвестная встречается только в двух уравнениях системы ограничений.

Опорный план является допустимым решением транспортной задачи и используется в качестве начального базисного решения при нахождении оптимального решения методом потенциалов. Существует три метода нахождения опорных планов: метод северо-западного угла, метод минимального элемента и метод Фогеля. "Качество" опорных планов, полученных этими методами, различается: в общем случае метод Фогеля дает наилучшее решение (зачастую оптимальное), а метод северо-западного угла – наихудшее.

Все существующие методы нахождения опорных планов отличаются только способом выбора клетки для заполнения. Само заполнение происходит одинаково независимо от используемого метода.

Методы составления начального опорного плана

Базисный план составляется последовательно, в несколько шагов (точнее, $m + n - 1$ шагов). На каждом из этих шагов заполняется одна клетка, притом так, что, полностью удовлетворяется один из заказчиков (тот, в столбце которого находится заполняемая клетка), либо полностью вывозится весь запас груза с одной из баз (с той, в строке которой находится заполняемая клетка).

Начиная с первоначально данной таблицы и повторив $(m + n - 2)$ раз описанный шаг, мы придем к “таблице”, состоящей из одной строки и одного столбца (иначе говоря, из одной пустой клетки). Другими словами, мы пришли к задаче с одной базой и с одним потребителем, причем потребности этого единственного заказчика равны запасу груза на этой единственной базе. Заполнив последнюю клетку, мы освобождаем последнюю базу и удовлетворяем потребность последнего заказчика. В результате, совершив $(m + n - 1)$ шагов, мы и получим искомый опорный план.

Замечание: Может случиться, что уже на некотором (но не на последнем!) шаге потребность очередного заказчика окажется равной запасу груза на очередной базе. Тогда после заполнения очередной клетки объем таблицы как бы одновременно уменьшается на одни столбец и на одну строку. Но и при этом мы должны считать, что уменьшение объема таблицы происходит либо на один столбец, а на базе сохраняется "остаток" равный нулю, либо на одну строку, а у заказчика еще осталась неудовлетворенная "потребность" в количестве нуля единиц груза, которая и удовлетворяется на одном из следующих шагов. Этот нуль ("запас" или "потребностью" – безразлично) надо записать в очередную заполняемую клетку на одном из последующих шагов. Так как при этом оказывается равной нулю одна из базисных неизвестных, то мы имеем дело с вырожденным случаем.

1. Метод северо-западного угла.

При этом методе на каждом шаге построения первого опорного плана заполняется левая верхняя клетка (северо-западный угол) оставшейся части таблицы. При таком методе заполнение таблицы начинается с клетки неизвестного x_{11} и заканчивается в клетке неизвестного x_{mn} , т. е. идет как бы по диагонали таблицы перевозок.

<u>Пример:</u>

Пункты Отправления	Запасы	Пункты назначения				
		B_1	B_2	B_3	B_4	B_5
		170	110	100	120	200
A_1	300	70 170	50 110	15 20	80	70
A_2	150	80	90	40 80	60 70	85
A_3	250	50	10	90	11 50	25 200

1. Заполнение таблицы начинается с ее северо-западного угла, т.е. клетки с неизвестным x_{11}.
2. Первая база A_1 может полностью удовлетворить потребность первого заказчика B_1 (a_1=300, b_1=170, $a_1 > b_1$). Полагая x_{11}= 170, вписываем это значение в клетку x_{11} и исключаем из

рассмотрения первый столбец. На базе A_1 остается измененный запас $a'_1 = 130$.

3. В оставшейся новой таблице с тремя строками A_1,A_2,A_3 и четырьмя столбцами B_1,B_2,B_3,B_4; северо-западным углом будет клетка для неизвестного x_{12}. Первая база с запасом $a'_1 = 130$ может полностью удовлетворить потребность второго заказчика B_2 $(a'_1 = 130, b_2 = 110, a'_1 > b_2)$. Полагаем x_{12} = 110, вписываем это значение в клетку x_{12} и исключаем из рассмотрения второй столбец. На базе A_1 остается новый остаток (запас) $a''_1 = 20$.
4. В оставшейся новой таблице с тремя строками A_1,A_2,A_3 и тремя столбцами B_3,B_4,B_5 северо-западным углом будет клетка для неизвестного x_{13}. Теперь третий заказчик B_3 может принять весь запас с базы A_1 $(a''_1 = 20, b_3 = 100, a''_1 < b_3)$. Полагаем x_{13} = 20, вписываем это значение в клетку x_{13} и исключаем из рассмотрения первую строку. У заказчика из B_3 осталась еще не удовлетворенной потребность $b'_3 = 80$.
5. Теперь переходим к заполнению клетки для неизвестного x_{23} и т.д. Через шесть шагов у нас останется одна база A_3 с запасом груза (остатком от предыдущего шага) $a'_3 = 200$ и один пункт B_5 с потребностью b_5=200 . Соответственно этому имеется одна свободная клетка, которую и заполняем, положив x_{35}=200. План составлен.
6. Базис образован неизвестными $x_{11},x_{12},x_{13},x_{23},x_{24},x_{34},x_{35}$. Правильность составленного плана легко проверить, подсчитав суммы чисел, стоящих в заполненных клетках по строкам и столбцам.
7. Общий объем перевозок в тонно-километрах для этого плана составит:

$$S_1 = 70 \cdot 170 + 50 \cdot 110 + 15 \cdot 20 + 40 \cdot 80 + 60 \cdot 70 + 11 \cdot 50 + 25 \cdot 200 = 30650$$

2. Метод наименьшей стоимости.

При этом методе на каждом шаге построения опорного плана первою заполняется та клетка оставшейся части таблицы, которая имеет наименьший тариф. Если такая клетка не единственная, то заполняется любая из них.

Пример

Пункты Отправления	Запасы	Пункты назначения				
		B_1	B_2	B_3	B_4	B_5
		170	110	100	120	200
A_1	300	70 **20**	50	15 **100**	80	70 **180**
A_2	150	80 **150**	90	40	60	85
A_3	250	50	10 **110**	90	11 **120**	25 **20**

1. В данном случае заполнение таблицы начинается с клетки для неизвестного x_{32}, для которого мы имеем значение $c_{32} = 10$, наименьше из всех значений c_{ij} . Эта клетка находится на пересечении третьей строки и второго столбца, соответствующим третьей базе A_3 и второму заказчику B_2.
2. Третья база A_3 может полностью удовлетворить потребность второго заказчика B_2 (a_3=250, b_2=110, $a_3 > b_2$) . Полагая x_{32} = 110, вписываем это значение в клетку x_{32} и исключаем из рассмотрения второй столбец. На базе A_3 остается изменённый запас $a'_3 = 140$.
3. В оставшейся новой таблице с тремя строками A_1, A_2, A_3 и четырьмя столбцами B_1, B_3, B_4, B_5 клеткой с наименьшим значением c_{ij} клетка, где c_{34}=11. Заполняем описанным выше способом эту клетку и аналогично заполняем следующие клетки. В результате оказываются заполненными (в

приведенной последовательности) следующие клетки: $x_{32} = 110, x_{34} = 120, x_{13} = 100, x_{35} = 20, x_{15} = 180, x_{11} = 20, x_{21} = 150$

4. На пятом шаге клеток с наименьшими значениями c_{ij} оказалось две (c_{11}=c_{15}=70) . Мы заполнили клетку для x_{15} , положив x_{15} = 180. Можно было выбрать для заполнения другую клетку, положив x_{11} = 170, что приведет в результате к другому опорному плану.
5. Общий объем перевозок в тонно-километрах для этого плана составит:

$$S_1 = 70 \cdot 20 + 15 \cdot 100 + 70 \cdot 180 + 80 \cdot 150 + 10 \cdot 110 + 11 \cdot 120 + 25 \cdot 20 = 30420$$

Замечание: В диагональном методе не учитываются величины тарифов, в методе же наименьшей стоимости эти величины учитываются, и часто последний метод приводит к плану с меньшими общими затратами (что и имеет место в нашем примере), хотя это и не обязательно.

Кроме рассмотренных выше способов иногда используется, так называемый, метод Фогеля. Суть его состоит в следующем: В распределительной таблице по строкам и столбцам определяется разность между двумя наименьшими тарифами. Отмечается наибольшая разность. Далее в строке (столбце) с наибольшей разностью заполняется клетка с наименьшим тарифом. Строки (столбцы) с нулевым остатком груза в дальнейшем в расчет не принимаются. На каждом этапе загружается только одна клетка. Распределение груза производится, как и ранее.