15. ЭММ. Транспортная задача. Нахождение опорного плана методом северо-западного угла. Нахождение опорного плана методом потенциалов. Транспортная задача с минимизацией времени. Сетевое планирование. Нахождение критического времени и критического пути. Задача сетевого планирования с вложением средств.

Постановка транспортной задачи

Имеется m пунктов отправления некоторого груза и n пунктов назначения этого груза.

Запас груза на пункте отправления A_i составляет a_i единиц, а заявка пункта назначения $B_j - b_j$ единиц, $i = 1, 2, \dots, m,\ j = 1, 2, \dots, n.$

Стоимость перевозки единицы груза из пункта отправления A_i в пункт назначения B_j составляет c_{ij} единиц.

Требуется найти план перевозок, имеющий минимальную общую стоимость.

Будем считать сначала, что сумма запасов равна сумме заявок (закрытая задача): $a_i + \dots + a_m = b_j + \dots + b_n$.

Обозначим x_{ij} – количество груза, перевозимого из пункта отправления A_i в пункт назначения B_j. Тогда задача примет следующий вид.

$$L = c_{11}x_{11} + c_{12}x_{12} + \dots + c_{mn}x_{mn} \to \min$$

Ограничения на запасы:

$$\begin{cases} x_{11} + x_{12} + \dots + x_{1n} = a_1 \\ \dots \\ x_{m1} + x_{m2} + \dots + x_{mn} = a_m \end{cases}$$

Ограничения на заявки:

$$\begin{cases} x_{11} + x_{21} + \dots + x_{m1} = b_1 \\ \dots \\ x_{1n} + x_{2n} + \dots + x_{mn} = b_n \end{cases}$$

Условия неотрицательности: $x_{11} \ge 0,\ x_{12} \ge 0,\ \dots,\ x_{mn} \ge 0.$

Для решения транспортной задачи применяются специальные методы, такие как метод потенциалов, венгерский метод и т.д.

Представим данные в виде таблицы.

	B_1	B_2	…	B_n	Всего
A_1	x_{11} $^{c_{11}}$	x_{12} $^{c_{12}}$	...	x_{1n} $^{c_{1n}}$	a_1
A_2	x_{21} $^{c_{21}}$	x_{22} $^{c_{22}}$	...	x_{2n} $^{c_{2n}}$	a_2
…	...	...	...	...	
A_m	x_{m1} $^{c_{m1}}$	x_{m2} $^{c_{m2}}$	...	x_{mn} $^{c_{mn}}$	a_m
Всего	b_1	b_2		b_n	Σ

Так как большинство $x_{ij} = 0,$ они записываться в таблицу не будут, а клетки в этом случае будут называться пустыми.

Решение задачи проводится в два этапа:

1) Нахождение опорного плана.
2) Нахождение оптимального плана.

Для первого этапа будем применять метод северо-западного угла, а для второго этапа – метод потенциалов.

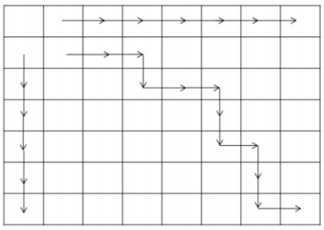
В отличие от задачи линейного программирования транспортная задача всегда имеет решение. Кроме того, если все запасы и заявки являются целыми числами, то решение также будет целочисленным.

Нахождение опорного плана методом северо-западного угла

Алгоритм северо-западного угла

1) Если $a_1 > b_1$, то полагаем, что $x_{11} = b_1$ и делаем шаг вправо;
2) Если $a_1 < b_1$, то полагаем, что $x_{11} = a_1$ и делаем шаг вниз;
3) Если $a_1 = b_1$, то получается вырожденный случай;
4) Вычеркиваем из таблицы заполненную строку или столбец, уменьшаем количество груза в незаполненном столбце или строке, переходим к первому пункту.

Повторяем эту процедуру до тех пор, пока не будет найден опорный план.



Подсчитаем число непустых клеток $(n-1)+(m-1)+1=m+n-1$.

Невырожденный план должен иметь ровно $m+n-1$ непустую клетку.

Это требование не является достаточным для невырожденного плана.

Нахождение оптимального плана методом потенциалов

Составим каждому пункту отправления A_i некоторое число $\tilde{a}_i$ и каждому пункту назначения B_j некоторое число $\tilde{b}_j$, так чтобы для любой непустой клетки $\tilde{a}_i+\tilde{b}_j=c_{ij}$.

Эти числа $\tilde{a}_i$ и $\tilde{b}_j$ называются псевдоплатежами.

Для их нахождения получаем линейную систему из $m+n-1$ уравнений на $m+n$ неизвестных. Одну неизвестную можно взять произвольно (обычно $\tilde{a}_1=0$). Тогда остальные псевдоплатежи находятся единственным образом по цепочке.

Далее находим псевдостоимости $\tilde{c}_{ij}=\tilde{a}_i+\tilde{b}_j$ для всех клеток. Записываем их в левый угол каждой клетки.

	B_1	B_2	$\ldots$	B_n	Всего
A_1	$\tilde{c}_{11}$ x_{11} c_{11}	...	...	...	a_1
A_2	...	...	...	...	a_2
$\ldots$	...	...	...	...	...
A_m	...	...	...	...	a_m
Всего	b_1	b_2	...	b_n	Σ

Если все $\tilde{c}_{ij}\le c_{ij}$, то оптимальный план найден.

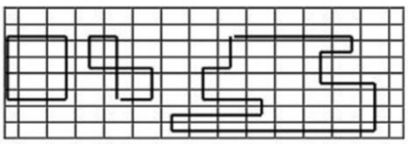
Любая клетка, в которой $\tilde{c}_{ij}>c_{ij}$ может быть взята в качестве разрешающей.

Заполняя выбранную клетку, необходимо изменить объемы поставок, записанных в ряде других занятых клеток и связанных с заполняемой клеткой так называемым циклом.

Циклом пересчета называется замкнутая ломаная линия, все стороны которой горизонтальны или вертикальны, одна вершина находится в пустой клетке, а остальные – в непустых.

План называется невырожденным, если для любой пустой клетки найдется единственный цикл пересчета.

Примеры некоторых циклов показаны на рисунке.



Находим цикл для разрешающей клетки. Далее расставляем знаки: в разрешающую клетку ставим «+», а далее «–» и «+» чередуются по обходу цикла. В любом цикле только четкое число вершин, поэтому расстановка знаков не зависит от направления обхода цикла.

Далее находим минимальное количество груза в клетках со знаком «–». Если этот минимум достигается более чем в одной клетке, получается вырожденный случай. Если такого нет, то делаем пересчет таблицы: в клетке с «+» добавляем этот минимум, в клетке с «–» вычитаем его.

При этом пересчете количество непустых клеток и сумма строкам и столбцам не меняется. После пересчета находим псевдоплатежи.

Транспортная задача с минимизацией времени

Пусть имеются пункты отправления груза A_1, A_2, ..., A_m и пункты назначения груза B_1, B_2, ..., B_n. Запасы груза на пунктах отправления A_i составляет a_i единиц, а заявки пунктов назначения B_j составляют b_j единиц. Полагаем, что $\sum a_i = \sum b_j$. Известно время t_{ij} перевозки груза из пункта отправления A_i в пункт назначения B_j (не зависит от количества груза).

Ставится задача найти минимальное время перевозок. Считается, что все перевозки начинаются одновременно.

$$L = \max_{x_{ij}>0} t_{ij} \to \min$$

$$\begin{cases} x_{11} + \ldots + x_{1n} = a_1, \\ \ldots \\ x_{m1} + \ldots + x_{mn} = a_m. \end{cases} \qquad \begin{cases} x_{11} + \ldots + x_{m1} = b_1, \\ \ldots \\ x_{1n} + \ldots + x_{mn} = b_n. \end{cases}$$

$$x_{11} \ge 0,\ x_{12} \ge 0,\ \ldots,\ x_{mn} \ge 0.$$

Эта задача является задачей нелинейного программирования, так как L нелинейна. Для её решения можно применить метод "вычёркивания клеток". Вычёркиваем из таблицы клетки, в которых t_{ij} максимально. Пытаемся найти план, не использующий этих клеток. Если план найти удается, то вычеркиваем следующие клетки. Если нет, то предыдущий полученный план является оптимальным.

Некоторые случаи, когда таблицу заполнить нельзя:

1. В строке (столбце) вычеркнуты все клетки.
2. В строке (столбце) остается одна невычеркнутая клетка и сумма в строке (столбце) больше суммы столбца (строки).

При заполнении таблицы надо рассмотреть все возможные способы заполнения. Можно ограничится только невырожденными, если они есть.

Проще найти план, если начинать заполнение таблицы со строк и столбцов с наименьшим количеством невычеркнутых клеток.

Сетевое планирование

Основные понятия

Пусть имеется некоторое количество видов работ $A_1, \ldots, A_n$. Некоторые из этих работ опираются на другие работы. Известна продолжительность выполнения каждой работы, и будем обозначать время выполнения работы A_i через t_i. Некоторые работы могут выполняться одновременно. Требуется определить минимальное время выполнения всех работ.

Будем называть работу работой первого ранга, если она не опирается ни на какие другие работы. Выполнение всех работ первого ранга может начинаться в начальный момент времени (начальный момент времени будем брать равным нулю, тогда время завершения последней работы определяет время выполнения комплекса работ). Работа называется работой ранга k, если она опирается на работы рангов меньше k и опирается, по крайней мере, на одну работу ранга $k-1$. Для комплекса работ можно построить схему – временной график.

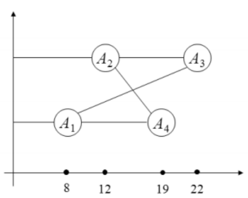
Работа называется критической, если при задержке начала её выполнения увеличивается время выполнения всего комплекса работ.

Если же выполнение некоторой работы может быть отложено на некоторый промежуток и при этом время выполнения всего комплекса работ не увеличится, то она называется некритической. Максимальный промежуток времени, на который мы можем отложить выполнение некритической работы без увеличения времени выполнения всех работ, называется резервом времени. У критических работ резерва времени нет.

Последовательность критических работ называется критическим путём. Критический путь может быть найден следующим образом. Для работ, которые заканчиваются последними, надо взять работы, на которые они опираются существенно. Для этих работ тоже взять работы, на которые они опираются существенно и т.д. Пройдя по цепочке от конца к началу, можно найти все критические работы.

Нахождение критического времени и критического пути

Имеется комплекс из четырех работ A_1, A_2, A_3, A_4. Работы A_3 и A_4 опираются на A_1 и A_2 каждая. Время выполнения работ: $t_1 = 8$, $t_2 = 12$, $t_3 = 10$, $t_4 = 7$. Надо найти критическое время, критический путь и резервы времени всех работ. Решим задачу с помощью временного графика.



$T_{кр} = 22$.

Критический путь: $P_{кр}: A_2 \to A_3$ $r(A_2) = r(A_3) = 0, r(A_1) = 4, r(A_4) = 3.$

Описанный выше графический способ построения и анализа плана работ пригоден только в случае, когда планируемый комплекс не слишком сложен (по количеству работ и логических связей).

На практике часто встречаются комплексы работ, состоящие из огромного числа звеньев (порядка тысяч и более), сложным образом опирающихся друг на друга. Естественно, что в таких случаях вычерчивание сетевого графика вручную – тяжелое и неблагодарное занятие, так как основное преимущество сетевого графика – его наглядность – при этом теряется.

Эту задачу можно решить с помощью таблицы:

Работы	A_1	A_2	A_3	A_4
Опираются на			A_1, A_2	A_1, A_2
Время	8	12	10	7
Общее время	8	12	**22**	19

Для нахождения резервного времени найдем все пути.

$p_1 : A_1 \to A_3, \qquad t(p_1) = 8 + 10 = 18.$

$p_2 : A_1 \to A_4, \qquad t(p_2) = 8 + 7 = 15.$

$p_3 : A_2 \to A_4, \qquad t(p_3) = 12 + 7 = 19.$

$r(A_2) = r(A_3) = 0.$

$r(A_1) = T_{кр} - \max(t(p_1), t(p_2)) = 22 - \max(18,15) = 4.$

$r(A_4) = T_{кр} - \max(t(p_2), t(p_3)) = 22 - \max(15,19) = 3.$

Если работы не упорядочены по рангу, задачу решаем следующим образом. Упорядочиваем все работы по возрастанию ранга, вводя при необходимости новые обозначения. Сначала вводим обозначения для работ, которые не опираются на другие работы. Затем обозначаем те работы, которые опираются на уже имеющие новые обозначения. Таких способов введения новых обозначений имеется несколько.

Определяем время завершения каждой работы, прибавляя к её продолжительности время работы, на которую она опирается существенно. Критическое время выполнения работ – это максимальное время завершения некоторой работы. Критический путь определяется по цепочке от конца к началу, беря на каждом шаге работы, на которые данная работа опирается существенно. Другими словами, берутся работы, которые завершаются последними из тех на которые опирается данная работа.

Задача сетевого планирования с вложением средств

Пусть имеется комплекс работ $A_1, A_2, \ldots, A_n$ и некоторое количество средств, которые могут быть вложены в эти работы. Известное критическое время выполнения всех работ, если средства в них не вкладываются и зависимость времени выполнения работ от вложенных средств. Пусть критическое время, если средства не вкладываются, равно $T_{кр}^0$.

Необходимо с помощью вложения минимального количества средств снизить это время до T_0. Другими словами, требуется определить какие количества средств необходимо вложить в определенные виды работ, так чтобы время критическое выполнения работы не превосходило T_0, а общее количество средств было минимальным. Если зависимость времени выполнения работы от количества вложенных средств линейная, то мы получаем задачу линейного программирования, которая может быть решена симплекс-методом.

При решении такой задачи возможны два случая:

1) При вложении средств критический путь не изменится. В этом случае достаточно рассмотреть вложения средств только в критические работы.

2) При вложении средств критический путь может измениться. В этом случае вложения средств могут быть сделаны во все работы.

Для определения того, что критический путь не меняется можно использовать одно из двух достаточных условий. Достаточно выполнения одного из них. В то же время невыполнение этих двух условий не говорит о том, что критический путь должен измениться.

Первое условие заключается в том, что если при максимальном вложении средств в критические работы они остаются критическими, то критический путь не меняется.

Второе условие заключается в том, что если при снижении времени выполнения критических работ на $\Delta T = T_{кр}^0 - T_0$ они остаются критическими, то критический путь не меняется.

Возможны случаи выполнения обоих условий, выполнения какого-то одного из них и невыполнение обоих условий. Последний случай не означает, что критический путь меняется.

Рассмотрим решение задачи в случае, когда критический путь не меняется. Задача в этом случае имеет более простой вид, чем в случае, когда критический путь может измениться.

Пусть известен критический путь $A_{i_1} \to A_{i_2} \to \cdots \to A_{i_k}$ и критическое время $T_{кр}^0$, если средства не вкладываются. Дано $T_0 < T_{кр}^0$ и зависимости времени выполнения критических работ от количества вложенных средств $t_{i_j} = t_{i_j}^0\left(1 - \alpha_{i_j} x_{i_j}\right)$, $0 \le x_{i_j} \le b_{i_j}$, $j = 1, \ldots, k$.

Количество вложенных средств определяется по формуле

$L = x_{i_1} + x_{i_2} + \cdots + x_{i_k}.$

Тогда задача принимает следующий вид.

$$L = x_{i_1} + x_{i_2} + \ldots + x_{i_k} \to \min,$$

$$\begin{cases} T_{кр}^0 - \alpha_{i_1} x_{i_1} t_{i_1}^0 - \alpha_{i_2} x_{i_2} t_{i_2}^0 - \ldots - \alpha_{i_k} x_{i_k} t_{i_k}^0 \le T_0, \\ x_{i_1} \le b_{i_1}, \\ x_{i_2} \le b_{i_2}, \\ \ldots, \\ x_{i_k} \le b_{i_k}, \end{cases}$$

$$x_{i_1} \ge 0,\ x_{i_2} \ge 0,\ \ldots,\ x_{i_k} \ge 0.$$

Получаем задачу линейного программирования. Рассмотрим пример решения такой задачи симплекс-методом.