# Математические методы сетевого моделирования

## Транспортные сети и потоки

Транспортные сети, пожалуй, одно из самых распространенных и высокоавтоматизированных приложений математического сетевого моделирования. Они используются в навигаторах, в логистике, в электрических и компьютерных сетях, в коммунальных службах, при прокладке трубопроводов и маршрутов городского транспорта и др.

### Задачи о соединении

В задаче о соединении требуется обеспечить минимальную стоимость проекта по созданию сети, если известны затраты на соединение каждой пары пунктов:

* строительство сети автомобильных или железных дорог между городами;
* строительство сети нефтепроводов и газопроводов;
* прокладка кабельной сети между некоторыми географическими точками;
* планирование авиарейсов;
* и др.

Все возможные соединения пунктов можно представить в виде графа, каждому ребру которого назначен вес (стоимость). Из всех этих вариантов соединения, нужно выбрать самый дешевый, который соединит все пункты.

Чтобы решить задачу о соединении необходимо найти *остов наименьшего веса*.

**Остов** графа – это такая его часть (подграф), которая включает все его вершины и является связной (деревом).

**Алгоритм Краскала (Крускала)** для построения остова наименьшего веса.Всегда содержит (n – 1) шагов, n – число вершин. На каждом шаге выбирается ребро, которое следует включить в остов.

Шаг 1. Выбрать самое короткое ребро в графе.

Шаг 2...(n – 1). Из оставшихся ребер выбрать самое короткое ребро, которое не образует цикла с уже выбранными ребрами.

Для удобства можно на каждом шаге вычеркивать ребра, которые образуют цикл с уже имеющимися.

**Пример 1. Задача о соединении городов.**

Известна стоимость постройки дорог между каждой парой населенных пунктов. Необходимо построить дорожную сеть наименьшей стоимости, соединяющую все города.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Поселок | A | B | C | D |
| A |  | 11 | 13 | 10 |
| B |  |  | 6 | 9 |
| C |  |  |  | 10 |
| D |  |  |  |  |

Этот граф можно изобразить следующим образом (если бы у нас была карта, можно было бы расположить поселки по ней):

A

B

C

D

(11)

(13)

(10)

(6)

(9)

(10)

**Решение:**

1. Добавляем B-C.
2. Добавляем B-D.
3. Добавляем A-D.

S= 9+6+10=25.

**Ответ:**

S = 25.

Задачу можно решать и «от обратного», удаляя из сети наиболее «дорогие» ребра.

**Алгоритм**

1. Найти ребро наибольшего веса. Если его удаление не нарушает связности графа, удалить это ребро.
2. Продолжать до тех пор, пока граф не станет деревом (не останется n–1 ребро).

**Пример 2.**

Построить остов минимального веса для графа, показанного на рисунке, удалив «лишние» ребра.

Обратите внимание, граф неполный, т.е. некоторые «дороги» невозможно построить в принципе. В матрице смежности такие ребра обозначаются прочерком или знаком бесконечности ∞.

(6)

(5)

(3)

(1)

(4)

(8)

(5)

(3)

(2)

(6)

A

B

C

D

E

G

F

**Решение:**

1. D-G (8).
2. B-E (6).
3. B-C (5).
4. E-G (5).

S = 6 + 1+ 3 + 4 + 3 + 2 = 19.

**Ответ:** S = 19.

(6)

(3)

(1)

(4)

(3)

(2)

**Пример 3.**

Необходимо провести свет в 8 поселков района. Стоимость прокладки ЛЭП между населенными пунктами показана в таблице. Разработать наиболее экономичную схему электрификации.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Поселок | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** |
| **1** |  | 13 | **9** | 14 | 14 | - | 20 | 18 |
| **2** |  |  | **6** | - | 15 | **9** | 21 | - |
| **3** |  |  |  | **12** | - | 11 | 17 | - |
| **4** |  |  |  |  | **8** | 17 | - | - |
| **5** |  |  |  |  |  | 16 | - | - |
| **6** |  |  |  |  |  |  | 19 | - |
| **7** |  |  |  |  |  |  |  | 31 |
| **8** |  |  |  |  |  |  |  |  |

**Решение:**

1. 2-3 (6).
2. 4-5 (8).
3. 1-3 (9).
4. 2-6 (9).
5. 3-4 (12)
6. 3-7 (17)
7. 1-8 (18)

S = 6 + 8 + 9 + 9 + 12 + 17 + 18 = 79.

**Ответ:** S = 79.

### Кратчайший маршрут

Найти кратчайший путь от одной вершины и до всех остальных вершин (или до другой конкретной вершины).

Именно эта задача решается при прокладке маршрута на навигаторе, установлении телефонного или интернет-соединения. Она может возникать и при принятии экономических и управленческих решений. Граф может описывать, например, возможные пути модернизации производства.

**Волновой метод (алгоритм Дейкстра).** Все ребра в сети должны иметь неотрицательный вес (≥0).

В алгоритме поочередно отмечаются вершины. Отметить вершину – значит найти самый короткий путь до нее, *отметка* – длина маршрута до вершины и номер предыдущей вершины (W;j).

Шаг 1. Отмечаем начальную вершину.

Шаг 2. Среди неотмеченных вершин, смежных с уже отмеченными, отметить ту, к которой ведет наименьший путь.

Шаг 3. Повторять шаг 2 до тех пор, пока не будут отмечены все вершины (или пока не будет достигнута целевая вершина).

Аналогично можно найти путь максимальной длины, только выбирать наибольший путь на шаге 2.

Необходимо учитывать, что не во всех задачах длины ребер нужно складывать, чтобы получить длину маршрута.

**Пример 4**.

Найти кратчайшее расстояние от вершины a до вершины e.

154

289

105

130

73

85

167

211

1270

**Решение:**

|  |  |  |
| --- | --- | --- |
| Шаг | Отмеченные вершины | Новые пути |
|  | a(0;-) | W(a-b) = **154**  W(a-d) = 289 |
|  | b(154;a) | W(a-b-c) = 154 + 105 = **259**  W(a-b-d) = 154 + 130 = **284** |
|  | c(259;b) | W(a-b-c-d) = 259 + 73 = **332**  W(a-b-c-e) = 259 + 211 = 470  W(a-b-c-f) = 259 + 85 = 334 |
|  | d(284;b) | W(a-b-d-e) = 284 + 167 = **332** |
|  | e(332;d) |  |

**Ответ:**

W(a-b-d-e) = 332.

**Пример 5**. **Максимальная надежность**

Надежность определяется вероятностью выполнения поставленной задачи.

Определить наиболее надежный путь для передачи сообщения от узла 1 к узлу 6.

0,93

0,85

0,77

0,95

0,93

0,86

0,99

0,94

0,95

Максимальная надежность = минимальная ошибка

p(1-2)=p(1)+p(2)–p(1)\*p(2)

**Решение:**

|  |  |  |
| --- | --- | --- |
| Шаг | Отмеченные вершины | Новые пути |
|  | 1(0;-) | p(1-3)=**0,07**  p(1-5)=0,15 |
|  | 3(0,07;1) | p(1-3-2)=0,07+0,05-0,07\*0,05=**0,1165**  p(1-3-4)=0,07+0,07-0,07\*0,07=0,1351  p(1-3-5)=0,07+0,05-0,07\*0,05=**0,1165** |
|  | 2(0,1165;3) | p(1-3-2-4)=0,1165+0,23-0,1165\*0,23=0,3197 |
|  | 5(0,1165;3) | p(1-3-5-4)=0,1165+0,01-0,1165\*0,01=**0,1253**  p(1-3-5-6)=0,1165+0,06-0,1165\*0,06=**0,1695** |
|  | 4(0,1253;5) | p(1-3-5-4-6)=0,1253+0,14-0,1253\*0,14=0,2402 |
|  | 6(0,1695;4) |  |

**Ответ:**

W(1-3-5-6)=1-**0,1695=** **0,8305**

**Пример 6**

Фирма выполняет заказ, состоящий из трех этапов работ. Фирма может выполнять заказ самостоятельно или нанять субподрядчиков (А или Б). В таблице указаны сроки и стоимость выполнения этапов. Если заказывать сразу несколько этапов у одного субподрядчика, то сроки и стоимость выполнения можно снизить.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Исполнитель | Этапы, дн./тыс.руб. | | | | | |
| I | II | III | I + II | II + III | I + II + III |
| А | 15/36 | 7/40 | 20/80 | 21/72 | 25/112 | 38/145 |
| Б | 18/32 | 6/40 | - | 22/70 | - | - |
| Свои силы | 20/15 | 6/50 | 26/65 | - | - | - |

Example Firm.emf

**Критерий: приоритет времени выполнения T (time)**

Если время совпадает, то выбираем по минимальной стоимости.

|  |  |  |
| --- | --- | --- |
| Шаг | Отмеченная вершина | Новые пути |
| 1. | S(0;-) | T(S-I)=15  T(S-II)=21  T(S-III)=38 |
| 2. | I(15;S) | T(S-I-II)=15+6=21  T(S-I-III)=15+25=40 |
| 3. | II(21;S) | T(S-II-III)=21+20=41 |
| 4. | III(38;S) |  |

**Критерий: приоритет стоимости C (cost)**

Если стоимость совпадает, то выбираем по минимальному времени.

|  |  |  |
| --- | --- | --- |
| Шаг | Отмеченная вершина | Новые пути |
| 1. | S(0;-) | C(S-I)=15  C(S-II)=70  C(S-III)=145 |
| 2. | I(15;S) | C(S-I-II)=15+40=55  C(S-I-III)=15+112=127 |
| 3. | II(55;I) | C(S-I-II-III)=55+65=120 |
| 4. | III(120;II) |  |

**Критерий: единый показатель W (дни => тыс.руб.)**

Будем считать, что каждый день, потраченный на выполнение заказа, стоит 3тыс.руб.)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Исполнитель | Этапы, дн./тыс.руб. | | | | | |
| I | II | III | I + II | II + III | I + II + III |
| А | =15\*3+36=81 | =7\*3+40=61 | =20\*3+80=140 | =21\*3+72=135 | =25\*3+112=187 | =38\*3+145=259 |
| Б | =18\*3+32=86 | =6\*3+40=58 | - | =22\*3+70=136 | - | - |
| Свои силы | =20\*3+15=75 | =6\*3+50=68 | =26\*3+65=143 | - | - | - |

|  |  |  |
| --- | --- | --- |
| Шаг | Отмеченная вершина | Новые пути |
| 1. | S(0;-) | W(S-I)=75  W(S-II)=135  W(S-III)=259 |
| 2. | I(75;S) | W(S-I-II)=75+58=133  W(S-I-III)=75+187=262 |
| 3. | II(133;I) | W(S-I-II-III)=133+140=273 |
| 4. | III(259;S) |  |

**Ответ**:

min T: T(S-III)=38. Все этапы работ следует заказать у субподрядчика.

min C: C(S-I-II-III)=120. I и III этапы следует выполнять своими силами, а II заказать у субподрядчика Б.

min W: W(S-III)=38/145 (259). Все этапы работ следует заказать у субподрядчика.

**Пример 7**.

Построить граф транспортной сети и найти кратчайшие пути от Самары до каждого из указанных городов.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Город | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| Волгоград | 1 |  | - | - | - | 505 | - | 376 | - | - |
| Казань | 2 |  |  | 1041 | - | - | 361 | - | 363 | 243 |
| Нижний Новгород | 3 |  |  |  | 435 | - | - | - | - | 460 |
| Пенза | 4 |  |  |  |  | - | - | 224 | 363 | 298 |
| Ростов-на-Дону | 5 |  |  |  |  |  | - | - | - | - |
| Самара | 6 |  |  |  |  |  |  | 442 | 103 | 235 |
| Саратов | 7 |  |  |  |  |  |  |  | - | - |
| Тольятти | 8 |  |  |  |  |  |  |  |  | 193 |
| Ульяновск | 9 |  |  |  |  |  |  |  |  |  |

Samara.emf

|  |  |  |
| --- | --- | --- |
| Шаг | Отмеченные вершины | Новые пути |
|  | 6(0;-) | S(6-2)=**361**  S(6-7)=**442**  S(6-8)=**103**  S(6-9)=**235** |
|  | 8(103;6) | ~~S(6-8-2)=103+363=466~~  S(6-8-4)=103+363=**466**  ~~S(6-8-9)=103+193=296~~ |
|  | 9(235;6) | S(6-9-3)=235+460=**695**  ~~S(6-9-4)=235+298=533~~ |
|  | 2(361;6) | ~~S(6-2-3)=361+1041=1402~~ |
|  | 7(442;6) | S(6-7-1)=442+376=**818**  ~~S(6-7-4)=442+224=666~~ |
|  | 4(466;8) | ~~S(6-8-4-3)=466+435=901~~ |
|  | 3(695;9) | - |
|  | 1(818;1) | S(6-7-1-5)=818+505=**1323** |
|  | 5(1323;1) |  |

**Ответ**:

|  |  |  |  |
| --- | --- | --- | --- |
| Волгоград | 1 | 818 | (6-7-1) |
| Казань | 2 | 361 | (6-2) |
| Н. Новгород | 3 | 695 | (6-9-3) |
| Пенза | 4 | 466 | (6-8-4) |
| Ростов-на-Дону | 5 | 1323 | (6-7-1-5) |
| Саратов | 7 | 442 | (6-7) |
| Тольятти | 8 | 103 | (6-8) |
| Ульяновск | 9 | 235 | (6-9) |