Транспортные сети

Поиск максимального потока в сети

Транспортная задача

Может возникать в физике, экономике и т.д.

На отдельные компоненты транспортной сети (сеть железнодорожных, автомобильных и т.д. путей; сеть трубопроводов и т.д.) наложены ограничения – их максимально допустимая нагрузка.

Необходимо определить максимально возможное количество пассажиров, товара, продукта и т.д., которое можно провезти по этой сети и каким образом.

Мы построим **графовую дискретную модель** этой транспортной задачи и решим ее в этой модели.

Математик **Джордж Бернард Данциг**, с **1941** года работая в отделе статистического управления Военно-воздушных сил США в Вашингтоне, впервые решил задачу о максимальном потоке в ходе подготовки воздушного моста во время блокады Западного Берлина.

В **1951** году Джордж Данциг впервые сформулировал задачу в общем виде. В **1955** году, **Лестер Форд** и **Делберт Фалкерсон** впервые построили алгоритм, специально предназначенный для решения этой задачи. Их алгоритм получил название алгоритм Форда-Фалкерсона.

В **2010** году исследователи **Джонатан Кёлнер** и **Александер Мондры** из МТИ вместе со своими коллегами **Дэниелем Спилманом** из Йельского университета и **Шень-Хуа Тенем** из Южно-Калифорнийского университета продемонстрировали очередное улучшение алгоритма.

Дан ориентированный граф (**транспортная сеть**) G=(V, E), вершина графа s (источник) и вершина t (сток).

Каждой дуге (i, j) приписана некоторая **пропускная способность** $c(i,j) \geq 0$ (без потери общности считаем её целочисленной величиной), определяющая **максимальное** значение потока, который может протекать по данной дуге.

Потоком в сети называют целочисленную функцию $f(i, j)$, заданную на множестве дуг E и обладающей следующими свойствами:

1. **Ограничение потока пропускной способностью**

Для любой дуги $(i, j) \in E$ выполняется неравенство $f(i, j) \le c(i, j)$.

2. Сохранение потока

Для любой вершины q∈V, q≠s и q≠t выполняется равенство

$$\sum_{\substack{i\in V\\(i,q)\in E}} f(i,q) = \sum_{\substack{j\in V\\(q,j)\in E}} f(q,j)$$

Т. е. сумма потока, заходящего в q, равна сумме потока, выходящего из q (поток без потерь и накоплений)

Требуется определить значение **максимального потока**, который можно пропустить от источника s к стоку t, и его распределение по дугам.

Пример

У компании Lycky Puck в Ванкувере есть фабрика (источник s), производящая хоккейные шайбы, а в Виннипеге – склад (сток t), где эти шайбы хранятся. Компания арендует места на грузовиках других фирм для доставки шайб с фабрики на склад. Поскольку грузовики ездят по определенным маршрутам (ребрам) между городами (вершинами) и имеют ограниченную грузоподъемность, компания Lycky Puck может перевозить не более c(u,v) ящиков в день между каждой парой городов u и v. Компания Lycky Puck не может повлиять на маршруты и пропускную способность. Ее задача – определить, какое наибольшее количество ящиков в день можно отгружать, и затем производить именно такое количество, поскольку не имеет смысла производить шайб больше, чем можно отправить на склад.

Методы решения задачи

Линейное программирование

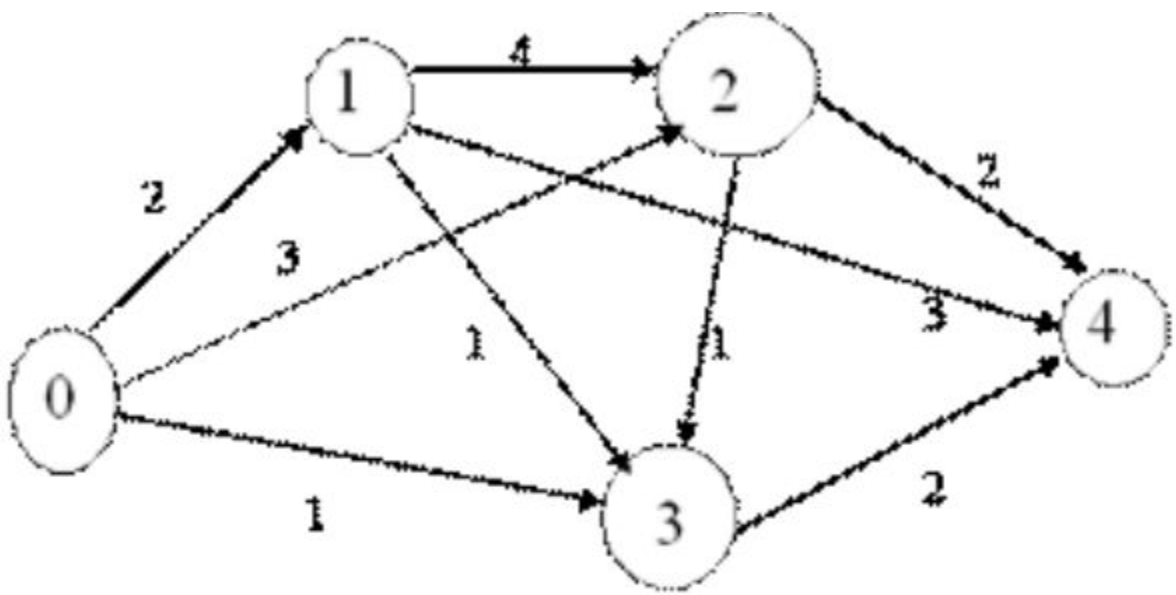
Представить задачу о максимальном потоке как задачу линейного программирования. Переменными являются потоки по рёбрам, а ограничениями - сохранение потока и ограничение пропускной способности.

Алгоритм Форда-Фалкерсона

Найти любой увеличивающий путь. Увеличить поток по всем его рёбрам на минимальную из их остаточных пропускных способностей. Повторять, пока увеличивающий путь есть. Алгоритм работает только для целых пропускных способностей.

Пример 1

Дадим формулировку **задачи о максимальном потоке в терминах линейного программирования**. Пусть X_{KM} - объем перевозок из пункта K в пункт M. K = 0,1,2,3, M = 1,2,3,4, причем перевозки возможны лишь в пункт с большим номером. Значит, всего имеется 9 переменных X_{KM}, а именно, X_{01}, X_{02}, X_{03}, X_{12}, X_{13}, X_{14}, X_{23}, X_{24}, X_{34}.



s=0
t=4

Задача линейного программирования, нацеленная на максимизацию потока, имеет вид:

$F \rightarrow \max$,

$X_{01} + X_{02} + X_{03} = F$

$-X_{01} + X_{12} + X_{13} + X_{14} = 0$

$-X_{02} - X_{12} + X_{23} + X_{24} = 0$

$-X_{03} - X_{13} - X_{23} + X_{34} = 0$

$-X_{14} - X_{24} - X_{34} = - F$

$X_{01} \leq 2$

$X_{02} \leq 3$

$X_{03} \leq 1$

$X_{12} \leq 4$

$X_{13} \leq 1$

$X_{14} \leq 3$

$X_{23} \leq 1$

$X_{24} \leq 2$

$X_{34} \leq 2$

$X_{КМ} \geq 0$, К, М = 0, 1, 2, 3, 4

$F \geq 0$.

Разрезом называют множество дуг, удаление которых из сети приводит к «разрыву» <u>**всех**</u> путей, ведущих из s в t.

Пропускная способность разреза – это суммарная пропускная способность дуг, его составляющих.

!!! Найти разрезы в примере 1

Теорема Л. Форда и Д. Фалкерсона:

Величина каждого потока из s в t не превосходит пропускной способности минимального разреза, разделяющего s и t, причем поток, достигающий этого значения, существует.

(Величина максимального потока в транспортной сети равна величине минимального разреза в ней)

!!! Найти минимальный разрез в примере 1

С алгоритмической точки зрения эта теорема малопродуктивна.

Генерация всех подмножеств дуг и проверка, является ли очередное подмножество разрезом – «лобовое решение», приводит к высокой сложности алгоритма.

Кроме того, данный факт не помогает найти способ распределения максимального потока по дугам.

Алгоритм Форда-Фалкерсона

«Техника меток» Л. Форда и Д. Фалкерсона заключается в последовательном (итерационном, поиском в ширину) построении максимального потока путем поиска на каждом шаге **увеличивающей цепи**, то есть пути, по которому можно увеличить поток.

При этом узлы (вершины графа) специальным образом помечаются. Отсюда и возник термин «метка».

Алгоритм Форда-Фалкерсона

Что представляет из себя метка вершины?

- первая цифра в метке – это номер вершины, из которой идет поток в данную вершину;
- вторая цифра в метке – численное значение потока, который можно передать в данную вершину.

Алгоритм Форда-Фалкерсона

На каждом шаге алгоритма вершины сети могут находиться в одном из трех состояний:

- вершина не имеет метки;
- вершине присвоена метка, и она не просмотрена, т. е. не все смежные с ней вершины обработаны;
- вершине присвоена метка, и она просмотрена.

Алгоритм Форда-Фалкерсона

Как только вершина-сток становится помеченной, это говорит о том, что очередная увеличивающая поток цепочка найдена, итоговый суммарный поток необходимо увеличить на величину потока найденной цепочки, и перейти к следующему шагу алгоритма.

Алгоритм Форда-Фалкерсона

Дуга $e=(u, v)$ сети является допустимой дугой из u в v относительно потока f, если

- $e=(u, v)$ и $f(e)<c(e)$ (дуги первого типа, **прямые**);
- $e=(v, u)$ и $f(e)>0$ (дуги второго типа, **обратные**).

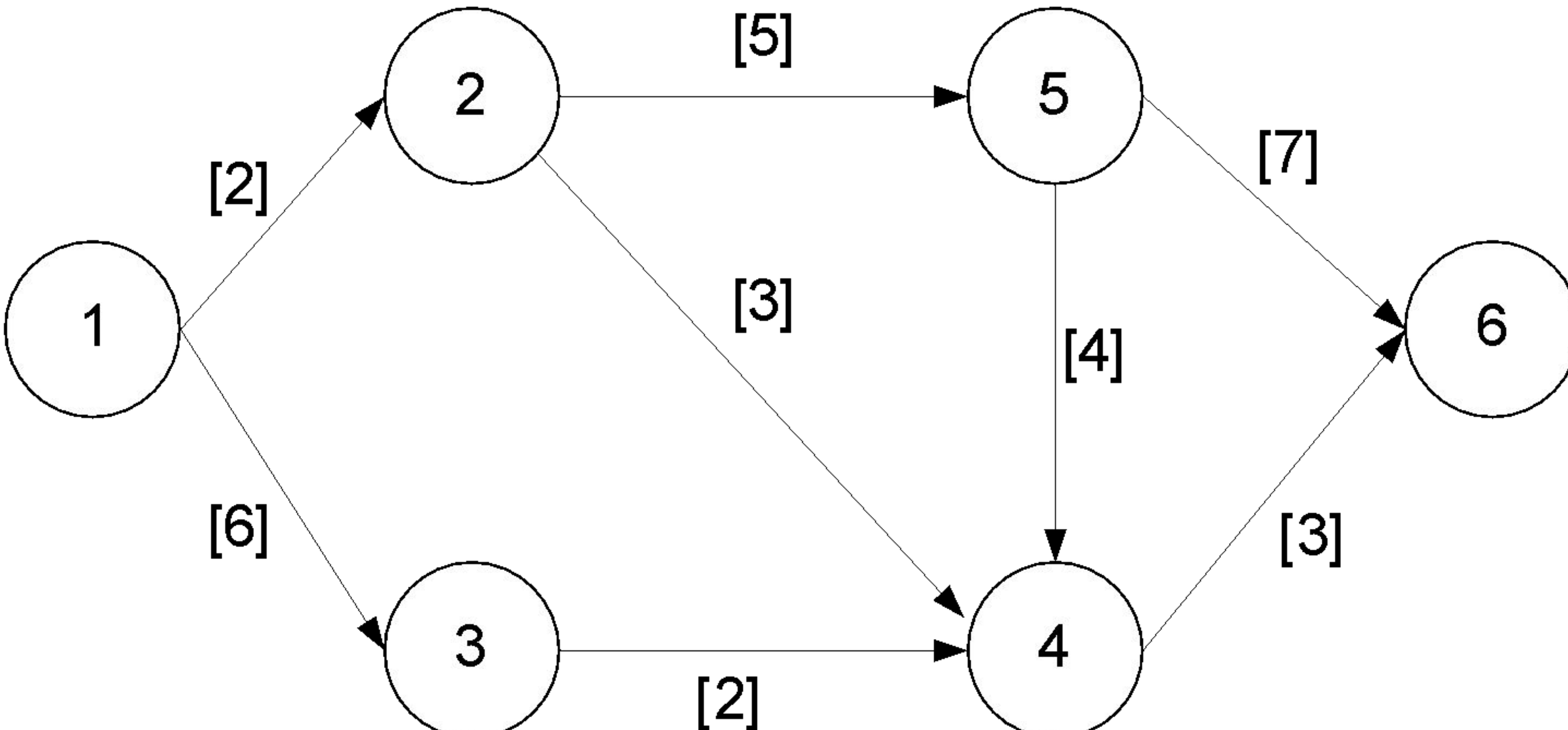
Второе условие говорит о том, что допустимыми являются и дуги, <u>входящие</u> в вершину u, по которым «уже пропущен ненулевой поток».

Пример 2

s=1

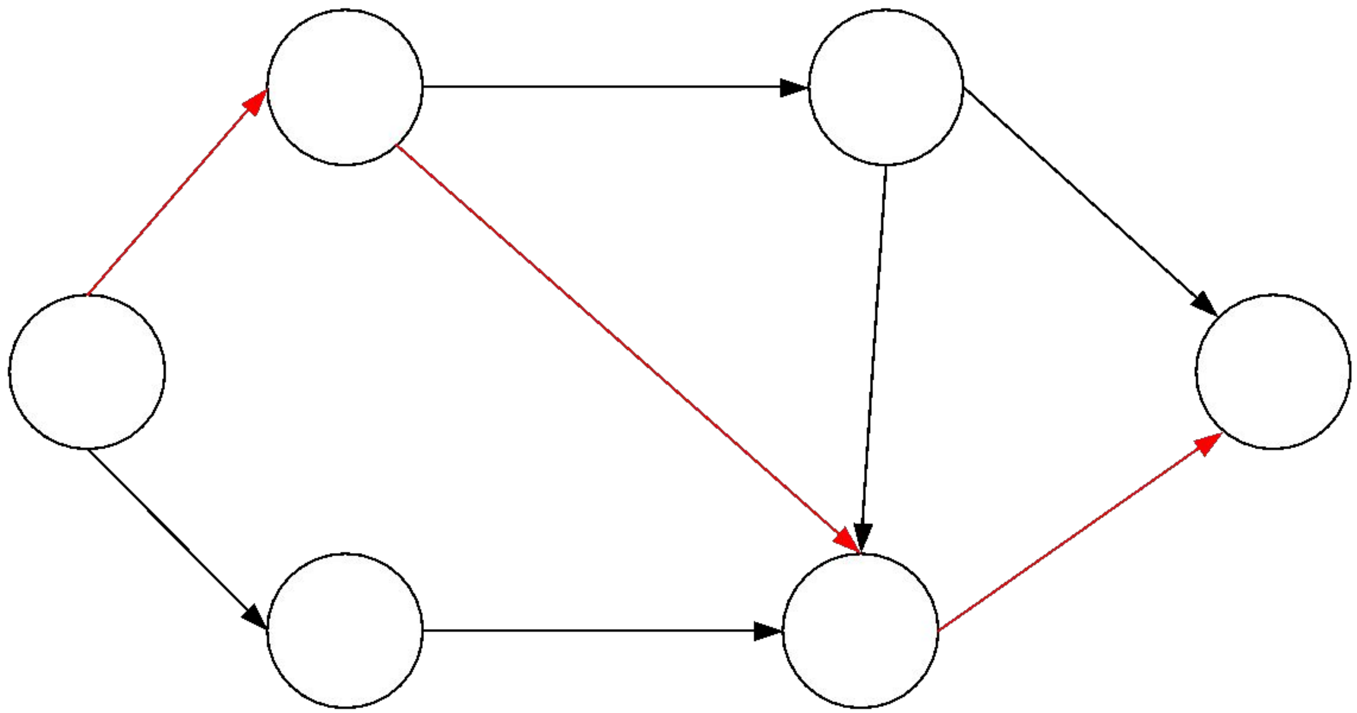
t=6

Найдите минимальный разрез



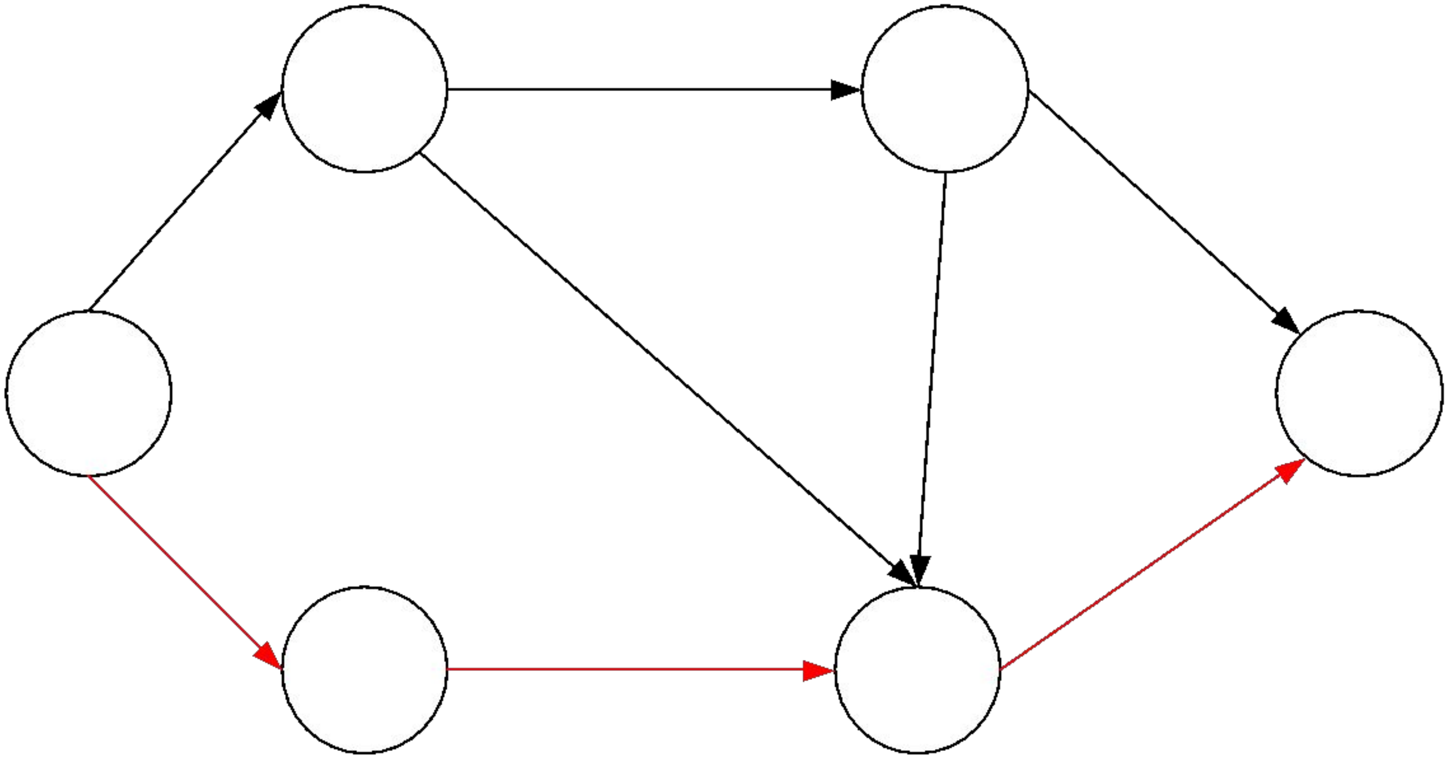
Алгоритм Форда-Фалкерсона

Шаг 1



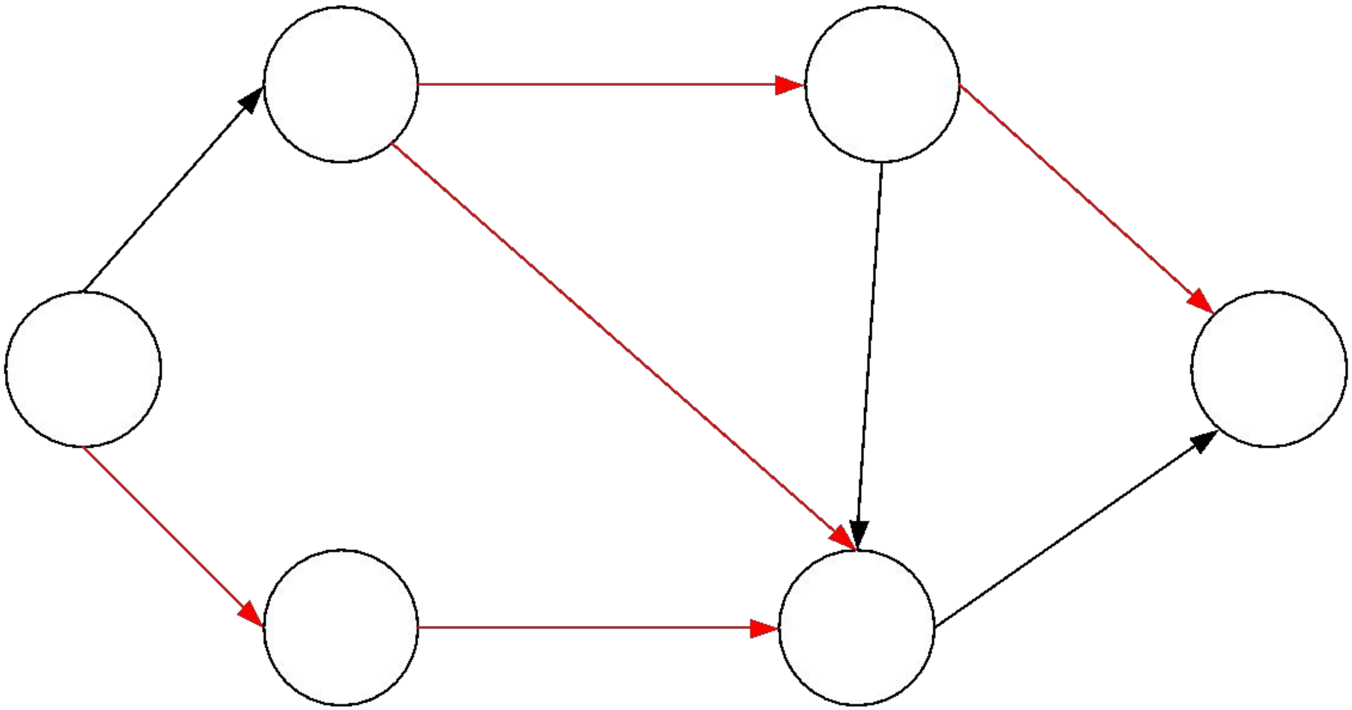
Алгоритм Форда-Фалкерсона

Шаг 2



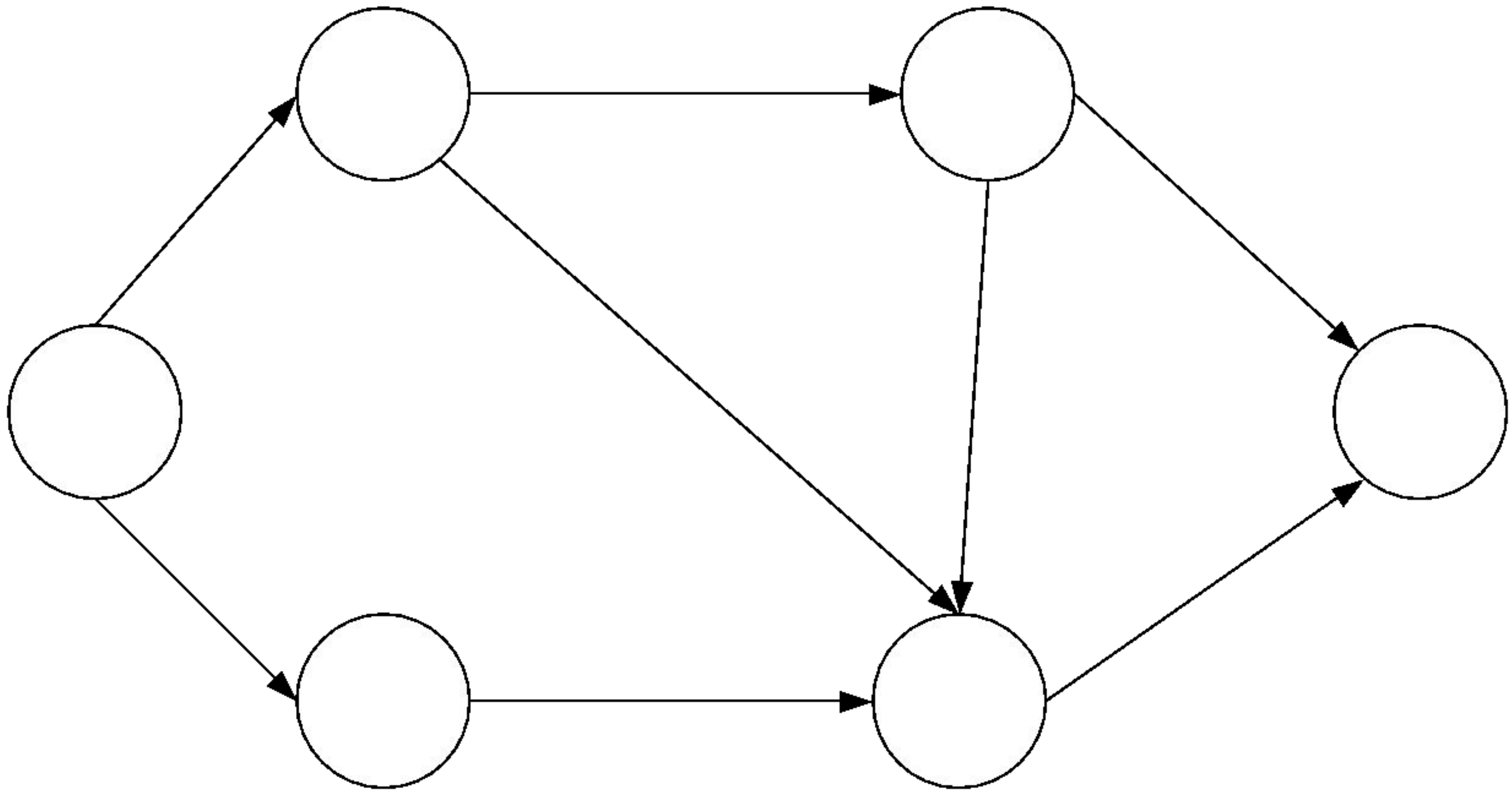
Алгоритм Форда-Фалкерсона

Шаг 3



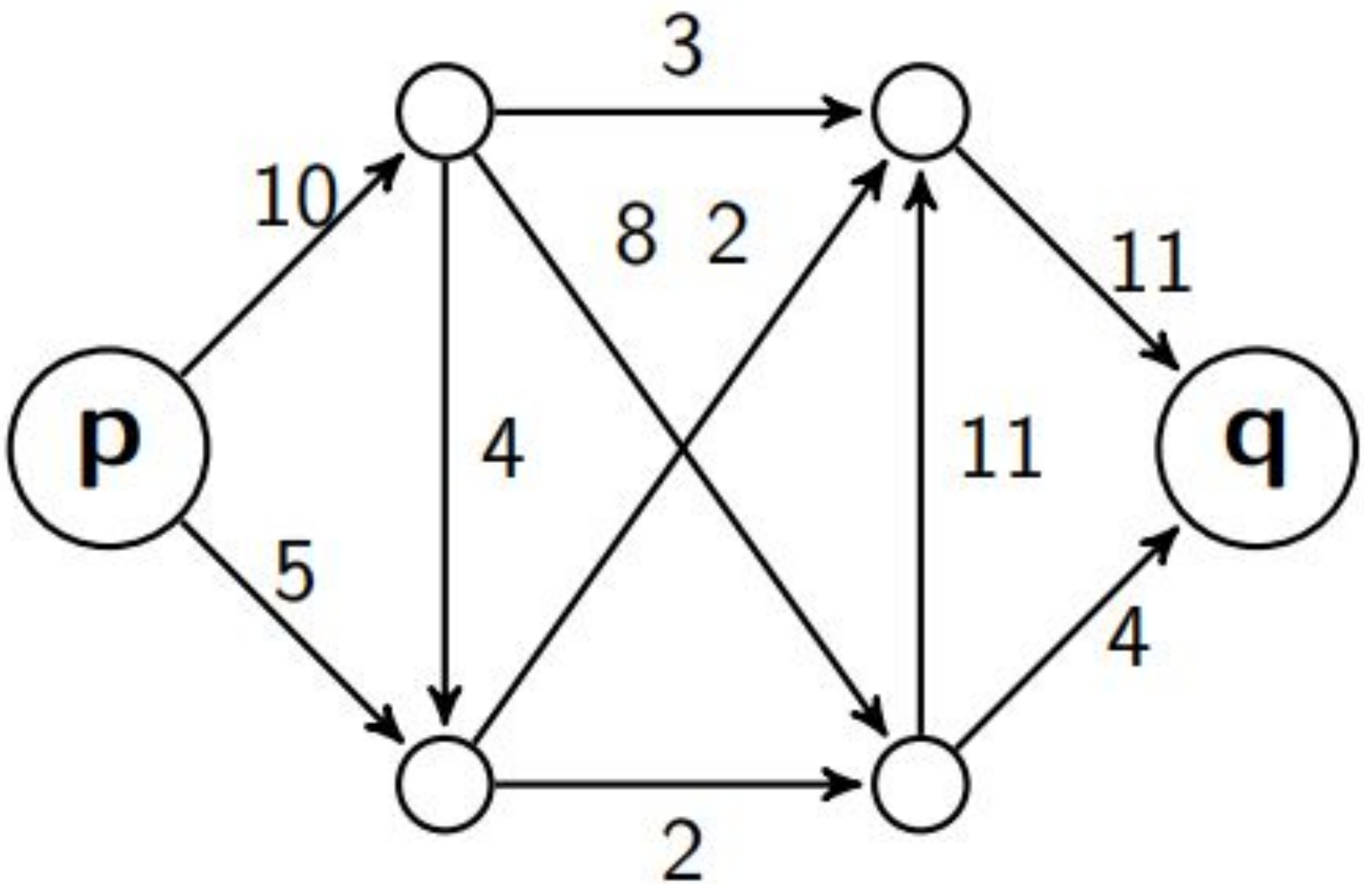
Алгоритм Форда-Фалкерсона

Шаг 4



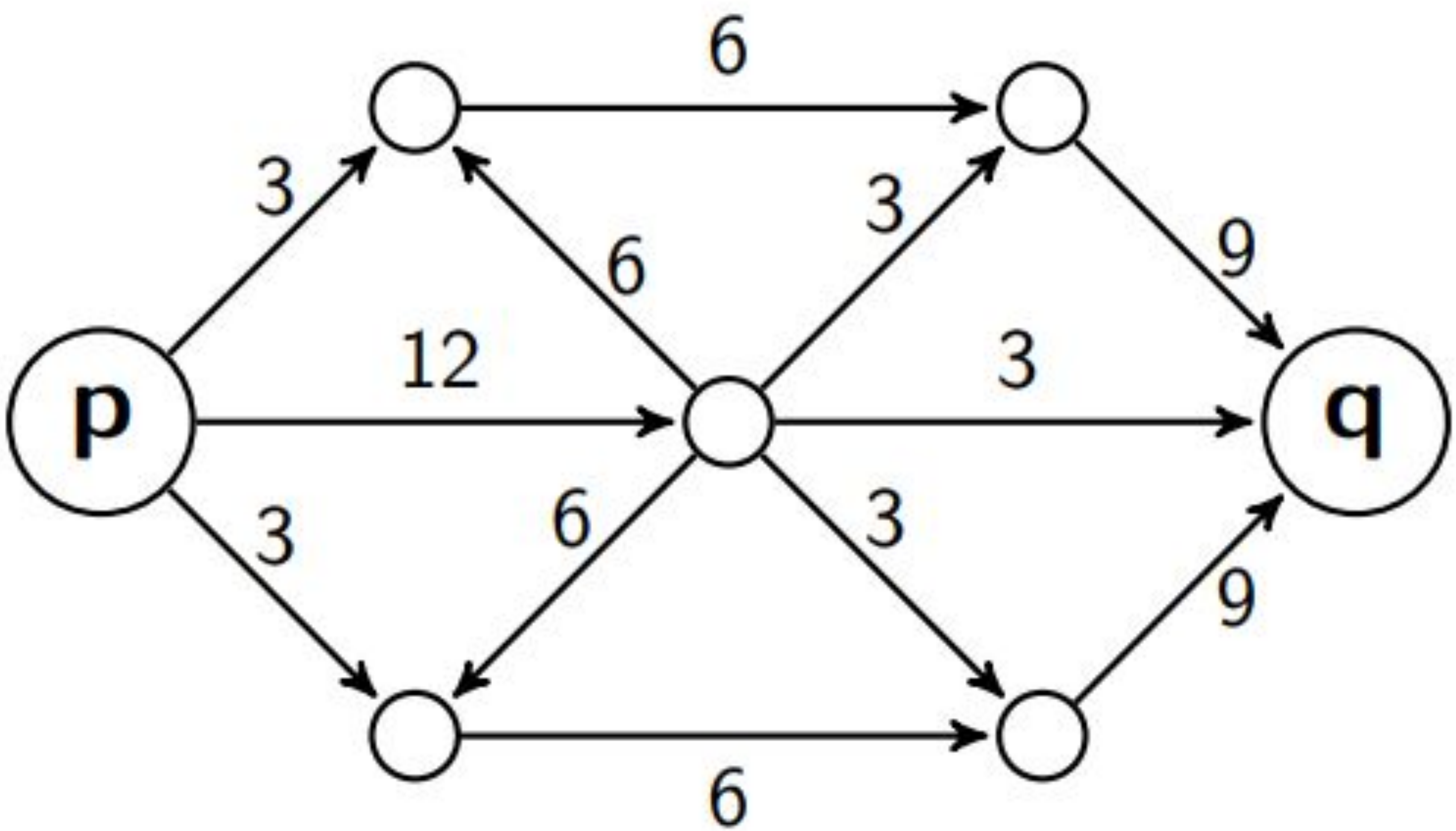
Задача 1

Найти и построить максимальный поток в транспортной сети



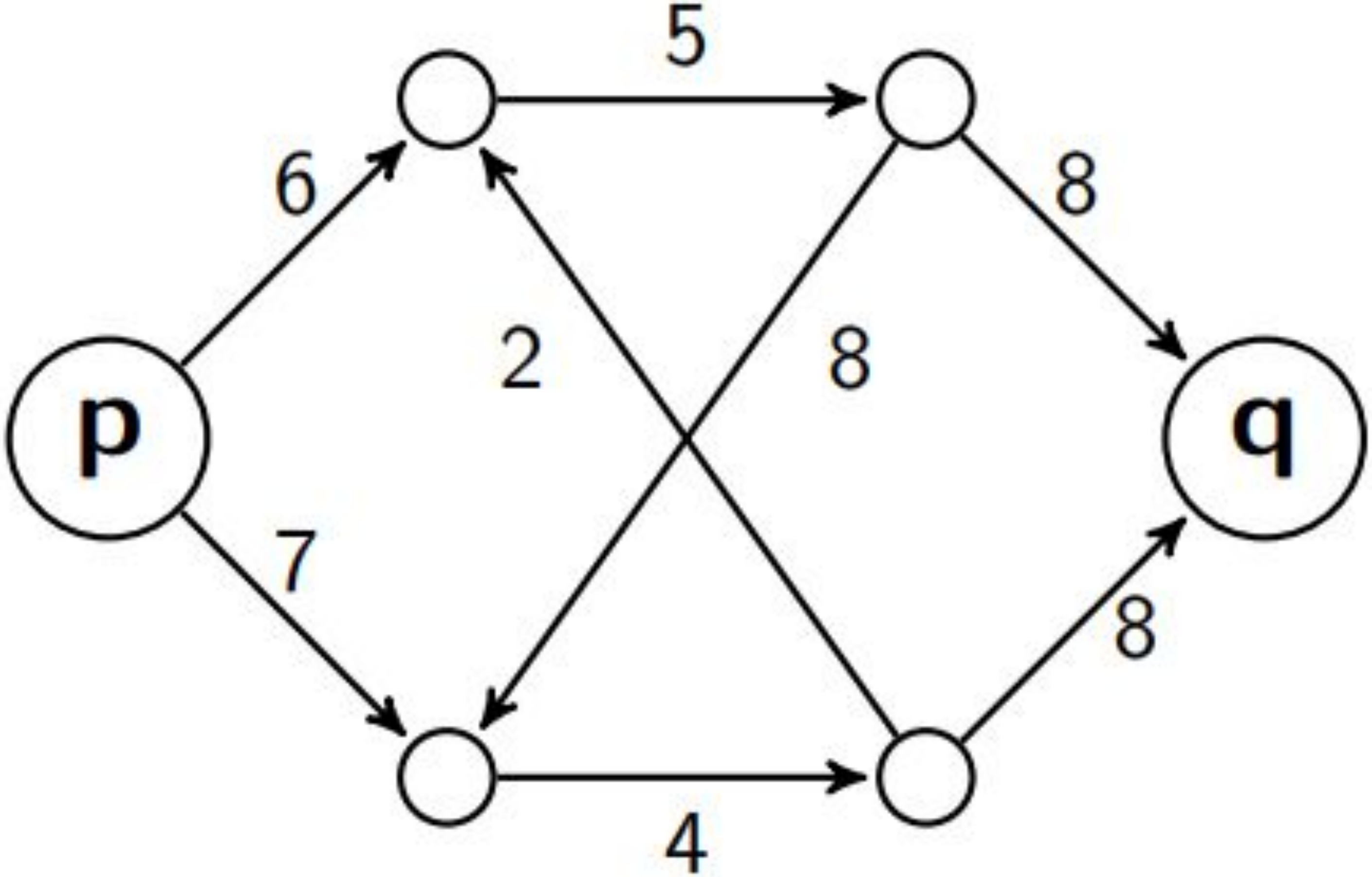
Задача 2

Найти и построить максимальный поток в транспортной сети



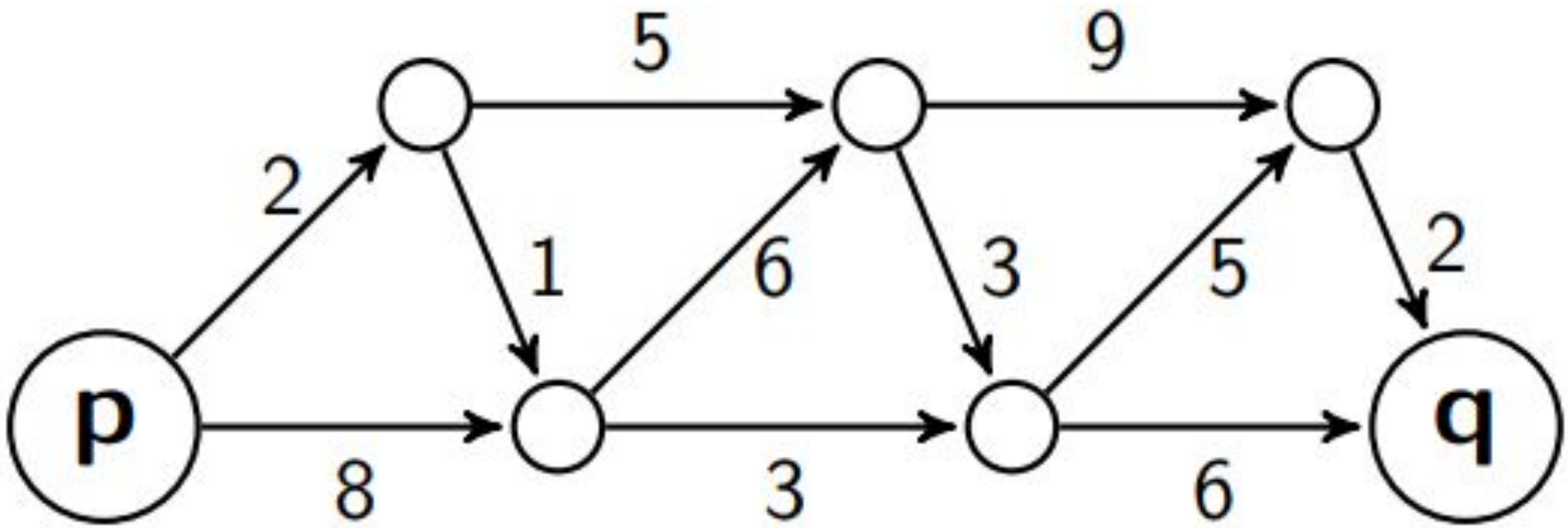
Задача 3

Найти и построить максимальный поток в транспортной сети



Задача 4

Найти и построить максимальный поток в транспортной сети



Задача 5

Найти и построить максимальный поток в транспортной сети