ИТМО Кафедра Информатики и прикладной математики

Отчет по лабораторной работе №4
«Поиск максимального потока в сети»
Вариант 7

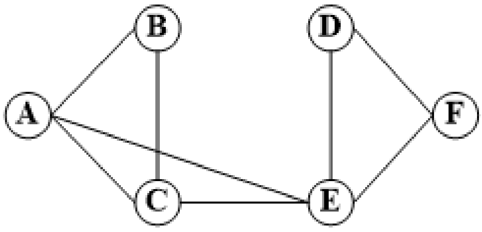
Выполнил: студент группы P3217
Плюхин Дмитрий
Преподаватель: Зинчик А. А.

2017 год

1. Задание

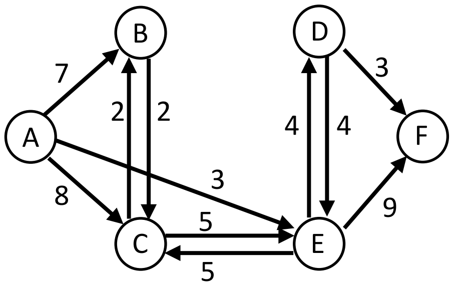
1. Самостоятельно задать пропускные способности дуг и построить максимальный поток в транспортной сети.

2. Найти минимальный разрез сети и проверить справедливость теоремы Форда – Фалкерсона.



2. Построение максимального потока в транспортной сети

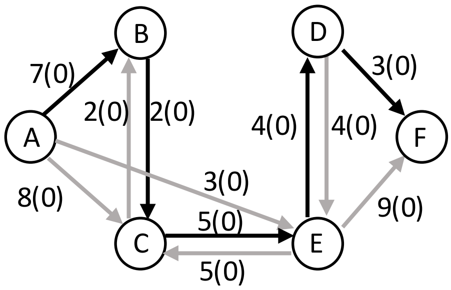
Для начала зададим пропускные способности дуг, а также выполним замену всех неориентированных дуг на пару ориентированных для промежуточных дуг и на единичные для истока и стока:

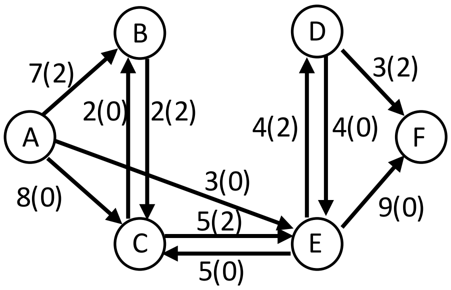


Начинаем поиск максимального потока в сети с учетом того, что изначально поток не задан (именно поток далее будет указываться в скобках рядом с пропускной способностью ребер). Выбираем увеличивающую цепь AB, BC, CE, ED, DF. Направление всех дуг совпадает с направлением потока.

$$\delta = \min(7 - 0, 2 - 0, 5 - 0, 4 - 0, 3 - 0) = 2$$

Соответственно, поток через выбранные ребра необходимо увеличить на 2.

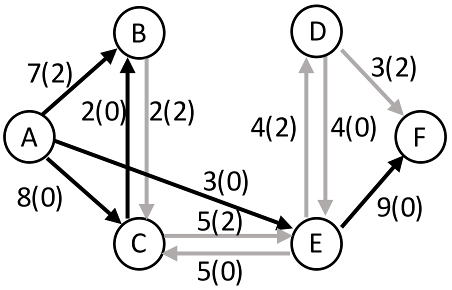


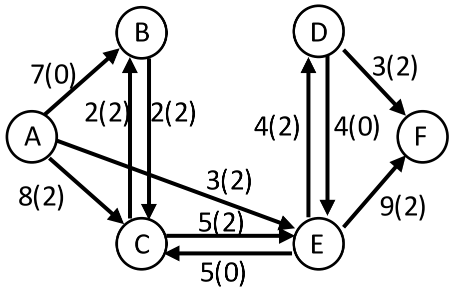


Выбираем увеличивающую цепь AC, CB, BA, AE, EF. Направление всех дуг совпадает с направлением потока за исключением дуги BA.

$$\delta = \min(8 - 0, 2 - 0, 7 - 2, 3 - 0, 9 - 0) = 2$$

Соответственно, поток через выбранные ребра необходимо увеличить на 2 (за исключением AB, поток через которую уменьшается на 2).

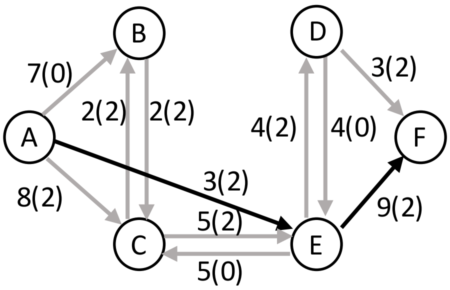


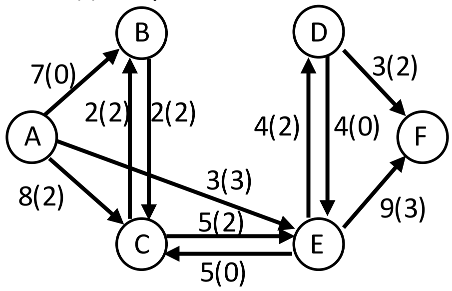


Выбираем увеличивающую цепь AE, EF. Направление всех дуг совпадает с направлением потока.

$$\delta = \min(3 - 2, 9 - 2) = 1$$

Соответственно, поток через выбранные ребра необходимо увеличить на 1.

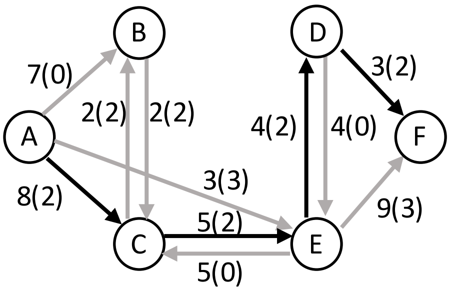


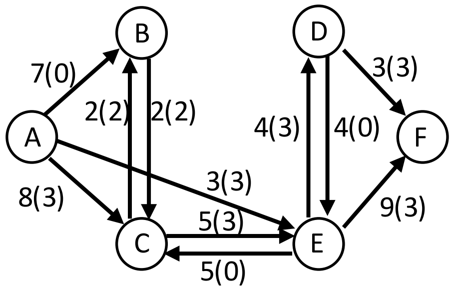


Выбираем увеличивающую цепь AC, CE, ED, DF. Направление всех дуг совпадает с направлением потока.

$$\delta = \min(8-2, 5-2, 4-2, 3-2) = 1$$

Соответственно, поток через выбранные ребра необходимо увеличить на 1.

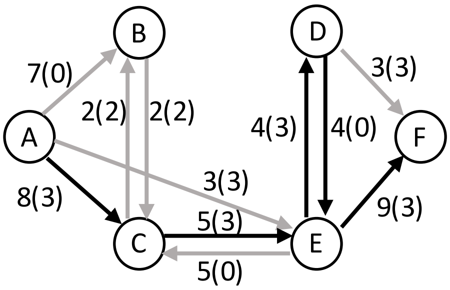


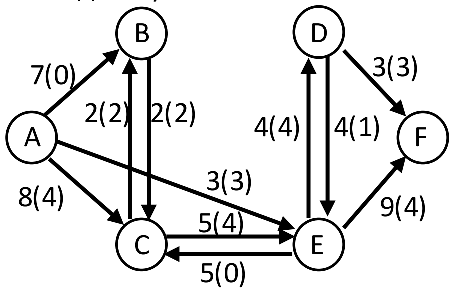


Выбираем увеличивающую цепь AC, CE, ED, DE, EF. Направление всех дуг совпадает с направлением потока.

$$\delta = \min(8-3, 5-3, 4-3, 4-0, 9-3) = 1$$

Соответственно, поток через выбранные ребра необходимо увеличить на 1.

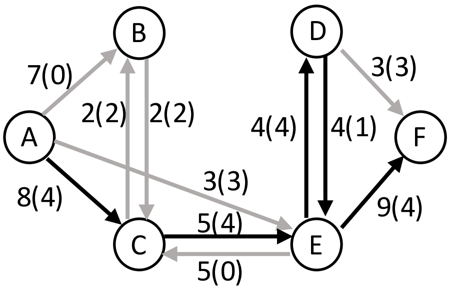


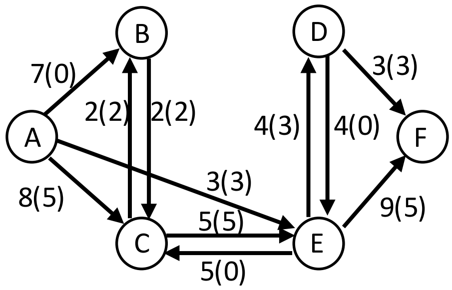


Выбираем увеличивающую цепь AC, CE, ED, DE, EF, но на этот раз дуги ED и DE выберем направленными противоположно потоку. Направление всех остальных дуг совпадает с направлением потока

$$\delta = \min(8-3, 5-3, 4-3, 4-0, 9-3) = 1$$

Соответственно, поток через выбранные ребра необходимо увеличить на 1, кроме ребер DE и ED – через них поток уменьшится на единицу .

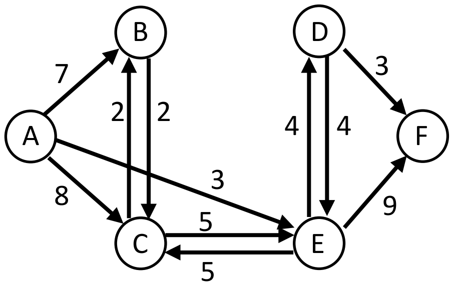




Больше увеличивающих цепей нет, потому что нельзя двигаться вдоль ребра AE (поток равен пропускной способности), вдоль ребра CE по той же причине, противоположно ребру EC (поток равен нулю). Без этих трех ребер становится невозможным достижение стока F из истока A. Следовательно, максимальный поток найден, он равен 8.

3. Проверка справедливости теоремы Форда - Фалкерсона

Для начала найдем все разрезы транспортной сети:



Эта пропуская сеть имеет 6 разрезов : {AB, AE, AC}, {AB, AE, CE}, {AC, AE, BC}, {CE, AE}, {DF, EF}, {ED, EF}. Их пропускные способности равны соответственно 18, 15, 13, 8, 12, 13. Поэтому максимальный поток в данной сети равен min(18, 15, 13, 8, 12, 13) = 8. Полученный результат характеризуется значением, совпадающим с полученным при непосредственном применении алгоритма поиска максимального потока, поэтому теорема Форда-Фалкерсона справедлива.

4. Вывод

В лабораторной работе было проведено ознакомление с двумя способами поиска максимального потока в сети – использованием алгоритма и применением теоремы Форда Флакерсона. Был сделан вывод о том, что способ, связанный с реализацией алгоритма наиболее прост в том плане, что представляет регламентированную последовательность действий, следовательно, его автоматизация может быть проведена сравнительно просто. Второй же способ, использующий поиск минимального разреза транспортной сети, может потребовать больших трудозатрат при попытке автоматизации, ведь следует обеспечить эффективный анализ всех возможных комбинаций пар вершин, однако данный метод быстрее приводит к результату в случае, когда граф не очень сложен (состоит из 2 – 5 вершин) и требуется провести анализ вручную.