УДК 004.432

Реализация метода построения максимального потока в сети на языке программирования DELPHI

Прохорова Наталья Юрьевна
Приамурский государственный университет имени Шолом-Алейхема
магистрант

Лучанинов Дмитрий Васильевич
Приамурский государственный университет им.Шолом-Алейхема
Старший преподаватель кафедры информационных систем, математики и правовой информатики

Аннотация
В исследовательской работе описаны теоретические сведения, необходимые для реализации метода построения максимального потока в сети, основанного на теории графов. Разобран алгоритм Форда-Фалкерсона и применен в программной реализации на объектно-ориентированном языке программирования Delphi, для нахождения максимального потока в сети.
Ключевые слова: максимальный поток, алгоритм Форда-Фалкерсона, сеть, пропускная способность, разрез, метки, матрица.

Implementation of the method of constructing the maximum network stream in the programming language DELPHI

Prokhorova Natalia Yurievna
Sholom-Aleichem Priamursky State University
Undergraduate

Luchaninov Dmitry Vasilyevich
Sholom-Aleichem Priamursky State University
Senior lecturer

Abstract
The research paper describes the theoretical information necessary to implement the method of constructing the maximum flow in the network based on graph theory. The Ford-Falkerson algorithm is analyzed and applied in the software implementation in the object-oriented programming language Delphi, to find the maximum flow in the network.
Keywords: maximum flow, Ford-Falkerson algorithm, network, bandwidth, cut, labels, matrix.

Задачи о потоках в сетях можно сконструировать в виде задач линейного программирования. Благодаря особой структуре потоковых задач для них было разработано множество эффективных алгоритмов, чем объясняется особенный интерес к таким задачам. Задачи о максимальном потоке заключаются в нахождении по данной сети потока максимальной величины. Подобные задачи занимают центральное место в теории сетей и применяются к различным видам практической реализации.

Многие исследователи рассматривают задачи максимального потока в различных областях жизнедеятельности. В рамках единой экономической модели Р.С. Рогулин, П.В. Нечаев, Д.Е. Плешанов [1] сделали обобщение задач линейного программирования: транспортной, учета времени и максимального потока. В докладе о вопросе расчёта потоковых моделей сетей Н.И. Прохорова [2] представила некоторые материалы по двойственным графам и их использованию для расчета максимального потока в сети. М.А. Чурсин, Л.Б. Афанасьевский, А.Н. Горин [3] разработали программу определения максимального потока в транспортной сети в среде Delphi 7. В своей статье Г.Г. Ягудаев, И.Э. Саакян, П.Г. Трегубов, С.В. Еремин, А.В. Кулаков [4] описали модель управления грузопотоками в системе ускоренных мультимодальных перевозок с использованием алгоритма поиска максимального потока в сети. Изучением максимальных потоков занимались, такие иностранные исследователи, как Т.Х. Кормен, Ч. И. Лейзерсон и др. [5], Ю.Л. Лолер [6], Пол Кристиано, Д.А. Келнер [7], описав свои труды в монографиях.

Теорема о максимальном потоке и минимальном разрезе – одна из важнейших в теории потоков в сетях. Она гласит, что для любой сети с одним источником и одним стоком величина максимального потока в графе путей будет равна величине пропускной способности его минимального разреза.

Доказательство теоремы заключается в трех пунктах:

1. Если можно найти дополняющий путь, то поток будет являться не максимальным.

2. Если дополняющий путь найти не удается, то в качестве S можно взять все вершины, которые достижимы из s в остаточной сети.

3. Величина абсолютно любого потока всегда будет меньше пропускной способности любого разреза.

Разрез – множество дуг, исключение которых из сети отделило бы некоторое множество узлов от остальной части узлов. При решении задачи о максимальном потоке будем рассматривать только разрезы, отделяющие источник от стока. Примером является сеть, в которой разрез, состоящий из дуг (2,4) и (3,4), отделяет узел 4 от группы узлов 1, 2, 3 (рис. 1) [8].

Пропускная способность (величина) разреза – сумма пропускных способностей всех дуг, входящих в разрез. Пропускная способность разреза, изображенного на рис. 1 равна $c_{24}+c_{34}$.

Поток будет считаться максимальным, при условии, что его величина максимальна среди всех потоков в данной сети.

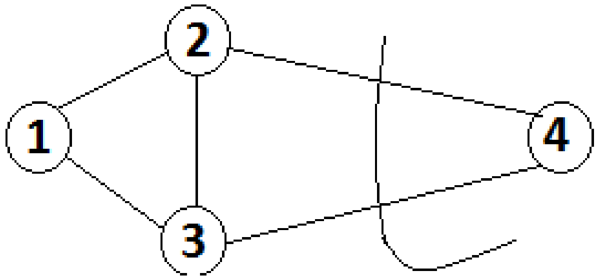

Рисунок 1 – Разрез сети

Пусть дана сеть G и поток f, существующий в ней. Остаточной пропускной способностью из u в v будет называться величина $c_f(u,v) = c(u,v) - f(u,v)$. А остаточной сетью сети G, порожденной потоком f, называется сеть $G_f = (V, E_f)$, где

$$E_f = \{(u,v) \in V \times V : c_f(u,v) > 0\}.$$

Пропускная способность цепи в сети – минимум из пропускных способностей, входящих в нее дуг.

Матрицей потоков сети G является квадратная матрица F размерности $n \times n$, где $n = |E|$, каждому элементу F_{ij} которой соответствует величина потока, протекающего по дуге (X_i, X_j) [8].

Алгоритм Форда-Фалкерсона был предложен в 1956 г. До этого времени решение задач осуществлялось с использованием методов линейного программирования, что было не результативно.

Идея алгоритма состоит в нахождении сквозных путей с положительными потоками от источника к стоку. Алгоритм начинает работу с исходного допустимого потока (может быть нулевым). После чего осуществляются попытки увеличить величину потока при помощи систематического поиска всех вероятных цепей из ***s*** в ***t***, на которых можно увеличить величину потока (дополняющие цепи).

Поиск дополняющих цепей производится путем расстановки меток, указывающих, на каких дугах и на сколько можно увеличить поток. Когда найдена одна из таковых цепей, поток вдоль нее возрастает. Затем все метки стираются, и новый полученный поток используется в качестве начального при новой расстановке меток [8].

Алгоритм прекращает работу, когда больше нельзя найти ни одну из дополняющих цепей. Последний найденный поток будет максимальным.

Этап расстановки меток является одной из главных частей алгоритма. Каждая существующая вершина в сети может принимать одно из трех состояний (рис. 2) [8].

Рисунок 2 – Состояние вершин

Входными данными алгоритма являются:

- матрица пропускных возможностей дуг;
- исходный поток, задаваемый матрицей потоков дуг.

При завершении работы алгоритм выдаст найденный максимальный поток, определяющий матрицей потоков дуг.

1 Этап. Инициализация (присвоение начального значения для переменной данных).

Присвоим все дуговые потоки равными нулю: $f_{i,j} := 0$.

2 Этап. Положим вершине s метку $L(s) = [+s, \delta(s) = \infty]$.

Теперь вершина s помечена, но не просмотрена. Все оставшиеся вершины пометок не имеют.

3 Этап. Просмотр вершин, которые помечены.

Найти любую с пометкой, но не просмотренную вершину x_i; присвоить ей метку $[\pm x_k, \delta(x_i)]$.

1. Каждой непомеченной вершине $x_j \in \Gamma x_i$, для которой $f_{i,j} < c_{i,j}$, назначить метку $[+x_i, \delta(x_j)]$, где $\delta(x_j) = \min[\delta(x_i), c_{i,j} - f_{i,j}]$.

2. Каждой непомеченной вершине $x_j \in \Gamma^{-1} x_i$, для которой $f_{j,i} > 0$ определить метку $[-x_i, \delta(x_j)]$, где $\delta(x_j) = \min[\delta(x_i), f_{j,i}]$.

Теперь вершина x_i не только помечена, но и просмотрена, а вершина x_j, имеет пометку, но не просмотрена.

4 Этап. Проверка пометки вершин.

Если на 3 Этапе какая-нибудь из вершин имеет пометку, то возможны два варианта действий:

- если пометку имеет вершина t, то переход на следующий этап;
- если пометка присвоена любой другой вершине, то возврат на предыдущий этап (3 Этап).

Если на 3 Этапе больше нет возможности присвоить никаких меток, то алгоритм завершает свою работу с некоторым наибольшим потоком.

5 Этап. Выбор вершины $x = t$.

6 Этап. Увеличение потока (зависит от вида вершины).

1. Если метка вершины x имеет вид $[+z, \delta(x)]$, то существует необходимость изменения потока вдоль дуги (z, x) с $f(z, x)$ на $f(z, x) + \delta(t)$.

2. Если метка вершины x имеет вид $[-z, \delta(x)]$, то необходимо поменять поток вдоль дуги (x, z) с $f(x, z)$, присвоив вершине $f(x, z) - \delta(t)$.

7 Этап. Проверка вершин.

Если $z = s$, абсолютно все метки стираются и делается возврат к Этапу 2. При этом используется уже увеличенный поток, полученный на Этапе 6.

Если $z \neq s$, то можно положить $x = z$ и вернуться на 6 Этап [8].

Для использования разработанной программы необходимо подготовить входные данные:

- матрица пропускной способности сети;
- массив дескрипторов планируемых работ;

Большим плюсом является, что пользователь имеет возможность сохранить результаты в численном их представлении из переменных программы в отдельные файлы с помощью команды save.

Максимальный поток в сети можно найти, принимая в качестве начального поток, тождественно равный нулю, и добавляя к нему последовательность простых потоков по цепям от источника к стоку.

Для нахождения максимального потока в сети на языке программирования Delphi выполним необходимые действия:

- заполним в программе матрицу пропускных возможностей дуг;
- зададим количество имеющихся вершин;
- укажем течение потока (из какой вершины в какую течет поток).

Найдем максимальный поток сети, состоящей из восьми вершин. Заполнив необходимые поля, нажимаем кнопку «Ок», а затем кнопку «Расчет», позволяющую вычислить максимальный поток (рис. 3).

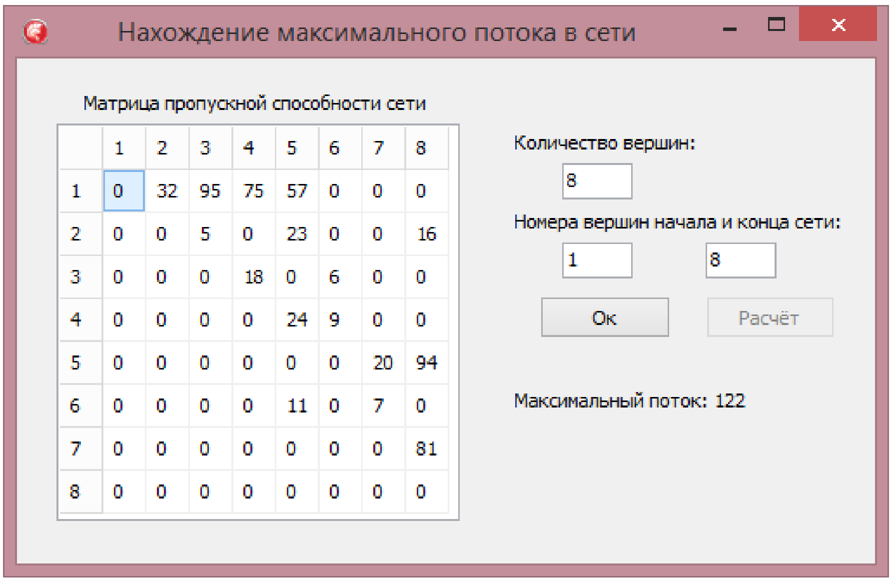

Рисунок 3 – Нахождение максимального потока сети

Программная реализация дает возможность нахождения максимального потока не только на всем промежутке цепи, но и на отдельных участках, что немаловажно (рис. 4).

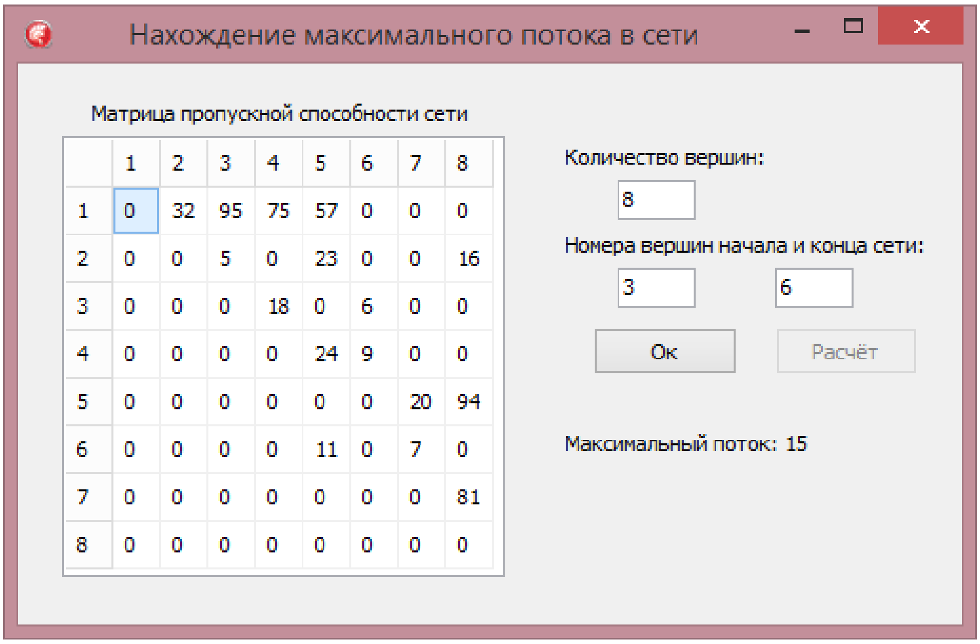

Рисунок 4 – Максимальный поток на отдельном участке сети

Созданную программу можно отнести к программному обеспечению систем, которые основаны на сетевых моделях. Разумеется, что данные модели описывают реальные масштабные ситуации и могут содержать тысячи переменных и ограничений. Но, заметим, что для этого потребуется надежное программное обеспечение [9].

В результате выполнения данной исследовательской работы был осуществлен алгоритм Форда – Фалкерсона для решение задач поиска максимального потока в графах с программной реализацией на объектно-ориентированном языке программирования Delphi. Данный алгоритм - один из наиболее эффективных методов решения задач, позволяющих найти точное решение.

В целом, метод нахождения максимального потока в сети позволяет получить результаты, для дальнейшего использования при решении последующих задач управления системой распределения.

Библиографический список

1. Рогулин Р.С., Нечаев П.В., Плешанов Д.Е. Обобщение задач транспортной, учета времени, максимального потока в рамках единой экономической модели // Экономика и предпринимательство. 2018. № 9 (98). С. 813-816.
2. Прохорова Н.И. К вопросу расчёта потоковых моделей сетей // В сборнике: современные проблемы телекоммуникаций. Материалы конференции. 2016. С. 65-68.

3. Чурсин М.А., Афанасьевский Л.Б., Горин А.Н. Разработка программы определения максимального потока в транспортной сети // В сборнике: Общество и экономическая мысль в XXI в.: пути развития и инновации Материалы V Международной научно-практической конференции. Российский экономический университет имени Г.В. Плеханова, Воронежский филиал. 2017. С. 562-566.
4. Ягудаев Г.Г., Саакян И.Э., Трегубов П.Г., Еремин С.В., Кулаков А.В. Модель мультимодальных перевозок на основе управляемых сетей // Наука и бизнес: пути развития. 2017. № 9 (75). С. 20-25.
5. Cormen T.H., Leiserson C.E., Rivest R.L., Stein C. Introduction to algorithms, Second Edition М.: Вильямс, 2012. 1296 с.
6. Lawler E. Combinatorial Optimization: Networks and Matroids // Dover Books on Mathematics M.: Dover Publications, 2011. С. 384.
7. Christiano P., Kelner J.A. Electrical Flows, Laplacian Systems, and Faster Approximation of Maximum Flow in Undirected Graphs // Труды 43-го ежегодного симпозиума ACM по теории вычислений, STOC '11, ACM Press, 2011. С. 273.
8. Оптимизация плана регламентных работ по критерию максимума среднего потока в сети применительно к задаче транспортировки нефти по магистральному нефтепроводу. URL: https://pandia.ru/text/78/161/10122.php (дата обращения: 23.01.19).
9. Черников А. В. Решение задачи о максимальном потоке в сети. URL: http://rpp.nashaucheba.ru/docs/index-8708.html (дата обращения: 23.01.19).