Задача о максимальном потоке

Станкевич А.*

1 Введение

Задачу о максимальном потоке в сети можно проиллюстрировать следующим образом: есть водопроводная сеть, состоящая из труб разной пропускной способности (которая может измеряться, например, максимально возможным количеством пропущенной жидкости в единицу времени), соединенных узлами. В одном из узлов находится источник воды (или какой-либо другой жидкости), в другом – сток. Задача о максимальном потоке заключается в определении максимального количества жидкости, которое может быть выпущено из источника и, пройдя через сеть труб (естественно, не превышая пропускной способности этих труб), благополучно удалиться в стоке.

Таким образом, эта задача имеет большую практическую ценность. Она моделируется в теории графов и имеет ряд интересных решающих ее алгоритмов. Впервые она была решена общим методом линейного программирования. После этого Форд и Фалкерсон разработали метод, предназначенный специально для этой задачи. При реализации и оптимизации этого метода появился ряд эффективных алгоритмов, рассмотрению которых и посвящен доклад.

Раздел 2 доклада посвящен основным определениям, в разделе 3 излагается метод Форда-Фалкерсона, разделы 4 и 5 рассмотрены две реализации этого метода, раздел 6 содержит алгоритм масштарбирования пропускной способности, в котором метод Форда-Фалкерсона взят за основу.

2 Определения и простейшие свойства потока

Определение 2.1. Пусть дан ориентированный граф $G = (V, E)$, где V – множество вершин, E – множество ребер. Ребром $(u, v) \in E$ мы на называем ребро с началом $u \in V$ и концом $v \in V$. Мы будем рассматривать только простые графы, то есть не содержащие петель (ребер вида $(v, v), v \in V$) и двойных ребер (если у двух ребер совпадают начала и концы, то это – одно и то же ребро). Пусть дано отображение $c : E \to \mathbb{R}^+$ – пропускная способность ребер. Можно индуцировать это отображение до $c : V \times V \to \mathbb{R}^+$

*Законспектировал лекцию Харчев Н.

следующим образом: $c(u,v) = c((u,v)), \text{если}(u,v) \in E, c(u,v) = 0, \text{иначе}$, что также понятно – пропускная способность отсутствующего ребра равна нулю. Также выделены две вершины в G $s,t \in V, s \neq t$. s – это исток сети, t – сток. Четверка (G,c,s,t) и называется сетью.

Буквы G,V,E,c,s,t мы фиксируем для использования в дальнейшем в том же смысле, что и в определении. Также буквами V,E мы будем пользоваться для обозначения количества элементов в соответствующих множествах.

Определение 2.2. Пусть дана сеть. $f : E \to \mathbb{R}$ или, с аналогичным распространением $f : V \times V \to \mathbb{R}$ называется потоком в этой сети, если удовлетворяет следующим свойствам

$$(1) f(u,v) \leq c(u,v), \forall u,v \in V,$$

$$(2) f(u,v) = -f(v,u), \forall u,v \in V,$$

$$(3) \forall u \in V \setminus \{s,t\}, \sum_{v \in V} f(u,v) = 0.$$

Эти три свойства имеют ясный смысл. (1) – поток не должен превышать пропускную способность сети, (2) – если мы направили поток из u в v, то обратно течет такой же по модулю, но отрицательный по величине поток. (3) – сумма всего, что втекло в вершину (отрицательные компоненты суммы) должно совпасть с суммой утекшего, что, естественно, не относится к истоку и стоку.

Определение 2.3. Пусть дана сеть и поток в ней. $|f| := \sum_{v \in V} f(s,v)$ – величина потока.

С величиной потока связано дополнительное условие на определение потока: $|f| \geq 0$, чтобы исток ненароком не оказался стоком.

Задача 2.1 (о максимальном потоке). Дана сеть. Найти поток f с масимальным $|f|$, который мы и называем максимальным.

Утверждение 2.1. *Во всякой сети существует максимальный поток.*

Доказательство. Сузим отображение f с пар вершин на ребра, упорядочим ребра. По потоку f мы можем построить точку $f' \in \mathbb{R}^E$, где $f'_i = f(e_i), e_i$ – i -ый элемент E. Рассмотрим множество $A := \{f' \in \mathbb{R}^E | f\text{– поток}\}$. Это множество непусто ($0 \in A$), замкнуто("предел потоков – поток") и ограничено пропускными способностями. $|f|$ – непрерывная функция (конечная сумма координат), значит, по теореме Вейерштрасса на A она достигает максимума. Таким образом мы, очевидно, получаем максимальный поток. □

Обозначение 2.1. Пусть $X,Y \subseteq V$.

$$f(X,Y) := \sum_{x \in X, y \in Y} f(x,y).$$

Утверждение 2.2. *$|f| = \sum_{v \in V} f(v,t)$, или, что то же самое $f(s,V) = f(V,t)$.*

Доказательство. По свойству потока (2), $0 = f(V,V) = f(s,V) + f(t,V) + f(V^1,V)$, где $V^1 = V \setminus \{s,t\}$. По 3-ему свойству $f(V^1,V) = 0$, а по 2-му $f(t,V) = -f(V,t)$. То есть, $0 = f(s,V) - f(V,t)$. □

Определение 2.4. $V = S \sqcup T$ – разбиение V на S и T, причем такое, что $s \in S, t \in T$, называется s-t-разрезом (или просто разрезом).

Утверждение 2.3. *Пусть (S,T) – s-t-разрез. Тогда $f(S,T) = |f|$.*

Доказательство.

$$f(S,V) = f(S,S) + f(S,T) = f(S,T).$$

С другой стороны, если $S^1 := S \setminus s$, то

$$f(S,V) = f(s,V) + f(S^1,V) = f(s,V) = |f|.$$

Проверить эти равенства – легкое упражнение на знание свойств потока. □

Утверждение 2.4. *$\forall$ потока f $\forall$ разреза (S,T) $|f| \le c(S,T)$. $c(S,T)$ определяется аналогично $f(S,T)$.*

Утверждение очевидно следует из предыдущего.

3 Метод Форда-Фалкерсона

Определение 3.1. Остаточная сеть. Вычитая из сети поток f, мы получаем остаточную сеть: (G_f, c_f, s, t). Множество вершин остаточной сети остается прежним, $c_f(u,v) := c(u,v) - f(u,v)$. Заметим, что это отображение действует, как и положено в $\mathbb{R}^+$. Множество ребер остаточной сети меняется следующим образом: ребра $(u,v) \in E$ такие, что $c_f(u,v) = 0$ уже не принадлежат E_f, а если $c_f(u,v) > 0 \Rightarrow (u,v) \in E_f$, даже если это ребро не принадлежало E.

Утверждение 3.1. *Если в остаточной сети G_f $\exists$ путь $p : s \to t$ (так мы обозначаем путь, который начинается в s и заканчивается в t) $\Longrightarrow f$ не максимален.*

Теорема 3.1 (Форда-Фалкерсона). *Если в G_f не $\exists p : s \to t \Longrightarrow f$ максимален.*

Доказательство. Построим по f такой объект, как канонический разрез – это разрез (S,T), такой, что $S := \{u \in V | \exists p : s \to u \in G_f\}, T := V \setminus S \Rightarrow t \in T$. Докажем, что $c(S,T) = |f|$ (из этого, по утверждению 2.4 сразу следует, что f – максимален). $c(S,T) = |f| \Leftrightarrow 0 = c_f(S,T)$. $c(S,T) = \sum_{u \in S, v \in T} c_f(u,v)$. Но $c_f(u,v) = 0$ по определению (S,T), иначе бы $v \in S$. □

Упражнение 3.1. Доказать, что для любой сети канонический разрез единственен, то есть не зависит от потока.

Алгоритм 3.1 (Форда-Фалкерсона). Шаг 0. Берем базовый поток f_0– нулевой.

Цикл. i-ый шаг цикла. Рассмотрим остаточную сеть $G_{f_{i-1}}$.

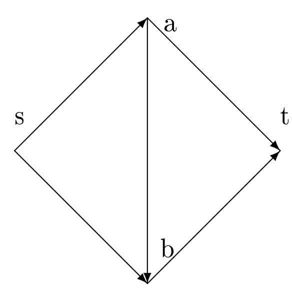
Случай 1 Если в ней существует дополняющий путь (путь $p : s \to t$), строим поток f' – пропускаем через этот путь столько, какова пропускная способность минимального входящего в него ребра (ребер конечное число, пропускная способность каждой больше 0, значит поток ненулевой). $G_{f_i} = (G_{f_{i-1}})_{f'} = G_{f_{i-1}+f'}$, тогда $f_i := f_{i-1} + f'$. Мы увеличили поток.

Случай 2 Дополняющего пути не существовало. Значит, по теореме Форда-Фалкерсона, максимальный поток построен, алгоритм заканчивает работу.

Конец цикла.

В этом методе не детерминирован выбор очередного дополняющего пути и, как мы увидим, это влечет некоторые печальные следствия.

У этого метода есть важное достоинство: он был первым, использующим понятие дополняющей сети. Польза этого понятия заключается в том, что, увеличивая поток, мы добавляем к нему ребра, которые идут в противоположную сторону тем, по которым был пропущен поток, благодаря чему не происходит блокировки.



Пример 3.1.

Пусть на этом графе заданы следующие пропускные способности:

$$c(s, a) = c(a, t) = c(s, b) = c(b, t) = 10^{10}, c(a, b) = 1.$$

Допустим алгоритм выбирает следующие дополняющие пути:

$$(s, a, b, t), (s, b, a, t), (s, a, b, t), (s, b, a, t), ...$$

Получается, как и было обещано, что самое слабое звено – ребро (a, b) не блокируется, а честно переворачивается каждый раз для построения нового дополняющего пути. С другой стороны, всего этих путей будет $2 \cdot 10^{10}$ – не самая маленькая сложность. Очевидно, в этом примере сложность можно легко уменьшить за счет выбора других путей. Как будет показано в разделах 4 и 5, улучшение возможно в общем случае.

4 Градиентная модификация

Определение 4.1. Пусть $u \in V$. $d[u]$ – это максимальная пропускная способность простого пути $p : s \to u$ (пропускная способность пути – это наименьшая пропускная способность его ребра).

Утверждение 4.1.
$$d[t] \geq \frac{|f_{max}|}{E}$$

Доказательство. Пусть это не так. Выкинув дуги (u, v) такие, что $c(u, v) < \frac{|f_{max}|}{E}$ получим несвязный граф. Пусть S – компонента связности s, $T := V \setminus S, t \in T$. (S, T) – разрез, значит $c(S, T) \geq |f_{max}|$. Но, по нашему предположению $c(S, T) < \frac{|f_{max}|}{E} \cdot E = f_{max}$. Получившееся противоречие доказывает утверждение. □

Путь $p : s \to t$, реализующий $d[t]$ можно найти с помощью алгоритма Дейкстры за $O(V^2)$. Применим теперь метою Форда-Фалкерсона, дополняющими путями в котором будут эти пути. Это и есть градиентный метод. Он сходится с экспоненчиальной скоростью от количества построенных путей. Действительно, если f'_i – пропускная способность i-го пути, $f_i := f'_1 + f'_2 + \ldots + f'_i, f_{max-i} := f_{max} - f_i$,то на i-ом шаге:

$$||f_{max-i}| - |f'_{i+1}|| \leq |f_{max-i}| - \frac{|f_{max-i}|}{E} = (1 - 1/E) \cdot |f_{max-i}| =$$

$$= (1 - 1/E) \cdot |f_{max-(i-1)} - f'_i| \leq \ldots \leq (1 - 1/E)^{i+1} \cdot |f_{max}|$$

Таким образом получаем сложность $O(V^2 \cdot \log_{1-1/E} |f_{max}|)$.

5 Алгоритм Эдмондса-Карпа

Алгоритм Эдмондса-Карпа заключается в том, чтобы в методе Форда-Фалкерсона в качестве дополняющего пути всякий раз выбирать кратчайший (по количеству ребер) путь. Его можно найти с помощью поиска в ширину за время $O(E)$.

Определение 5.1. Пусть $u \in V$. $d_f[u]$ – это расстояние (в количестве ребер) от s до u в остаточной сети f.

Лемма 1. *$d_{f_1}[u] \geq d_f[u]$, где f_1 получается из f пропусканием потока f'_1 по кратчайшему пути.*

Утверждение 5.1. *Максимальное количество итераций в алгоритме Эдмондса-Карпа – VE.*

Доказательство. Введем понятие критического ребра – это то ребро, которое (или обратное к которому) на данном шаге удаляется. Возьмем произвольное $(u, v) \in E$, докажем, что оно может быть критическим не более V

раз. Пусть $f_1, f_2, ...$ – итерации нашего алгортма на которых это ребро стало критическим, $f_1', ...$ – соответствующие дополняющие пути. Пусть $d_{f_1}[u] = d, d_{f_1}[v] = d+1$ (не умаляя общности, так как $(u,v) \in f_1'$). Тогда на шаге f_2' окажется, что (v,u) принадлежит кратчайшему пути; но по лемме $d_{f_2}[v] \geq d_{f_1}[v] = d+1$, а $d_{f_2}[u] = d_{f_2}[v]+1$, значит $d_{f_2}[u] \geq d+2$. Продолжая так, и зная, что $d_f[u] \leq V \ \forall u, f$, получаем требуемую оценку. Таким образом, все критические ребра графа можно оценить сверху числом VE. В то же время понятно, что на каждой итерации алгоритма мы получаем как минимум одно критическое ребро. □

Мы получили оценку на алгоритм: $O(E^2V)$.

6 Алгоритм масштабирования

Этот алгоритм работает в предположении, что все пропускные способности целые. Пусть U – максимальная пропускная способность. Тогда запишем пропускную способность каждого ребра в двоичной записи (для каждого ребра отведем $[\log U] + 1 =: n+1$). Занумеруем биты с младшего (0-ой) по старший (n-ный). Тогда

$$c(u,v) = a_n(u,v)2^n + ... + a_2(u,v)2 + a_1(u,v),\ a_i(u,v) \in \{0,1\}.$$

Будем решать задачу методом Форда-Фалкерсона сначала для графа с урезанными пропускными способностями $c_0(u,v) := a_n(u,v)$. Решив ее, получив поток f_0, добавляем следующий бит, и вычитаем "грубое приближение". То есть $c_1(u,v) := a_n(u,v)2 + a_{n-1}(u,v) - f_0(u,v)2$. Действуя так дальше, наше грубое приближение становится точнее и точнее, пока не станет решением для исходной задачи. Сложность этого алгоритма $O(E^2 \log U)$. Здесь $\log U$ –количество итераций. Докажем, что сложность каждой итерации $O(E^2)$.

На первой итерации мы имеем ребра веса только 1. Это значит, что $|f| \leq V$. Значит количество итераций (дополняющих путей) не превосходит V, поиск доп. пути занимет $O(E)$. Получаем сложность $O(VE) \leq O(E^2)$.

Теперь рассмотрим переход ко второй итерации (по аналогии получается оценка на последующие итерации). Граф G_{f_0} несвязен. Рассмотрим компоненту связности s – S, $T := V \setminus S$, $t \in T$. $\Rightarrow$ $(c_0)_{f_0}(S,T) = 0$. Значит в новом графе с пропускными спосбностями c_1, $c_1(u,v) \leq 1 \forall u \in S, v \in T$. Так как (S,T) – разрез, то $|f_1'| = f_1'(S,T) \leq c(S,T) \leq E$. (Здесь f_1' – максимальный поток в G_1 с пропускными способностями c_1, а $f_1 = f_0 + f_1'$). Так как пропускная способность каждого дополняющего пути не меньше 1, мы получили оценку на количесво итераций. Дополняющий путь же мы можем найти за $O(E)$.