УДК 519.17

И.В.Козлов
Московский физико-технический институт
E-mail: volokno@inbox.ru

УСТОЙЧИВОСТЬ В ЗАДАЧАХ ПОИСКА МИНИМАЛЬНОГО РАЗРЕЗА И МАКСИМАЛЬНОГО ПОТОКА

Резюме

Задача комбинаторной оптимизации называется устойчивой, если ее решение сохраняется при возмущении входных параметров, не превышающих некоторого порогового значения — радиуса устойчивости. В предположении об устойчивости входа возможно построение точных полиномиальных алгоритмов для некоторых NP-трудных задач о разрезах.

В настоящей работе показано как строить алгоритмы с малой трудоемкостью для достаточно устойчивых полиномиальных задач. Рассмотрены задачи поиска минимального разреза в ориентированном графе и в сети и задача поиска максимального потока в сети. Также построены полиномиальные алгоритмы вычисления радиуса устойчивости данных задач.

1. Постановка задачи. Обзор существующих подходов

Исследование устойчивости в задачах дискретной оптимизации имеет большую историю, т.к. если вход задачи является устойчивым, то ее решение совпадает с решением всех задач, входы которых незначительно отличаются от исходного. В данной работе будет рассмотрена устойчивость в задаче поиска минимального разреза во взвешенных ориентированных графах и задаче поиска максимального потока в сети. Сформулируем вначале данные задачи, а затем дадим формальное определение устойчивости.

Задача поиска минимального разреза. (MINCUT) Задан граф $G = (V, E)$, $|V| = n$, $|E| = m$ и функция весов $w\colon V \times V \to \mathbb{R}^+$, $w(x, x) = 0$ (если граф неориентированный, то w — симметрическая). Требуется найти разрез $(S, \overline{S})$, $S \subset V$, для которого пропускная способность разреза $f(S) = \sum_{a \in S, b \in \overline{S}} w(a, b)$ минимальна. В случае неориентированного графа пропускная способность называется также весом разреза.

В этом определении разрез задается множеством вершин S, но далее в тексте будет встречаться понятие разреза как множества ребер между S и $\overline{S}$. Эти определения разреза эквивалентны и будут употребляться в тексте в зависимости от контекста. Задавая функцию весов не на множестве ребер E, а на всей матрице $V \times V$ мы получаем возможность задавать весь вход задачи через функцию весов w. Если какие-то 2 вершины u и v в графе $G = (V, E)$ не соединены ребром, то можно считать $w(u, v) = 0$.

Если в задаче MINCUT добавить дополнительное условие, что минимальный разрез в графе должен разделять две заданные вершины s и t, то такую задачу будем называть задачей поиска минимального разреза в сети или задачей поиска минимального s-t разреза. Задача поиска максимального потока в сети (MAXFLOW) является двойственной к задаче поиска минимального s-t разреза и заключается в том, чтобы найти величину максимального потока, который можно направить в графе из истока s в сток t. Величина максимального потока в сети оказывается равна минимальной пропускной способности среди всех s-t разрезов.

Данная двойственность впервые была доказана в работе Форда и Фалкерсона [4], где также был приведен полиномиальный по входу алгоритм решения задачи MAXFLOW. Для того чтобы найти минимальный разрез в графе без дополнительных условий, можно построить s-t разрезы для всех возможных пар s и t и решить задачу MINCUT за время $O(n^2 \cdot F)$, где F — время нахождения максимального потока в сети. В 1961 году Гомори и Ху в работе [5] представили алгоритм, находящий все возможные s-t разрезы за время $O(n \cdot F)$. В работе Карзанова и Тимофеева [6] все минимальные разрезы находятся за время $O(\lambda \cdot n^2)$, где λ — величина минимального разреза. Позднее Хао и Орлин [7] показали как произвести $n-1$ вычисление максимального потока в графе, необходимое для нахождения минимального разреза, за время, сравнимое с вычислением одного максимального потока, т. е. решить задачу MINCUT за время $O(F)$.

Параллельно с этим были разработаны методы, не опирающиеся на решение задачи MAXFLOW и позволяющие решать задачу MINCUT быстрее, чем за $O(F)$. Одним из наиболее перспективных подходов оказалось использование процедуры слияния вершин. Впервые она была использован в работе Поддерюгина [8], где минимальный разрез в графе находится за время $O(mn)$. Позднее данный метод получил развитие в работах Каргера [9], где представлен вероятностный алгоритм, который находит точное решение задачи MINCUT с высокой вероятностью за время $O(n^2 \log^3 n)$, и Штор-Вагнера [10], в которой изложен детерминированный алгоритм с трудоемкостью $O(n(m + n\log n))$.

Для решения задачи MAXFLOW на сегодня наименьшей трудоемкостью обладает комбинация алгоритмов из работ [11, 12]. Первый из них используется, когда число ребер сравнимо с числом вершин в графе, а второй — когда число ребер в графе достаточно велико. Тогда трудоемкость поиска максимального потока в сети получится равной $O(nm)$.

В данной работе мы будем рассматривать данную задачу с точки зрения устойчивости. Неформально, задача называется устойчивой, если оптимальное решение сохраняется при некотором незначительном изменении входных данных, в нашем случае - весов ребер графа. Это важное свойство, т.к. в случае устойчивого входа, решив исходную задачу мы получаем решение для целого семейства входных данных. Кроме того, оказывается что некоторые задачи в предположении об устойчивости входа могут быть решены быстрее, чем в общем случае. Дадим теперь строгое определение устойчивости.

Вначале сформулируем несколько обозначений, которые будут использоваться в дальнейшей работе. Для пары непересекающихся подмножеств вершин $A, B \subset V$ обозначим множество ребер между ними как $E(A,B) = \{ab \mid a \in A, b \in B\}$ и суммарный вес ребер этого множества как $w(A,B) = \sum_{uv \in E(A,B)} w(u,v)$. Также обозначим суммарный вес ребер, исходящих из множества A, как $\tau_w(A) = \tau(A) = w(A, \overline{A})$. Пусть зафиксирован разрез $(S, \overline{S})$, тогда через $\xi(A, S)$ обозначим суммарный вес ребер разреза, исходящих из множества A, т. е. $\xi(A, S) = \sum_{uv \in E(A,\overline{A}) \cap E(S,\overline{S})} w(u, v)$, а через $\iota(A, S) = \tau(A) - \xi(A, S)$ — суммарный вес ребер, исходящих из A и не лежащих в разрезе. Для вершины $A = \{v\}$ и ребра $e = uv$ будем писать $\tau(v)$ и $\tau(e)$, соответственно. Если минимальный разрез единственный, будем писать вместо $\xi(A, S)$ и $\iota(A, S)$ просто $\xi(A)$ и $\iota(A)$.

Определение. Пусть $w\colon V \times V \to \mathbb{R}^+$ вход задачи и $\gamma \geq 1$. Вход $w'\colon V \times V \to \mathbb{R}^+$ назовем γ-возмущением w, если $\forall u, v \in V$, $w(u, v) \leq w'(u, v) \leq \gamma w(u, v)$. Вход w назовем γ-устойчивым, если существует разрез, остающийся минимальным при всех γ-возмущениях w' входа w.

В задаче MAXFLOW введем определение устойчивости, основанное на её двойственности с задачей поиска минимального разреза в сети.

Определение. Вход w задачи MAXFLOW назовем γ-устойчивым, если минимальный разрез, порождаемый максимальным потоком, остается неизменным при всех γ-возмущениях w' входа w.

Введем теперь понятие радиуса устойчивости задачи дискретной оптимизации.

Определение. Радиусом устойчивости задачи дискретной оптимизации называется максимальное $\gamma \geq 1$, при котором вход задачи является γ-устойчивым.

Это определение радиуса устойчивости было сформулировано в недавних работах [1, 2] и несколько отличается от более общего определения, предложенного Леонтьевым [14, 15], где вход w_0 является γ-устойчивым, если при всех γ-возмущениях не возникает новых оптимальных решений. Отличия возникают когда в задаче есть несколько оптимальных решений. Пусть в задаче MINCUT вход w_0 таков, что имеется несколько оптимальных решений, тогда мы можем для каждой пары минимальных разрезов увеличить вес ребер, принадлежащих одному из них, но не принадлежащих другому. Таким образом, минимальным останется только один из двух разрезов, и радиус устойчивости задачи для входа w_0 равен 1. По Леонтьеву же радиус устойчивости при таком входе и возмущении может быть и больше единицы.

Подходы, описанные в работах [1, 2], позволяют, в случае когда вход задачи является устойчивым с коэффициентом γ порядка $\Omega(\sqrt{n})$, решать NP-трудную задачу максимального разреза за полиномиальное время. В недавней работе [3] этот результат улучшен и получен алгоритм, возвращающий точное решение для $\gamma \geq c\sqrt{\log n} \log\log n$ или выдающий ответ о том, что вход задачи не является γ-устойчивым с достаточно большим параметром γ.

В предыдущей работе [13] был приведен быстрый алгоритм для задачи поиска минимального разреза во взвешенном неориентированном графе с коэффициентом устойчивости γ порядка $O(n)$. Кроме того, там был представлен полиномиальный алгоритм для нахождения радиуса устойчивости в такой задаче. В данной работе будет показано почему тот быстрый алгоритм, основанный на принципе слияния вершин, неприменим в случае ориентированного графа. Кроме того, в данной работе доказывается, что нахождение радиуса устойчивости для задачи поиска максимального потока в ориентированном графе в ряде случаев является NP-полной задачей.

План статьи:

В разделе 2 показано почему алгоритмы из [13] не переносятся на случай ориентированного графа. В разделе 3 показано что при определенных условиях задача нахождения радиуса устойчивости в задаче поиска максимального потока в сети является NP-полной. В разделе 4 приведена модификация алгоритма из работы [2] для задачи MINCUT с радиусом устойчивости порядка $O(\sqrt{n})$. В заключительном разделе 5 формулируются открытые задачи.

2. Некорректность алгоритма слияния вершин в ориентированном графе.

В данном разделе показано почему результаты из работы [13], связанные с быстрыми алгоритмами для n-устойчивых графов, а также с нахождением радиуса устойчивости в задаче MINCUT, не обобщаются на случай ориентированных графов.

Быстрый алгоритм поиска минимального разреза во взвешенном неориентированном графе не переносится на случай ориентированного графа из-за того, что он основан на принципе слияния вершин. В частности, в доказательстве его корректности используется следующая лемма, впервые сформулированная в работе [1] для задачи максимального разреза.

Лемма. Пусть w — γ-устойчивый вход задачи MINCUT, $\gamma > 1$, а w' получается из w слиянием двух вершин, лежащих по одну сторону от минимального разреза в w. Тогда w' также является γ-устойчивым и его минимальный разрез состоит из ребер, входящих в минимальный разрез в w.

Данная лемма очевидным образом доказывается от противного в случае неориентированного графа, но неверна для графа ориентированного. Рассмотрим граф со следующей функцией весов:

$$\begin{matrix} 0 & 2 & 0 \\ 0 & 0 & 1 \\ 0 & 3 & 0 \end{matrix}$$

Минимальный разрез в этом графе равен $(S, \overline{S}) = (\{a, b\}, \{c\})$. Ребро $\{c; b\}$ не принадлежит разрезу, и его вершины не лежат с одной стороны от него.

Алгоритм же вычисления радиуса устойчивости не переносится на случай ориентированных графов, т.к. для них не существует дерева Гомори-Ху. Более того, как показано в [16], в ориентированном графе невозможно построить аналог дерева Гомори-Ху, т. е. дерево удовлетворяющее заданным условиям.

3. Сложность нахождения радиуса устойчивости.

В данном разделе мы сформулируем задачу нахождения радиуса устойчивости в виде проблемы распознавания и докажем что при некоторых дополнительных ограничениях входа данная задача является NP-полной. Итак, сформулируем задачу РАДИУС.

Вход: граф $G = (V, E, w)$, фиксированное число $R > 0$.

Выход: ответ на вопрос, существует ли минимальный разрез C и разрез C', отличный от C такой, что C' становится минимальным в графе G', полученном из графа G при некотором γ-возмущении, где $\gamma < R$.

В статье [13] был предложен полиномиальный алгоритм решения задачи РАДИУС в случае неориентированного графа. Но в предыдущем разделе показано, что тот подход принципиально неприменим в случае ориентированных графов. Для решения вопроса о сложности данной задачи в случае ориентированного графа мы вначале рассмотрим задачу $s - t$ РАДИУС, в которой в графе ищется разрез, разделяющий заданную пару вершин. Как уже было сказано ранее, задача поиска минимального $s - t$ разреза является двойственной к задаче поиска максимального $s - t$ потока. Покажем, что сложность нахождения радиуса устойчивости для двойственных задач совпадает.

4. Алгоритм для $O(\sqrt{n})$-устойчивого входа задачи MINCUT.

В работе [2] были предложены полиномиальные алгоритмы для решения задачи поиска максимального разреза в графе с коэффициентами устойчивости γ порядка $O(n)$ и $O(\sqrt{n})$. Аналог алгоритма для $O(n)$ устойчивого входа в задаче MINCUT был представлен в статье [13]. Теперь покажем, что для второго из алгоритмов из [2] также можно построить аналог, находящий решение задачи MINCUT. В данном разделе мы рассматриваем задачу MINCUT на взвешенном неориентированном графе.

Введем несколько определений, затем сформулируем алгоритм, докажем его корректность и оценим трудоемкость.

Определение. Пусть w — γ-устойчивый вход задачи MINCUT, $\gamma > 1$. Будем называть ребро uv тяжелым, если $w(uv) > \frac{1}{\gamma+1}\tau(v)$.

Определение. Пусть w — вход задачи MINCUT. Определим функцию соседства между парой вершин u и v как $n(u, v) = \sum_{z \in V} w(vz) \cdot w(zu)$.

Теперь можно сформулировать алгоритм.

Алгоритм 4

Вход: γ-устойчивый граф $G = (V, E, w), \gamma > \sqrt{2n + \frac{1}{4}} - \frac{1}{2}$.

Выход: минимальный разрез в графе G и его вес.

1: **пока** $|V_G| \geq 2$
2:     **пока** в G есть хотя бы одно тяжелое ребро e
3:         $G \leftarrow G/e$;
4:     найти пару вершин с достаточно большой функцией соседства: $n(u, v) > \frac{2}{(\gamma+1)^2}\tau(u)\tau(v)$
5:     $G \leftarrow G/uv$;
6: вернуть минимальный разрез и его вес.

Докажем корректность представленного алгоритма. Для этого покажем, что всякий раз мы сливаем вершины, находящиеся с одной стороны относительно разреза. Из Леммы 1 получаем

$$\tau(v) = \xi(v) + \iota(v) \geq \xi(v) + \gamma \cdot \xi(v) \Rightarrow \xi(v) \leq \frac{\tau(v)}{\gamma + 1}.$$

Следовательно, вес тяжелого ребра больше, чем суммарный вес ребер, исходящих из вершины v и лежащих в разрезе. Значит все тяжелые ребра в разрезе не лежат, и по Лемме 2 после слияния инцидентных вершин минимальный разрез остается неизменным.

Покажем теперь, что в графе без тяжелых ребер две вершины с достаточно большой функцией соседства находятся по одну сторону от разреза. Заметим, что для всех ребер и вершин в таком графе выполнены неравенства $w(uv) \leq \frac{\tau(v)}{\gamma+1}$ и $\xi(v) \leq \frac{\tau(v)}{\gamma+1}$. Но тогда функция соседства двух вершин по разные стороны от разреза будет достаточно малой. Пусть $v \in S,\ u \in \overline{S}$, тогда

$$\begin{aligned} n(u,v) = \sum_{z \in \overline{S}} w(vz) \cdot w(zu) + \sum_{z \in S} w(vz) \cdot w(zu) \leq \\ \leq \frac{1}{\gamma+1}\xi(v)\tau(u) + \frac{1}{\gamma+1}\xi(u)\tau(v) \leq \frac{2}{(\gamma+1)^2}\tau(u)\tau(v). \end{aligned}$$

Теперь осталось показать, что в графе с достаточно большим коэффициентом устойчивости γ всегда найдется пара вершин с достаточно большой функцией соседства. Докажем это от противного, пусть в графе нет тяжелых ребер и функция соседства для всех пар вершин достаточно мала.

$$\begin{aligned} \sum_{v \in V} \tau^2(v) = \sum_{u,v,z \in V} w(vz) \cdot w(zu) = \sum_{\substack{u,v \in V \\ u \neq v}} n(u,v) + \sum_{u,z \in V} w^2(uz) \leq \\ \leq \frac{2}{(\gamma+1)^2} \sum_{\substack{u,v \in V \\ u \neq v}} \tau(u)\tau(v) + \sum_{u \in V} \frac{1}{\gamma+1}\tau(u) \sum_{z \in V} w(uz) \leq \\ \leq \frac{2n}{(\gamma+1)^2} \sum_{u \in V} \tau^2(u) + \frac{1}{\gamma+1} \sum_{u \in V} \tau^2(u) \end{aligned}$$

Решая это неравенство получаем, что $\gamma \leq \sqrt{2n + \frac{1}{4}} - \frac{1}{2}$, что противоречит условию.

Оценим теперь трудоемкость предложенного алгоритма. Поиск и слияние тяжелых ребер в графе происходит за $O(n^2)$, самая же затратная по времени часть это расчет функций соседства для всех пар вершин в графе без тяжелых ребер. Исходя из определения функции соседства $n(u,v) = \sum_{z \in V} w(vz) \cdot w(zu)$ ее расчет представляет собой возведение в квадрат матрицы смежности графа, которое может быть сделано с помощью, например, алгоритма Штрассена за время $O(n^{\log_2 7})$. Таким образом, доказана следующая теорема.

Теорема 1. В $O(\sqrt{n})$-устойчивой задаче MINCUT точное решение может быть найдено за время $T(n)$, где $T(n)$ — трудоемкость умножения пары $n \times n$ матриц.

Сравним полученную трудоемкость с быстрейшими алгоритмами, решающими задачу минимального разреза в общем случае. Трудоемкость представленного алгоритма больше, чем у вероятностного алгоритма Каргера, но меньше чем у детерминированного алгоритма Штор-Вагнера. Более того, на некоторых входах она может быть уменьшена, т.к. перемножение матриц на них выполняется за квадратичное время.

Следствие 1. В $O(\sqrt{n})$-устойчивой задаче MINCUT с матрицей входа w ранга $O(1)$ точное решение может быть найдено за время $O(n^2)$.

Показана принципиальная возможность переноса идей алгоритмов для устойчивых входов NP-трудной задачи поиска максимального разреза на задачу поиска минимального разреза в графе, имеющую полиномиальное решение.

4. Заключение. Открытые вопросы.

В данной работе были рассмотрены полиномиальные задачи поиска минимального разреза в ориентированном графе и в сети, а также задача поиска максимального потока в сети и показано, что при дополнительном предположении о достаточно большой устойчивости входа решение может быть найдено быстрее, чем в общем случае. Кроме того, для рассматриваемой задачи были предложены полиномиальные алгоритмы нахождения радиуса устойчивости. Также была приведена модификация алгоритма из работы [2] для задачи MINCUT с радиусом устойчивости порядка $O(\sqrt{n})$.

Дальнейшие направления исследования связаны с получением аналогичных результатов для других задач на графах, например в задаче поиска минимального k-разреза с различными ограничениями. Также открытым вопросом остается построение ускоренных алгоритмов для рассмотренных полиномиальных задач с качественно меньшим радиусом устойчивости. Также полученные полиномиальные алгоритмы интересно рассмотреть на возможность эффективного распараллеливания, т. е. на принадлежность классу RNC.

5. Благодарности.

Автор выражает признательность Тимофееву Е.А. и Вялому М.В. за ценные замечания при подготовке данного текста.

Список литературы

[1] *Bilu Y., Linial N.* Are stable instances easy? // Innov. in Comp. Sci. 2010. P. 332–341.

[2] *Linial N. et.al.* On the practically interesting instances of MAXCUT // STACS. 2013. P. 526–537.

[3] *Makarychev K., Makarychev Y., Vijayaraghavan A.* Bulu-Linial stable instances of max cut and minimum multiway cut // Proc. of the 25-th ACM-SIAM Symp. on Disc. Alg. 2014. P. 890–906. DOI: 10.1137/1.9781611973402.67

[4] *Ford L., Fulkerson D.* Maximal flow through a network // Canadian Journal of Mathematics. 1956. № 8. P. 399–404.

[5] *Gomory R., Hu T.* Multi-terminal network flows // Journal of the Society of Industrial and Applied Mathematics. Dec.1961. P. 551–570.

[6] *Карзанов А.В., Тимофеев Е.А.* Эффективный алгоритм нахождения всех минимальных реберных разрезов неориентированного графа // Кибернетика. 1986. № 2. С. 8–12.

[7] *Hao J., Orlin J.* A faster algorithm for finding the minimum cut in a directed graph // Journal of Algorithms. Nov.1994. Vol. 17, № 3. P. 424–446.

[8] *Поддерюгин В.Д.* Алгоритм для нахождения реберной связности графа // Вопросы кибернетики. 1973. № 2. С. 136.

[9] *Karger D., Stein C.* A new approach to the minimum cut problem // Journal of the ACM. Jul.1996. Vol. 43, № 4, P. 601–640.

[10] *Stoer M., Wagner F.* A simple min-cut algorithm // Journal of the ACM. Jul.1997. Vol. 44, № 4. P. 585–591.

[11] *Orlin J.* Max flows in O(nm) time, or better // STOC. 2013. P. 765–774.

[12] *King V., Rao S., Tarjan R.* A Faster Deterministic Maximum Flow Algorithm // Journal of Algorithms. Nov.1994. Vol. 17, № 3. P. 447—474.

[13] *Козлов И.В.* Устойчивость в задаче поиска минимального разреза в графе // Моделирование и анализ информационных систем. 2014. Т. 21, № 4. С. 54–63.

[14] *Леонтьев В.К.* Устойчивость задачи коммивояжера // Ж. выч. матем. и матем. физ. 1975. Т. 5, № 15. С. 1298–1309.

[15] *Гордеев Э.Н., Леонтьев В.К.* Общий подход к исследованию устойчивости решений в задачах дискретной оптимизации // Ж. выч. матем. и матем. физ. 1996. Т. 1, № 36. С. 66–72.

[16] *Benzur A.* Counterexamples for directed and node capacitated cut-trees // SIAM Journal on Computing. 1995. Vol. 15, № 3. P. 505–510.

Гордеев Э.Н. Устойчивость решения в задаче о кратчайшем пути на графе // Дискрет. матем. – 1989, Т. 1, No 3, С. 39–46.