ДОКЛАДЫ АКАДЕМИИ НАУК, 2017, том 476, № 6, с. 614–616

МАТЕМАТИКА

УДК 519.178

КРАТЧАЙШЕЕ ПРЕОБРАЗОВАНИЕ ГРАФА

© 2017 г. К. Ю. Горбунов[1,*], В. А. Любецкий[1,2,]**

Представлено академиком РАН А.Л. Семеновым 13.05.2017 г.

Поступило 17.05.2017 г.

Получены полные доказательства того, что алгоритмы, предложенные авторами, решают задачу кратчайшего преобразования одного графа в другой. Задача решается как прямым линейной сложности алгоритмом, так и сведением её к квадратичному целочисленному линейному программированию.

DOI: 10.7868/S0869565217300028

ФОРМУЛИРОВКА ДВУХ ЗАДАЧ И РЕЗУЛЬТАТЫ

П е р в а я з а д а ч а. Даны ориентированные графы a и b и список операций над графом. Операции фиксированы, они предложены в [1] (их описание приведено также в [2, с. 162]); это – двойная, полуторная и одинарная (разрез и склейка) переклейки, удаление и вставка. Каждой операции присвоена цена – строго положительное рациональное число. Найти последовательность операций с наименьшей суммарной ценой, которая преобразует a в b. Графы в этой и следующей задачах – конечные множества непересекающихся цепей и циклов; каждому ребру приписано имя – натуральное число. В [1, 2] такие графы называют хромосомными структурами. В первой задаче имена рёбер в каждом графе не повторяются.

Наименьшую суммарную цену назовём расстоянием между a и b; хотя оно не является обычной метрикой. Решение задачи назовём кратчайшей последовательностью. Ориентация рёбер внутри цепей и циклов не предполагается согласованной.

В т о р а я з а д а ч а. Данные те же, что в первой задаче, за исключением: имена рёбер в графах могут повторяться, но зато цены всех операций равны. Свести её к квадратичному целочисленному линейному программированию (ЦЛП). “Квадратичному” означает: число переменных и ограничений квадратично зависит от суммарного размера исходных графов a и b, например, от числа рёбер в них. Множество рёбер с именем n обозначим $X(n, a)$ и $X(n, b)$ соответственно в a и b. “Свести” означает: для каждого n найти частично определённое инъективное отображение меньшего по числу элементов из множеств $X(n, a)$ и $X(n, b)$ в большее, для которого соответствующие графы a' и b' с именами рёбер уже без повторений имеют минимальное расстояние. Точнее, в a' рёбра, одноимённые в a, получают уникальные имена (и аналогично в b' и b), так чтобы уникальные имена сохранялись при этих отображениях. Уникальные имена можно получить, например, просто добавляя согласованные по отображениям вторые позиции к исходным именам, тогда уникальное имя имеет вид $n.k$ одновременно в a и b; тем самым повторение новых имён отсутствует в a (и аналогично в b).

Мы нигде не предполагаем, что множества имён в a и b совпадают, что принципиально усложняет обе задачи.

Кроме теоремы 1, сформулированной ниже, авторы получили строгое решение первой задачи для случая: цена d вставки находится в интервале от c – равных цен всех других операций до $2c$ (решение минимально с точностью до аддитивной константы $d - c$) [2]. Этот случай можно назвать нестационарным, а тот, что в теореме 1а, – стационарным. Отсюда сразу следует решение задачи для случая равных цен операций.

Первая задача поставлена в [3, 4], вторая – в упрощённом виде в [5, 6]. Первая задача широко исследовалась при решении разных прикладных вопросов, при этом предполагались сильные условия в её постановке, и в их присутствии предлагались эвристические алгоритмы решения;

[1] *Институт проблем передачи информации им. А.А. Харкевича Российской Академии наук, Москва*
[2] *Московский государственный университет им. М.В. Ломоносова*
**E-mail: gorbunov@iitp.ru*
***E-mail: lyubetsk@iitp.ru*

подробные ссылки можно найти, например, в [3, 4]. Насколько авторы осведомлены, математические результаты, связанные с первой задачей в её указанной постановке, содержатся только в [7, 8]. В [7] предложено решение этой задачи для частного случая равных цен операций. В [8] предложен план решения этой задачи в условиях, сформулированных ниже в теореме 1. Насколько авторам известно, до сих пор соответствующее доказательство не было опубликовано; по-видимому, предложенный план не удаётся реализовать; доказательство этого утверждения получено авторами на другом пути, см. ниже. Алгоритмы в [7, 8] принципиально отличаются от предлагаемых нами. Для второй задачи математические результаты содержатся только в [5, 6]. В [5] для частного случая: в графах множества имён рёбер совпадают; в [6] для другого частного своеобразного случая: графы могут иметь несовпадающие множества имён рёбер, но после установления биекции между $X(n, a)$ и $X(n, b)$ рёбра, представленные только в одном из графов, игнорируются при приведении a к b.

Для равнодоступной адресной машины с равномерным весовым критерием первая задача подразумевает решение посредством линейного алгоритма, вторая – квадратичного алгоритма.

Компонентой назовём цепь или цикл. Циклическим назовём граф, состоящий только из циклов. В [9] для случая совпадающих в циклических графах a и b множеств имён и равных цен операций предложен переход к двойственной задаче: приведению определяемого там (breakpoint) графа к финальному виду двойственными операциями. Мы определили такой граф для несовпадающего множества имён в произвольных графах a и b, ниже он обозначается $a + b$. Формулировки этих понятий приведены в [2, с. 163]. Другое введённое нами понятие – взаимодействие цепей, которое включает понятия типа и размера компоненты [2, с. 163–167].

Т е о р е м а 1. *Получен линейный алгоритм, который решает первую задачу для случая*:

а) *равные цены вставки и удаления не больше половины от равных цен остальных операций*;

б) *исходные графы циклические, цены вставки и удаления равны, цены других операций также равны между собой*.

Т е о р е м а 2. *Получен квадратичный алгоритм, который сводит вторую задачу к квадратичному* ЦЛП.

А л г о р и т м р е ш е н и я п е р в о й з а д а ч и. Для $a + b$ выполним шаги:

1) удалить все петли;

2) вырезать [2, с. 164, шаг 2] все обычные рёбра кроме 2-циклов;

3) применить взаимодействия 3.1, 3.2, 3.12, 3.13 [2] начиная с первого из возможных;

4) цепи строго положительного размера замкнуть [2, с. 170, пункт 4.21] ребром в циклы;

5) циклы (размера строго большего двух) разбить [2, с. 171, шаг 5] на 2-циклы;

6) удалить [2, шаг 5] оставшиеся особые вершины.

Н а б р о с о к д о к а з а т е л ь с т в а т е о р е м ы 1 а. Пусть $c(G)$ – минимально возможная суммарная цена последовательности операций, приводящей граф $G = a + b$ к финальному виду, $T(G)$ – суммарная цена операций в нашем алгоритме, который делает то же самое. Покажем, что для любого G выполняется

$$T(G) \leqslant c(G). \tag{1}$$

Важное равенство:

$$T(G) = (1 - w)(0{,}5d + 0{,}5f - c) + \\ + w(B + S + D) - P(G), \tag{2}$$

где w – цена операций удаления и вставки, d, f и c – соответственно суммарный размер компонент в графе $a + b$, числа нечётных цепей и циклов в нём; B – суммарное число особых вершин в $a + b$; S – сумма целых частей половин длин максимальных связных участков (назовём их отрезками) из обычных рёбер плюс число крайних нечётных отрезков и минус число циклических отрезков; D равно 1 в случае цепи одного из типов 1a, 1b, 3a, 3b, 3 (и 0 иначе) [2]; $P(G)$ – разность суммарных цен операций в указанном выше алгоритме, включая или исключая применение шага 3.

Неравенство (1) сразу следует из

$$c(o) \geq T(G) - T(o(G)) \tag{3}$$

для любой операции o и любого G. Для проверки (3) нужно перебрать все операции. Пусть o – удаление особой вершины. При переходе от G к $o(G)$ величина B уменьшается на 1. Далее рассмотрим возможности.

Удаляется изолированная особая вершина. При том же переходе S, D, c не меняются, d увеличивается на 1, f уменьшается на 1. Величина P не меняется или уменьшается на w; действительно, при удалении цепи она не может увеличиться, а уменьшиться больше, чем на w, не может, поскольку все взаимодействия, содержащие аргументы типов 2a или 2b, экономят цену w. В результате величина T не меняется или уменьшается на w.

Особая вершина удаляется из цикла или петли. При том же переходе S не меняется (если оба отрезка, примыкающих к удаляемой вершине, чётные, или вершина единственная в цикле) или увеличивается на 1, другие величины не меняются.

Из цепи удаляется внутренняя особая вершина, т.е. с обеих её сторон имеются другие особые вершины. Тогда тип цепи не меняется: он определяется размером цепи, краями (висячие или невисячие) и (в случае невисячего края) длиной крайнего отрезка. Повторяем рассуждение из предыдущего абзаца.

Другие операции рассматриваются аналогично.

Алгоритмы и набросок доказательства теоремы 1б. В $a + b$ все компоненты – циклы или петли; алгоритмы, указанные ниже, применяются к $a + b$. Из переклеек можно оставить только двойную. Пусть цена w вставки и удаления одинакова, а цены остальных операций равны 1. Особая операция вместе с её аргументами – та, при которой две особые вершины объединяются в одну. Особую двойную переклейку обозначим ОДП и говорим об ОДП-действии.

Пусть $0 < w \leqslant 1$. Здесь алгоритм и доказательство те же, что и выше, цена $T(G)$ равна

$$(1 - w) \cdot (0{,}5d - c) + w \cdot (B + S).$$

Пусть $1 < w \leqslant 2$. a-Цикл в общем графе содержит a-, но не b-вершины; аналогично определяется b-цикл; (a, b)-цикл содержит a- и b-вершины.

Алгоритм по сути близок к вышеуказанному:

1) вырезать обычные рёбра;

2) ОДП-объединить вершину петли с одноимённой особой вершиной любой компоненты;

3) удалить петли;

4) (a, b)-цикл размера строго больше 4 ОДП-разбить на 2-цикл и меньший цикл; в последнем ОДП-вырезать обычное ребро;

5) (a, b)-цикл размера 4 ОДП-инверсией преобразовать в цикл с одним обычным ребром, вырезать это ребро;

6) ОДП-объединить два (a, b)-цикла в один цикл, ОДП-вырезать обычное ребро;

7) удалить особые вершины.

Пусть C_a и C_b – число a-циклов в графе $a + b$ и соответственно b-циклов. В алгоритме число удалений особых вершин равно $C_a + C_b + 2I_{ab} + I_{pa} + I_{pb}$; здесь $I_{ab} = 1$, если в $a + b$ имеется (a, b)-цикл (иначе 0), $I_{pa} = 1$, если имеется a-петля, но нет цикла с a-вершиной (иначе 0), I_{pb} – аналогичный показатель для b-петли. Операции, кроме выполняемых на шаге 1, особые. В алгоритме число особых операций равно B, число неособых равно S, тогда

$$T(G) = w \cdot (C_a + C_b + 2I_{ab} + I_{pa} + I_{pb}) + (B + S - C_a - C_b - 2I_{ab} - I_{pa} - I_{pb}) = (w - 1) \cdot (C_a + C_b + 2I_{ab} + I_{pa} + I_{pb}) + B + S.$$

Нужно перебрать все операции o, применяемые к G.

Пусть $w > 2$. Первые шесть шагов предыдущего алгоритма продолжим следующими шагами:

7) ОДП-объединить (a, b)-цикл и a-цикл, вырезать любое из двух обычных рёбер. Аналогично объединить (a, b)-цикл и b-цикл;

8) ОДП-объединить два a-цикла, вырезать любое из трёх обычных рёбер. Аналогично объединить два b-цикла;

9) удалить особые вершины.

В алгоритме всего операций $B + S + C_a - I_{ca} \times (1 - I_{ab}) + C_b - I_{cb} \cdot (1 - I_{ab}) = B + S + C_a + C_b - (I_{ca} + I_{cb}) \cdot (1 - I_{ab})$; здесь $I_{ca} = 1$ ($I_{cb} = 1$), если в $a + b$ имеется a-цикл (b-цикл), и 0 иначе. Из них операций удаления $I_a + I_b$; здесь $I_a = 1$ ($I_b = 1$), если в $a + b$ имеется a-вершина (b-вершина), и 0 иначе. Тогда

$$T(G) = w \cdot (I_a + I_b) + (B + S + C_a + C_b - (I_{ca} + I_{cb}) \times (1 - I_{ab}) - I_a - I_b) = (w - 1) \cdot (I_a + I_b) + B + S + C_a + C_b - (I_{ca} + I_{cb}) \cdot (1 - I_{ab}).$$

Перебрать все операции.

СПИСОК ЛИТЕРАТУРЫ

1. *Yancopoulos S., Attie O., Friedberg R.*//Bioinformatics. 2005. № 21. P. 3340–3346.
2. *Gorbunov K.Yu., Lyubetsky V.A.*//CEUR Workshop Proceedings (CEUR-WS.org), Selected Papers of the First International Scientific Conference Convergent Cognitive Information Technologies (Convergent 2016). Moscow, November 25–26. 2016. M., 2016. V. 1763. P. 162–172.
3. *Braga M.D.V., Willing E., Stoye J.*//J. Comput. Biol. 2011. V. 18. P. 1167–1184.
4. *da Silva P.H., Machado R., Dantas S., Braga M.D.V.*// Algorithms Mol. Biol. 2013. V. 8. P. 21.1–21.15.
5. *Shao M., Lin Y., Moret B.* Proc. of RECOMB 2014// Lect. Notes Comp. Sci. 2014. V. 8394. P. 280–292.
6. *Martinez F.V., Feijão P., Braga M.D.V., Stoye J.*// Algorithms Mol. Biol. 2015. V. 10. P. 13.1–13.10.
7. *Compeau P.E.C.*//Algorithms Mol. Biol. 2013. V. 8. P. 6.1–6.9.
8. *Compeau P.E.C.* Proc. of 14th Intern. Workshop “Algorithms in Bioinformatics”. Wroclaw, September 8–10. 2014. Wroclaw, 2014. V. 8701. P. 38–51.
9. *Alekseyev M.A., Pevzner P.A.*//Theor. Comput.Sci. 2008. V. 395. № 2–3. P. 193–202.