О методе Ицуми–Вадаяма подсчета числа совершенных паросочетаний в графе

М. Н. Вялый*

ФИЦ ИУ РАН, ВЦ РАН им. А.А.Дородницына

Как известно, задача подсчёта числа совершенных паросочетаний в графе $\#P$-трудна [16]. Она остаётся $\#P$-трудной даже для 2-дольных 3-регулярных графов [6].

Самые быстрые известные алгоритм подсчёта числа совершенных паросочетаний в графе имеют трудоёмкость $O^*(2^{n/2})$ — это алгоритм Райзера для двудольных графов [14] и алгоритм Бйорклунда для произвольных графов [2]. Здесь и далее $O^*(f(n))$ означает оценку с точностью до множителей, полиномиальных по n.

Для многих классов графов известны более быстрые алгоритмы. Для планарных графов подсчёт числа совершенных паросочетаний выполняется за полиномиальное время [11]. Полиномиальные алгоритмы подсчёта числа совершенных паросочетаний найдены также для графов без минора $K_{3,3}$ [13] и графов без минора K_5 [15].

Число совершенных паросочетаний вычисляется за полиномиальное время также для графов ограниченной древесной ширины [5].

Активно изучался случай разреженных графов (т.е. графов с ограниченной средней степенью вершины Δ).

Наиболее сильный результат для разреженных графов получен в работе Т. Ицуми и Т. Вадаяма [10]. Они предложили алгоритм подчета числа совершенных паросочетаний, основанный на вычислении весового многочлена пространства разрезов двудольного графа. Оценка трудоемкости этого алгоритма $O^*(2^{\frac{1}{2}(1-1/(5\Delta\log\Delta))n})$ (здесь и далее n обозначает количество вершин в графе). Эта оценка лучше оценки алгоритма Райзера для двудольных графов (нужно обратить внимание, что количество вершин в доле равно $n/2$).

В данной работе мы исследуем возможности метода, предложенного в работе [10] метода для недвудольных графов и получаем ряд более быстрых алгоритмов, чем известные.

Теорема 1. *Для графов путевой ширины p существует алгоритм подсчета числа совершенных паросочетаний за время $O^*(2^{O(p^2)})$.*

Теорема 2. *Для регулярных графов степени* 3 *существует алгоритм подсчета числа совершенных паросочетаний за время $O^*(2^{(1/6+\varepsilon)n})$ для всех $\varepsilon > 0$.*

Теорема 3. *Для графов с запрещёнными минорами размера не больше h существует алгоритм подсчета числа совершенных паросочетаний за время $O^*(2^{O(h^{3/2}n^{1/2})})$.*

*Работа выполнена при поддержке РФФИ, номер проекта 14–01–00641

1. Связь между парочетаниями и весовыми многочленами

Метод Ицуми–Вадаяма основан на выражении числа совершенных паросочетаний как коэффициента в весовом многочлене пространства разрезов графа.

Метод применим только к графам, в которых все вершины имеют нечетную степень. Добавлением 2 вершин и $\leqslant n$ рёбер из любого графа легко конструируется граф с нечетными степенями вершин, имеющий то же количество совершенных паросочетаний, что и исходный граф.

Обозначим $\mathrm{Cyc}_G(x, y)$ однородный весовой многочлен пространства циклов графа G, а через $[k]\mathrm{Cyc}_G(x, y)$ — его коэффициент при мономе $x^k y^{m-k}$, т. е. количество точек в пространстве веса k.

Лемма 1 ([10]). *Пусть в графе G степени всех вершин нечетны. Тогда число совершенных паросочетаний равно $[m - n/2]\mathrm{Cyc}_G(t)$.*

Как известно, пространство циклов ортогонально пространству разрезов Cut_G. Весовые многочлены этих пространств связаны тождеством Мак-Вильямс:

$$\mathrm{Cyc}_G(x, y) = \frac{1}{|\mathrm{Cut}_G|}\mathrm{Cut}_G(y - x, y + x). \tag{1}$$

Таким образом, подсчёт числа совершенных паросочетаний сводится к нахождению весового многочлена пространства разрезов. Размерность Cut_G равна $n - c$, где c — число компонент связности графа.

Разумеется, достаточно находить весовой многочлен разрезов для связных графов.

2. Многомерные многочлены разрезов

Для каждого ребра e графа G введём переменные t_{e0} и t_{e1}. С каждой вершиной свяжем моном

$$D_u = \prod_{e\in\Gamma(u)} t_{e0} + \prod_{e\in\Gamma(u)} t_{e1}$$

и определим многомерный многочлен разрезов как

$$D_G = \prod_{u\in V(G)} D_u. \tag{2}$$

Коэффициенты $\mathrm{Cut}_G(x, y)$ выражаются через коэффициенты многочлена D_G. Связь задается линейным отображением $\varphi\colon \Bbbk[t_{e\alpha}] \to \Bbbk[x, y]$, которое на мономе $\prod_{e\in E,\alpha\in\{0,1\}} t_{e\alpha}^{s(e,\alpha)}$, где $s(e, \alpha) \in \{0, 1\}$ равно $x^a y^b$, если для ровно a рёбер выполняется равенство $s(e, 0)+s(e, 1) = 1$. На остальных мономах отображение φ может быть каким угодно.

Из определений очевидно следует

Утверждение 1. $\varphi(D_G) = 2\mathrm{Cut}_G(x, y)$ *для связного графа G.*

Множитель 2 связан с тем, что для связного графа отображение разрезов имеет одномерное ядро. В общем случае множитель равен 2^c, где c — количество компонент связности.

Вычисление D_G «по определению» занимает время $O^*(2^n)$. Нас интересуют способы ускорения этого вычисления. Они основаны на том, что хотя отображение φ и не является гомоморфизмом колец, оно, точнее его модификации, всё же обладает некоторыми хорошими свойствами.

Обозначим $R = \Bbbk[x, y]$. Продолжим отображение φ до R-линейного отображения кольца $R[t_{e0}]$ в себя. Это отображение задается на мономах следующими правилами

$$\begin{aligned}
&\varphi\colon t_{e\alpha} \mapsto t_{e\alpha},\\
&\varphi\colon t_{e0}t_{e1} \mapsto x,\\
&\varphi\colon t_{e0}t_{e0} \mapsto y,\\
&\varphi\colon t_{e1}t_{e1} \mapsto y,\\
&\varphi\colon \prod_{e\in E',\alpha} t_{e\alpha} \prod_{e\in E'',\alpha} t_{e\alpha} \mapsto \varphi(\prod_{e\in E',\alpha} t_{e\alpha})\varphi(\prod_{e\in E'',\alpha} t_{e\alpha}), \quad \text{если } E' \cap E'' = \varnothing.
\end{aligned} \tag{3}$$

По линейности последнее равенство продолжается на любые многочлены, в которых переменные не пересекаются.

Утверждение 2 (слабая мультипликативность). *Если в многочленах F, G переменные относятся к различным рёбрам, то $\varphi(FG) = \varphi(F)\varphi(G)$.*

Правила (3) не задают действие φ на мономах вида $t_{e0}^i t_{e1}^j$. Нам этот случай и не потребуется. Но для полноты определения скажем, что такие мономы чётной степени переходят в многочлены нулевой степени, а мономы нечётной степени — в многочлены первой степени. Образ монома определяется так: выбираем паросочетание на множителях монома (кроме разве что одного в случае мономов нечётной степени) и применяем правил (3) к выбранным парам. Суммируем по всем выборам с равными весами так, чтобы сумма коэффициентов равнялась 1.

В силу этих определений отображение φ переводит всё кольцо $R[t_{e\alpha}]$ в подкольцо R-полилинейных многочленов.

Для алгоритмических приложений важно, что арифметика в кольце R выполняется быстро, так как это кольцо многочленов от двух переменных. И размеры возникающих многочленов от двух переменных не очень большие (так как степени многочленов полиномиально ограничены).

Отсюда получается общая идея ускорения вычисления многомерного многочлена разрезов: применять слабую мультипликативность при раскрытии скобок, уменьшая размер записи текущего многочлена. Ниже мы рассмотрим три способа реализации этой идеи.

3. Алгоритм последовательного раскрытия скобок

Вычисление D_G можно организовать так, чтобы проводить частичные сокращения по правилам (3) с учётом мультипликативности по непересекающимся множествам рёбер.

Упорядочим вершины $V(G)$: $v_1, v_2, \ldots, v_n$ и будем раскрывать скобки в D_G в этом порядке.

Применяем слабую мультипликативность как только это возможно.

На i-м шаге получаем многочлен

$$D^{(i)}(G) = \left(\sum_{j=1}^{N_i} f_j(x, y) M_j(t_{e\alpha})\right) \prod_{v_k : k>i} D_{v_k},$$

где $f_j \in \Bbbk[x, y]$, M_j — моном, не включающий переменных $t_{e\alpha}$ для $e = (v_p, v_r)$, $p, r \leqslant i$.

Трудоёмкость алгоритма определяется максимальным N_i в процессе раскрытия скобок.

Оценить трудоемкость алгоритма в зависимости от порядка выбора скобок достаточно легко. Упорядочим вершины графа:

$$v_1,\ v_2,\ \ldots, v_n$$

и будем раскрывать скобки в выбранном порядке.

Обозначим $V_i = \{v_1, \ldots, v_i\}$, а через ∂V_i — те вершины из V_1, из которых ведёт ребро в $\bar{V}_i$.

Утверждение 3. *После раскрытия i скобок и приведения подобных количество мономов не превосходит $2^{\partial V_i}$.*

Доказательство. Рёбрам, соединяющим вершины V_i, отвечают переменные, которые не встречаются в нераскрытых мономах. Поэтому в силу частичной мультипликативности к таким переменным допустимо применение правил сокращения (3). Таким образом, с точностью до множителей из кольца $k[x, y]$ остаются только мономы, отвечающие раскрытию скобок двучленов, индексированных множеством ∂V_i. □

Минимальная (по выбору порядка) величина максимума (по i) ∂V_i напрямую связана с такой характеристикой графа как путевая ширина графа (path width). Нам удобно определить линейное представление графа следующим образом. Это отображение вершин графа в множество отрезков на пути P_k такое, что отрезки, отвечающие смежным вершинам, пересекаются. Путевой шириной линейного представления называется максимальное (по вершинам пути) количество отрезков, проходящих через вершину пути. Путевой шириной графа $\mathrm{pw}(G)$ называется минимальная ширина линейного представления для данного графа, уменьшенная на 1.

Уменьшение на 1 делает равной 1 ширину линейного представления для пути.

Такое определение равносильно стандартному, в котором выбрасываются те вершины пути, которые не являются началом или концом отрезка, отвечающего вершине.

Обозначим минимальную (по выбору порядка) величину максимума ∂V_i по выбору i через $\min \partial V$.

Лемма 2. $\mathrm{pw}(G) \leqslant \min \partial V \leqslant \mathrm{pw}(G) + 1$.

Доказательство. Выберем оптимальный порядок вершин графа. Возьмём путь длины n. Сопоставим вершине v_i отрезок $[i, j]$, где j — максимальный номер соседей вершины v_i (или $[i, i]$, если все соседи имеют меньшие номера).

Ширина такого линейного представления не больше $\min \partial V + 1$ по выбору i. Действительно, над любой вершиной пути i проходят как отрезки, отвечающие вершинам из ∂V_i, так и отрезок, отвечающий вершине i (которая, вообще говоря, не обязана лежать в ∂V_i).

Таким образом, $\min \partial V \geqslant \mathrm{pw}(G)$.

В обратную сторону. Возьмём оптимальное по ширине линейное представление и занумеруем вершины в порядке начал отвечающих им отрезков (если начал отрезков совпадают, выбираем между ними порядок произвольно). Если начало i-го отрезка равно k, то в ∂V_i не больше вершин, чем отрезков, проходящих через k (у вершин, отрезки которых закончились раньше, нет рёбер в v_i и более поздние вершины). Поэтому $\min \partial V \leqslant \mathrm{pw}(G) + 1$. □

Следствие 1. *Если $\mathrm{pw}(G) \leqslant f(G)$ и для класса графов существует алгоритм построения линейного представления ширины $f(G)$ за время $O^*(2^{f(G)})$, то существует и алгоритм подсчёта числа совершенных паросочетаний, работающий за время $O^*(2^{f(G)})$.*

Теорема 1 получается комбинацией этого следствия и следующей теоремы, которая даёт наилучшую оценку на время построения оптимального линейного представления графа.

Теорема 4 (Bodlaender, Kloks, 1993, [3, 4]). *Линейное представление, реализующее путевую ширину графа, находится за время $O(2^{O(\mathrm{pw}(G)^2)}n)$, где n — число вершин, $\mathrm{pw}(G)$ — ширина графа.*

Ограничение сложности алгоритмов через ширину линейного (или древесного) представления — хорошо известный факт для многих задач алгоритмической теории графов. Метод Ицуми–Вадаяма добавляет в этот список задачу подсчёта числа совершенных паросочетаний.

4. Графы средней степени Δ

В этом разделе мы рассмотрим применение японского метода к графам со средней степенью вершины Δ.

Заметим, что сводимость к случаю графов нечётной степени состоит в добавлении одной или двух вершин, поэтому путевая ширина графа увеличивается максимум на 2.

В [12] получена оценка $m/5.769 + O(\log n)$ на путевую ширину графа с m рёбрами, причём соответствующее разложение графа находится за полиномиальное время. Применение этой оценки даёт при $\Delta = 3$ алгоритм со временем работы $O^*(2^{0.26n})$, что лучше оценки оригинального японского метода для двудольных графов (множитель в показателе $0.5(1 - 1/(15\log 3) \approx 0.479)$.

Однако при достаточно больших $\Delta > 11$ оценка $m/5.769$ становится больше n и хуже тривиальной.

В [8, Cor.6.6, p. 130] содержатся как указанная выше оценка на путевую ширину, так и более точная для $\Delta = 3$ (а именно, $1/6 + \varepsilon$) и для $\Delta = 4$ (а именно, $1/3 + \varepsilon$, лемма 6.7, p. 133). Заметим, что оценка $1/6 + \varepsilon$ для кубических графов была получена впервые в работе [7].

Более точно, для любого $\varepsilon > 0$ для средней степени Δ за полиномиальное время строится линейное представление путевой ширины $\frac{\Delta-2}{6}\,n + \varepsilon n$.

Из оценки путевой ширины для графов со средней степенью 3 немедленно следует теорема 2.

5. Алгоритм рассечения

Пусть после раскрытия части скобок оставшийся граф G' распался на связные компоненты. В силу частичной мультипликативности их можно обработать по отдельности. Это позволяет сократить количество мономов, которые используются одновременно.

Таким образом, вопрос об оценке числа мономов свёлся к оценкам размеров вершинных сепараторов.

Подмножество вершин графа называется ε-сепаратором, если подграф, индуцированный дополнением к этому множеству, состоит из связных компонент, размер каждой из которых не больше εn.

Обозначим через $S(\alpha, \beta)$ класс графов, каждый индуцированный подграф в которых имеет α-сепаратор размера не больше βn и существует алгоритм построения такого сепаратора за время $O^*(2^{n\beta/(1-\alpha)})$. Рекурсивное выделение сепараторов даёт следующую лемму.

Лемма 3. *Существует алгоритм вычисления многочлена разрезов для графов из класса $S(\alpha, \beta)$ с временем работы $O^*(2^{n\beta/(1-\alpha)})$.*

Наличие маленьких вершинных сепараторов в деревьях (и даже планарных графах) даёт для них псевдополиномиальные или субэкспоненциальные оценки времени вычисления многочлена разрезов.

Теорема Алона и др. [1] утверждает, что в графах с запрещёнными минорами размера h есть 2/3-сепаратор размера $h^{3/2}n^{1/2}$. (Это обобщение теоремы Липтона–Тарьяна для планарных графов.) Отсюда получаем субэкспоненциальный алгоритм подсчёта числа совершенных паросочетаний для любого класса графов с фиксированным списком запрещённых миноров (или даже с запрещёнными минорами логарифмического размера), т.е. теорему 3.

Вершинные сепараторы связаны с древесной шириной графа. Она отличается от линейной тем, что вершинам графа сопоставляются поддеревья некоторого дерева. Обозначим древесную ширину через $\mathrm{tw}(G)$.

В [9] указана оценка ($\tilde{s}$ — это сбалансированный показатель вершинного разделения):

$$\tilde{s}(G) - 1 \leqslant \mathrm{tw}(G) \leqslant 1 + \tilde{s}(G) \cdot \log n. \qquad (4)$$

Поэтому выполняется следующая лемма.

Лемма 4. *Если для графов из некоторого класса можно достаточно быстро находить оптимальное древесное представление ширины βn, то можно вычислять многочлен разрезов за время $O^*(2^{2\beta n/(1+\beta)})$.*

Хотя $\mathrm{tw}(G) \leqslant \mathrm{pw}(G)$ по определению, множитель $2/(1+\beta)$ в показателе делает две приведённые оценки несравнимыми.

Литература

[1] N. Alon, P. Seymour, and R. Thomas. A separator theorem for graphs with an excluded minor and its applications. // STOC'90 Proceedings of the twenty-second annual ACM symposium on theory of computing. - 1990. - P. 293-299.

[2] Björklund A. Counting perfect matchings as fast as Ryser // SODA '12 Proceedings of the twenty-third annual ACM-SIAM symposium on Discrete Algorithms. 2012. P. 914-921.

[3] Hans L. Bodlaender. A linear time algorithm for finding tree-decompositions of small treewidth // STOC'25. 1993. P. 226–234.

[4] Hans L. Bodlaender, T. Kloks. Efficient and Constructive Algorithms for the Pathwidth and Treewidth of Graphs. J. of Algorithms. 1996. Vol. 21, issue 2, P. 358–402.

[5] B. Courcelle, J. A. Makowsky and U. Rotics. On the fixed parameter complexity of graph enumeration problems definable in monadic second-order logic. Discrete Applied Mathematics, 108:23-52 (2001).

[6] P. Dagum, M. Luby. Approximating the permanent of graphs with large factors.

[7] F.V.Fomin, K. Høie. Pathwidth of cubic graphs and exact algorithms. Inf. Proc. Letters. 97 (2006), p. 191–196.

[8] Serge Gaspers. Exponential Time Algorithms: Structures, Measures, and Bounds. VDM Verlag Dr. Mueller e.K. 2010.

[9] Hermann Gruber. On Balanced Separators, Treewidth, and Cycle Rank. arXiv:1012.1344v2. 2013.

[10] Taisuke Izumi and Tadashi Wadayama. 2012. A New Direction for Counting Perfect Matchings. In Proceedings of the 2012 IEEE 53rd Annual Symposium on Foundations of Computer Science (FOCS '12). IEEE Computer Society, Washington, DC, USA, 591-598.

[11] P. W. Kasteleyn. Graph theory and crystal physics. In F. Harary, editor, Graph Theory and Theoretical Physics, pages 43–110. Academic Press, 1967.

[12] Joachim Kneis, Daniel Mölle, Stefan Richter, and Peter Rossmanith. A Bound on the Pathwidth of Sparse Graphs with Applications to Exact Algorithms. SIAM J. Discrete Math., (2009) 23(1), 407–427.

[13] C. H. C. Little. An extension of Kasteleyn's method of enumerating the 1-factors of planar graphs. In D. A. Holton, editor, Combinatorial Mathematics, volume 403 of Lecture Notes in Mathematics, pages 63–72. Springer Berlin Heidelberg, 1974.

[14] Ryser, H.J.: Combinatorial mathematics. Carus Math. Monographs, No. 14. Math. Assoc. of America, Washington, DC (1963)

[15] Simon Straub, Thomas Thierauf, Fabian Wagner. Counting the Number of Perfect Matchings in K_5-free Graphs. ECCC, TR–79, 2014.

[16] Valiant L.G. The complexity of computing the permanent. Theoretical Computer Science. 1979. V. 8. P. 189–201.