

Подсчёт циклов в двудольных графах с длиной менее трёх обхватов

А.Н.Воропаев

Петрозаводский государственный университет, Петрозаводск, Россия

Поступила в редколлегию 22.12.2011

Аннотация—Результаты предыдущей работы, посвящённой подсчёту коротких циклов в двудольных графах, обобщены на случай, когда длина цикла меньше трёх обхватов графа. Посредством вычислительных экспериментов показано, что при значениях обхвата $g = 4, 6, 8, 10$ наибольшие кратности сумм в явных формулах для подсчёта циклов длиной $2g + 2, 2g + 4, \dots, 3g - 2$ в двудольных графах равны 4. В то же время, найдено семейство двудольных форм замкнутых маршрутов длиной $3g$ с обхватом g , которым соответствуют шестикратные суммы (для всех чётных значений $g \geq 4$). Оценка практической эффективности явных формул для подсчёта циклов длиной менее $3g$ выполнена в сравнении с несколькими известными перечислительными методами.

1. ВВЕДЕНИЕ

Данная работа является завершением исследования эффективности подсчёта циклов в двудольных графах с учётом обхвата. Ограничение длин циклов относительно обхвата интересно, в первую очередь, тем, что позволяет существенно уменьшить сложность вычисления количества циклов по сравнению с общим случаем.

Вычислительная сложность явных формул для количества циклов длиной вплоть до $2g$, представленных в [1], сопоставима со сложностью умножения $n \times n$ -матриц. По причине громоздкости явные выражения известны только для небольших значений обхвата. В рамках [1] был выведен набор формул для случаев $k \leq \min\{18; g + 10\}$. Когда длина цикла превышает значение $2g$, в выражениях возникают суммы кратности 4 и более. С увеличением обхвата неизбежно уменьшается верхняя граница плотности графа, поэтому при подсчёте циклов могут оказаться целесообразны и перечислительные методы.

В [1] теоретические оценки сложности различных алгоритмов, включая явные формулы и известные перечислительные методы, выводились в рамках класса регулярных двудольных графов. Поскольку все рассматриваемые в работе алгоритмы полиномиальны (за счёт ограничения длины цикла), то достоверные практические выводы о соотношении временных затрат можно сделать только при существенном отличии порядков сложности. Если же эти порядки близки, то возникает вопрос, начиная с какого количества вершин в графе начинают «работать» асимптотические оценки. Кроме того, заранее неизвестно, насколько сильно сказывается структура графа на вычислительных затратах.

Настоящая работа была выполнена с двумя целями.

- Расширить набор известных данных о кратностях сумм, входящих в явные формулы, для различных значений обхвата графа и длины цикла. Основной практический интерес представляет выяснение условий, при которых наибольшая кратность суммы равна 4.
- Экспериментально исследовать эффективность явных формул в сравнении с перечислительными методами на различных семействах графов. Как и в [1], задействовались поиск с возвратом на основе [2] и алгоритм «цепей» [3, 1].

В разделе 3 кратко представлена структура явных формул и указано семейство форм замкнутых маршрутов длиной $3g$, которым соответствуют шестикратные суммы. Приводятся характеристики реализованного набора формул для подсчёта циклов длиной менее $3g$ в двудольных графах с обхватами $g = 4, 6, 8$. В разделах 4 и 5 перечислены тестированные реализации алгоритмов и семейства графов, для которых выполнялись расчёты. Раздел 6 отведён для результатов вычислительных экспериментов. В приложении приводится «статистика» циклов для ряда графов, вычисленная в рамках данной работы.

2. ОСНОВНЫЕ ОПРЕДЕЛЕНИЯ

В основном, в работе используется терминология [4] и рассматриваются двудольные неориентированные конечные графы без петель и кратных рёбер.

Маршрутом длиной k называется упорядоченный набор $(v_1; v_2; \dots; v_{k+1})$ вершин графа, такой что вершины v_i и v_{i+1} смежны. Цепь — это маршрут, в котором все вершины различны, а цикл есть маршрут длиной не менее 3, в котором все вершины кроме последней различны, а последняя совпадает с первой. При подсчёте циклы, отличающиеся только выбором начальной вершины или направления обхода вершин, рассматриваются как один.

Символы n , m и g закреплены за количеством вершин (порядком), количеством рёбер (размером) и длиной кратчайшего цикла (обхватом) графа, в котором подсчитываются циклы. Буквой k будем обозначать длину или наибольшую длину подсчитываемых циклов, а символом c_k — количество циклов длиной k . Граф, состоящий из одного простого цикла с k вершинами, обозначается C_k , а из одной простой цепи с k вершинами — P_k .

При оценке сложности алгоритмов на графах важное место занимает понятие плотности графа. Вычислительные затраты методов, рассматриваемых в данной работе, при дополнительном условии регулярности графа удобно выразить в виде $O(n^\alpha d^\beta)$, где n — порядок графа, d — значение степеней вершин графа, а α и β — константы. Если зафиксировать связь между порядком и степенью так, что $d = \Theta(n^\rho)$, то получим оценку в терминах только количества вершин графа n : $O(n^{\alpha+\beta\rho})$.

Указанное соотношение между d и n можно использовать для определения величины ρ (плотности) в случае бесконечного семейства регулярных графов. Например, для полных двудольных графов $\rho = 1$, так как $d = n/2 = \Theta(n)$. Естественно обобщить данное определение плотности на случай конкретного двудольного графа (не обязательно регулярного), формально положив $\rho = \log_{n/2} m - 1$. Тогда для полных двудольных графов $K_{n/2; n/2}$ в точности $\rho = 1$.

3. ЯВНЫЕ ФОРМУЛЫ

Явные формулы для количества циклов длиной k основаны на перечислении всевозможных форм замкнутых маршрутов длиной k . Сопоставим с маршрутом подграф, состоящий из вершин и рёбер, по которым проходит маршрут. Например, циклу длиной k соответствует подграф, изоморфный C_k , а замкнутому маршруту длиной k , проходящему по единственному ребру, — подграф, изоморфный P_2 . Отвлекаясь от обозначений вершин, будем говорить о графе, сопоставленном маршруту, как о форме этого маршрута.

Формам маршрутов соответствуют определённые суммы, составленные из произведений элементов матрицы смежности графа, её степеней и более сложных вспомогательных матриц [5, 1]. Комбинация таких сумм даёт формулу для подсчёта циклов длиной k . Например,

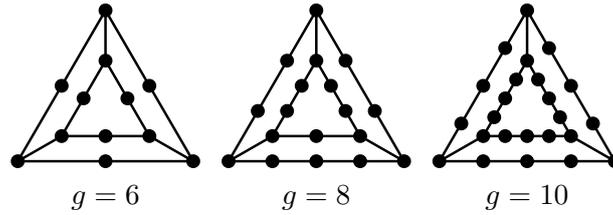


Рис. 1. Двудольные формы замкнутых маршрутов длиной $3g$ с обхватом g , приводящие к шестикратным суммам

количество циклов длиной 8 в графе с обхватом 6 равно

$$c_{8g6} = \frac{1}{16} \sum_{i=1}^n \left(a_{ii}^{(8)} - 8 a_{ii}^{(6)} d_i + 8 a_{ii}^{(6)} + 6 d_i^4 - 28 d_i^3 + 44 d_i^2 - 11 d_i \right) + \frac{1}{4} \sum_{i=1}^n \sum_{j=1}^n \left(a_{ij}^{(4)} + 4 a_{ij}^{(2)} d_i^2 - 8 a_{ij}^{(3)} \right), \quad (1)$$

где символом a_{ij} обозначается элемент матрицы смежности, $a_{ij}^{(l)}$ — элемент l -й степени матрицы смежности, а d_i — степень вершины i . Если обхват равен 4, то формула для подсчёта циклов длиной 8 состоит не из 10 членов, как (1), а из 20. Общее же выражение для числа циклов длиной 8 в произвольных графах содержит 35 членов, причём одна из сумм является четырёхкратной.

Кратность суммы, входящей в формулу, не превосходит количества вершин формы со степенями не менее трёх [5]. Если степени всех вершин не меньше трёх, то кратность суммы в точности равна порядку формы [5]. В [1] приводится класс форм замкнутых маршрутов длиной $2g + 2$ с обхватом g , которым соответствуют четырёхкратные суммы. При меньших длинах маршрутов кратности всех сумм могут быть понижены до значений менее трёх, так что наиболее трудоёмкой операцией оказывается умножение матриц. Именно такая ситуация имеет место в случае c_{8g6} .

Начиная со значения длины $3g$, появляются формы, для которых суммы имеют кратности 6 и более. При $g = 4$ одной из форм замкнутых маршрутов длиной 12 является полный двудольный граф $K_{3,3}$, которому соответствует шестикратная сумма. Примеры «тяжёлых» форм для больших значений обхвата, также дающих шестикратные суммы, представлены на рисунке 1.

Когда g не кратно четырём, такие формы получаются из призматического графа $C_3 \times P_2$ путём разбиения каждого ребра «оснований» на $g/2 - 1$ рёбер. Если же значение g кратно четырём, то ребро одного «основания» разбивается на $g/2$ рёбер, «параллельное» ему ребро другого «основания» — на $g/2 - 2$ рёбер, а остальные рёбра обоих «оснований» — на $g/2 - 1$ рёбер. Замкнутый маршрут длиной $3g$, проходящий по всем рёбрам построенного графа, можно составить из маршрутов, которые проходят по «внутреннему» и по «внешнему» циклам. Рёбра, соединяющие оба цикла, можно «захватывать» по мере прохождения одного из циклов, затрачивая на каждое ребро два шага.

Для диапазонов длин с $2g + 2$ по $3g - 2$ при $g = 4, 6, 8$ и с $2g + 2$ по $3g - 4$ при $g = 10$ были сгенерированы наборы всех форм маршрутов, как описано в [5, 1]. Проверка кратностей сумм, соответствующих этим формам, с учётом уменьшения кратностей согласно [5], показала, что наибольшая кратность суммы равна 4.

С увеличением длины маршрута количество форм резко возрастает. При фиксированном обхвате и увеличении длины маршрута на два наблюдается примерно четырёхкратный рост, а при увеличении обхвата на два число форм замкнутых маршрутов длиной $2g + 2$ становится приблизительно в шесть раз больше. В силу такого роста при $g = 10$ и $k = 28$ вместо генерации всех форм оказалось эффективнее проверить ряд небольших графов с обхватом 10, которым соответствуют суммы кратности более 4, и выяснить, являются ли они формами замкнутых маршрутов длиной 28.

Перебор графов осуществлялся с помощью программ `gens` и `rickg` из пакета `nauty` [6]. Кратности соответствующих графам сумм находились посредством вывода и упрощения самих сумм. Поиск же маршрутов длиной 28, проходящих по всем рёбрам графа, выполнялся путём бэктрекинга. Используемые средства пакета `nauty` удобны для порождения узкого класса «потенциальных» форм, как в указанном выше случае. Аналогичный подход к генерации всех форм практически неприемлем, поскольку исходное количество графов, порождаемых `gens`, оказывается очень велико (при не слишком малых значениях длины).

Исследование кратностей сумм при больших значениях обхвата, начиная с 12, затруднительно и по причине не слишком большой практической значимости не проводилось. Во-первых, при десятках тысяч форм проблематично выводить сами формулы (такое количество возникает уже при обхвате 10). Во-вторых, наиболее интересны именно случаи небольшого обхвата (4, 6, 8). Например, в коллекции [7] большинство графов имеют обхваты 6, только один граф — обхват 12, а остальные — обхваты 4 и 8.

В ходе выполнения данной работы были выведены и реализованы наборы формул для подсчёта циклов длиной менее $3g$ в двудольных графах с обхватами не более 8. В таблице 1 представлено общее количество сумм, используемых для подсчёта циклов с длинами из диапазона $g, g + 2, \dots, 3g - 2$. Для каждого значения длины число циклов получается путём комбинации значений определённых сумм, так же как в (1).

Таблица 1. Количество сумм в явных формулах для подсчёта циклов длиной до $3g$

g	4	6	8
Все суммы	59	528	6384
Четырёхкратные суммы	1	10	127

При значениях $g = 6, 8$ в исследовании наибольших кратностей сумм удалось продвинуться до границы $k = 4g$ (при помощи пакета `nauty` [6]). Выяснилось, что до этой границы кратность 6 «не сохраняется» в качестве наибольшей (по аналогии со случаями $2g$ и $3g$). Уже при значении $k = 3g + 2$ возникают формы, которым соответствуют восьмикратные суммы. В случае $g = 8$ и $k = 30 = 4g - 2$ можно отыскать форму, приводящую к десятикратной сумме.

Таким образом, продвижение по длине цикла более трёх обхватов сопряжено не только с резким ростом числа форм, но и с увеличением наибольшей кратности суммы. Данные обстоятельства существенно снижают интерес к формулам для подсчёта циклов длиной более $3g$ и саму доступность их вывода. В то же время, известные альтернативные способы подсчёта циклов, которые формально применимы для произвольных значений длины цикла, имеют сложность ещё большего порядка.

4. ПРОГРАММЫ

В данной работе вычисления выполнялись в тех же условиях, что и в [1].

Явные формулы ($g = 4, 6, 8, k = g, g + 2, \dots, 3g - 2$) реализованы в трёх вариантах. В первых двух основным типом данных является `long long int` (64 бит). Второй вариант отличается от

первого только тем, что при умножении на матрицу смежности учитывается разреженность графа. За счёт этой оптимизации в ряде случаев достигается семикратное ускорение. В третьем варианте использован тип Int128 (128 бит), разработанный в рамках [8], разреженность не учитывается. Этот вариант потребовался для подсчёта циклов в плотных графах с обхватами 4 и 6. Программа с применением Int128 работает в 3,5–11 раз медленнее «64-битового» аналога (тип Int128 реализован на языке C++ без использования архитектурных особенностей). На рисунках и в таблицах перечисленные варианты реализации обозначаются соответственно «фор», «фор2» и «фор3». Исходный код программ на языке C++ можно найти на сайте [9].

Алгоритм «цепей» для подсчёта циклов длиной с 4 по k воплощён в двух вариантах. В первом случае дерево цепей строится до высоты $k/2 - 1$, как предписано в [3]. Во втором случае дополнительно строится ещё один уровень [1]. Согласно [1], временные затраты данных версий алгоритма «цепей» для семейства регулярных графов с плотностью ρ оцениваются величинами $O(n^{1+(k-1)\rho})$ и $O(n^{1+k\rho})$ соответственно.

Во всех проведённых экспериментах, за исключением случая полных двудольных графов, второй вариант алгоритма оказался эффективнее оригинальной версии. Данное «расхождение» с указанной выше теоретической оценкой объясняется тем, что наличие дополнительного уровня дерева позволяет отсортировать цепи по конечным вершинам и избежать за счёт этого явной проверки многих пар цепей, заведомо не образующих циклы. На рисунках и в таблицах оригинальный и улучшенный варианты алгоритма «цепей» обозначаются «цеп0» и «цеп».

Поиск с возвратом основан на алгоритме [2]. Описание модификации можно найти в [1]. Как и первый вариант алгоритма «цепей», разработанная версия бэктрекинга характеризуется сложностью $O(n^{1+(k-1)\rho})$, но практически оказывается существенно медленнее алгоритма «цепей». Дополнительные затраты связаны с тем, что вторая половина цикла всякий раз строится заново. На рисунках и в таблицах бэктрекинг сокращённо обозначается «бэк».

Более подробное описание упомянутых выше перечислительных методов можно найти в [1].

5. ГРАФЫ

Для расчётов использовались, как и в [1], следующие семейства графов.

- $K_{n/2;n/2}$ — полные двудольные графы.
- $PG_2(q)$ — графы инцидентности точек и прямых дезарговых конечных проективных плоскостей порядков q , где q — степень простого числа. Граф $PG_2(q)$ содержит $2(q^2 + q + 1)$ вершин, является регулярным графом степени $q + 1$ и имеет обхват 6.
- $W(q)$ — графы инцидентности точек и прямых некоторых обобщённых четырёхугольников, где q — степень простого числа. Граф $W(q)$ содержит $2(q^3 + q^2 + q + 1)$ вершин, является регулярным графом степени $q + 1$ и имеет обхват 8.
- $R_{n/2;n/2}(g; \rho)$ — «случайные» графы с одинаковым количеством вершин в долях, обхватом g и размером, равным ближайшему целому числу к $(n/2)^{1+\rho}$.
- Графы некоторых LDPC-кодов, представленных в [7], с обхватами 6 и 8 (таблица 2). Обхват 8 имеет только последний граф. В связи с возросшей сложностью подсчёта циклов набор графов из [1] был заменён набором графов меньшего порядка. Графы $PEGirReg252x504$ и $PEGReg252x504$, являвшиеся самыми малыми, теперь стали самыми большими.

6. РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТОВ

Все расчёты выполнялись для графов с обхватами $g = 4, 6, 8$. На графиках и в таблицах приводится общее время, затрачиваемое программами на подсчёт циклов с длинами в диапазоне $g, g + 2, \dots, 3g - 2$. В случае явных формул при заданном обхвате вычисляются значения

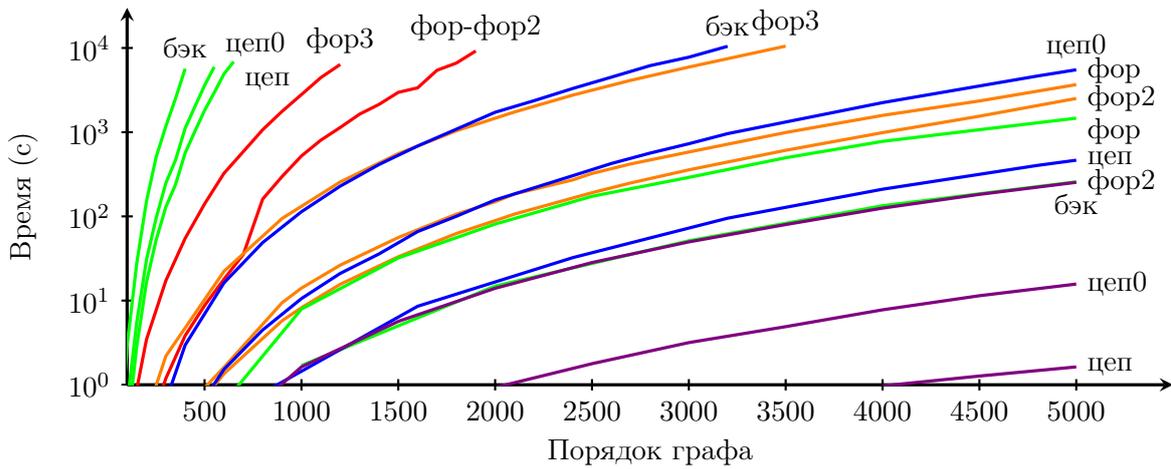


Рис. 2. Время подсчёта циклов длиной с 4 по 10 в графах с обхватом 4. Цвета соответствуют семействам графов: красный — $K_{n/2;n/2}$ ($\rho = 1$), оранжевый, зелёный, синий, фиолетовый — $R_{n/2;n/2}(4; \rho)$, $\rho = 3/4, 1/2, 1/3, 1/4$.

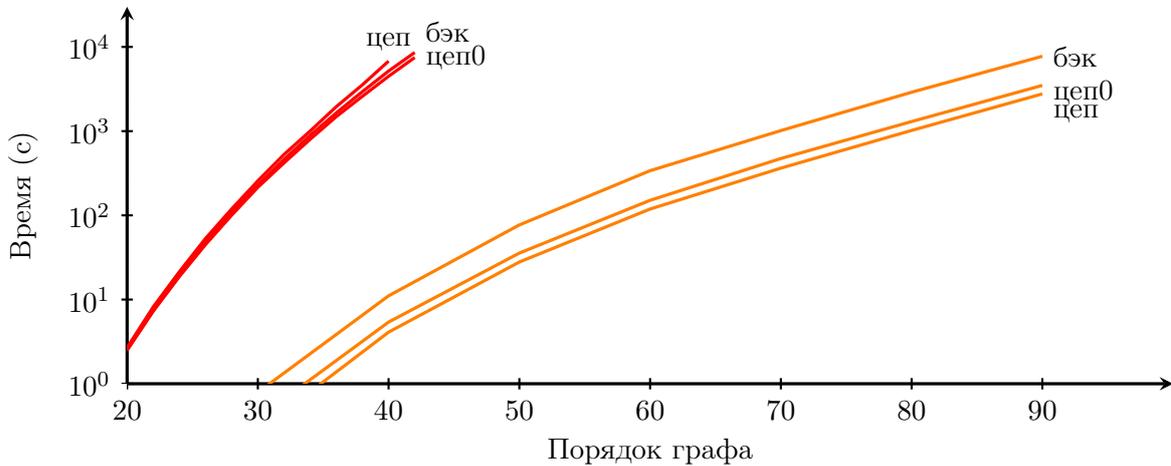


Рис. 3. Время подсчёта циклов длиной с 4 по 10 в плотных графах с обхватом 4 путём перечисления. Цвета соответствуют семействам графов: красный — $K_{n/2;n/2}$, оранжевый — $R_{n/2;n/2}(4; 3/4)$.

всех необходимых сумм, из которых затем комбинируется количество циклов. В оригинальном варианте алгоритма «цепей» строится дерево цепей высоты $3g/2 - 2$, а в улучшенном варианте — $3g/2 - 1$. Поиск с возвратом осуществляется до глубины $3g - 3$.

Для обхвата 4 были рассмотрены пять значений плотности: 1, $3/4$, $1/2$, $1/3$ и $1/4$ (рисунок 2). На графах с плотностью $1/2$ и более перечислительные алгоритмы выполняются существенно медленнее, чем явные формулы (даже с использованием типа Int128). Графики, соответствующие перечислению циклов в графах с плотностью $1/2$, расположены на рисунке 2 ближе всего к оси ординат (зелёные линии). Для больших значений плотности аналогичные кривые (оранжевого и красного цветов) почти сливались бы с вертикальной осью, поэтому они изображены отдельно на рисунке 3. Рисунок 3 не содержит линий для явных формул, поскольку при порядках графов до 100 расчёты по ним выполняются практически мгновенно (за доли секунды).

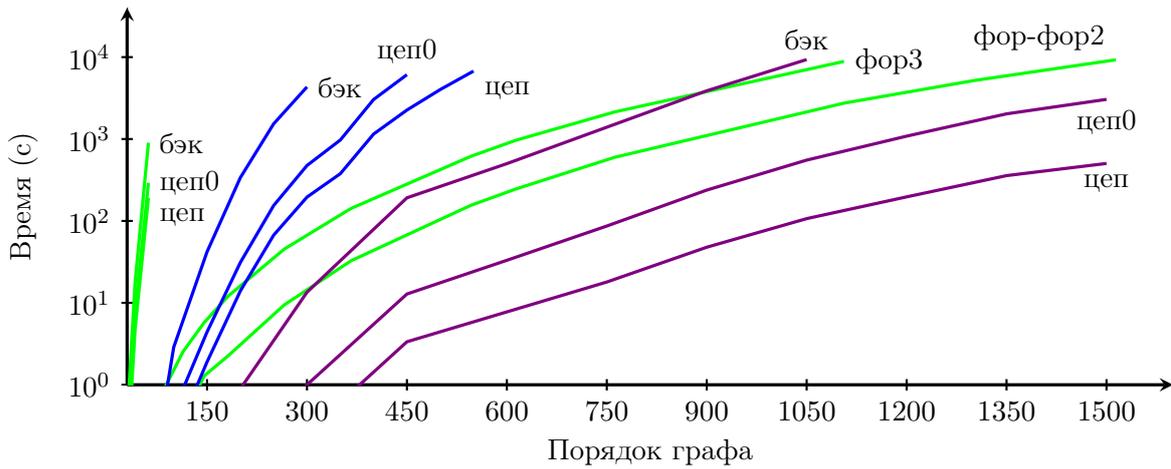


Рис. 4. Время подсчёта циклов длиной с 6 по 16 в графах с обхватом 6. Цвета соответствуют семействам графов: зелёный — $PG_2(q)$, синий — $R_{n/2;n/2}(6; 1/3)$, фиолетовый — $R_{n/2;n/2}(6; 1/4)$.

Даже для графов с плотностью $1/3$ перечислительные алгоритмы оказались медленнее явных формул. Из рисунка 2 видно, что только алгоритм «цепей» с дополнительным уровнем дерева (синяя линия «цеп») почти достиг по времени вычислений явные формулы с учётом разреженности (зелёная линия «фор2», закрытая фиолетовой линией «бэк»). Однако в случае плотности $1/4$ оба варианта алгоритма «цепей» (фиолетовые кривые) эффективнее явных выражений: оригинальный вариант — на порядок, а улучшенная версия — на два порядка.

Уменьшение плотности ниже значения $1/2$ слабо сказывается на времени расчётов по явным формулам, поэтому графики для значений $1/3$ и $1/4$ (синего и фиолетового цветов) на рисунке 2 не приводятся. Кроме того, по представленным линиям (красным, оранжевым и зелёным) можно заметить, что с уменьшением плотности увеличивается разрыв во времени вычислений по явным формулам с учётом и без учёта разреженности (при умножении на матрицу смежности). В случае большой плотности ($3/4$ и 1) основная часть времени приходится на вычисление единственной четырёхкратной суммы, несложная структура которой допускает существенную оптимизацию при уменьшении плотности. При значении плотности $1/2$ большая часть времени тратится уже на выполнение матричных умножений. В программе «фор» все 14 умножений выполняются без учёта разреженности, а в версии «фор2» — только 2.

При подсчёте циклов в графах $K_{n/2;n/2}$ и $R_{n/2;n/2}(4; 3/4)$ по явным формулам с использованием 64-битового типа переполнение возникает при $n = 152$ и $n = 622$ соответственно. Запас в 128 бит обеспечивает корректные результаты, по крайней мере, до $n = 1200$ и $n = 3500$.

В случае обхвата 6 перечислительные алгоритмы «обогнали» явные формулы также только при значении плотности $1/4$ (рисунок 4). Для явных формул (64 бит) приводится один общий график при $\rho = 1/2$ (зелёная линия). На менее плотных графах обе программы «фор» и «фор2» работают незначительно быстрее. Кроме того, сравнивая рисунки 2 и 4, можно заметить, что временные затраты на подсчёт циклов длиной до 16 в графах с обхватом 6 и плотностью $1/2$ близки ко времени, которое требуется для подсчёта циклов длиной до 10 в графах с обхватом 4 и плотностью 1, нежели $1/2$. Такая «нечувствительность» к разреженности, в отличие от рассмотренного выше случая обхвата 4, связана с тем, что в данном случае не все четырёхкратные суммы допускают эффективный учёт разреженности.

В ряде расчётов возникли переполнения. Для графов $PG_2(q)$ (наиболее плотных) 64-битовый тип переполнился при $q = 16$ ($n = 546$). Запаса в 128 бит хватило для всех выполненных расчётов (до $q = 23$, $n = 1106$).

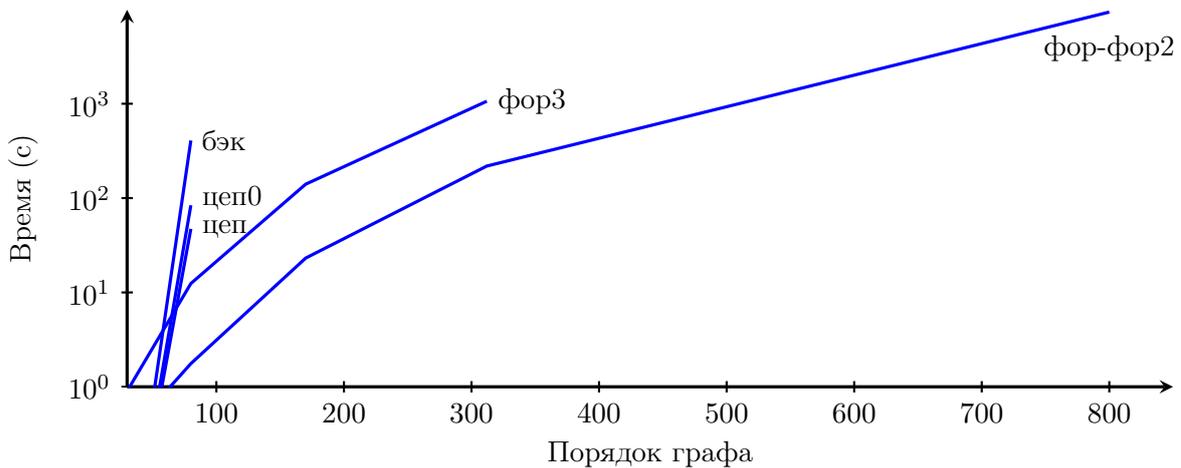


Рис. 5. Время подсчёта циклов длиной с 8 по 22 в графах $W(q)$ (с обхватом 8)

При обхвате 8, как и для рассмотренных выше значений 4 и 6, плотность $1/3$ слишком велика для того, чтобы перечислительные алгоритмы могли конкурировать с явными формулами (рисунок 5). В силу возросших длин подсчитываемых циклов (до 22), разрыв существенно больше, чем при подсчёте циклов длиной до 16 в графах той же плотности $1/3$.

Результаты подсчёта циклов в графах LDPC-кодов отчётливо демонстрируют влияние плотности на соотношение временных затрат при использовании явных формул и перечислительных алгоритмов. В таблице 2 представлены данные для нескольких пар графов одинакового порядка, но разного размера. Сближение явных выражений и перечислительных методов по

Таблица 2. Время (с) подсчёта циклов длиной с g по $3g - 2$ в графах LDPC-кодов

	n	m	ρ	фор	фор2	цеп0	цеп	бэк
96.3.967	144	288	0,32	1,4	1,3	0,5	0,2	5,0
96.44.443	144	384	0,39	1,4	1,4	122	70	1 004
120.64.3.110	184	360	0,30	3,3	3,2	0,3	0,1	5,3
204.33.484	306	612	0,28	26	24	0,8	0,2	14
204.55.153	306	1 020	0,38	25	25	11 557	5 700	98 577
273.82.3.353	355	819	0,30	73	73	64	25	978
273.82.4.2763	355	1 092	0,35	73	72	14 625	7 399	147 689
271.127.3.113	398	813	0,27	75	74	1,7	0,4	34
252.252.3.252	756	1 512	0,23	906	906	1,6	0,3	36
<i>PEGirReg252x504</i>	756	2 014	0,28	905	898	50 087	18 961	1 615 500
<i>PEGRReg252x504</i>	756	1 512	0,23	9 371	9 233	1 250	249	24 947

эффективности наблюдается при плотности от $1/4$ до $1/3$. К этой же оценке приводят и результаты расчётов на регулярных и «случайных» графах, представленные выше на рисунках 2 и 4. Хорошо заметна сильная чувствительность временных затрат на перечисление циклов: прирост плотности в пределах значения 0,1 приводит к увеличению времени расчётов на несколько порядков. В случае графов 252.252.3.252 и *PEGRReg252x504* различия во временных затратах связаны с тем, что в первом графе, имеющем обхват 6, подсчитываются циклы длиной с 6 по 16, а во втором графе, имеющем обхват 8, находится количество циклов длиной с 8 по 22.

7. ЗАКЛЮЧЕНИЕ

Расширение диапазона длин подсчитываемых циклов до $3g$ увеличивает разрыв в производительности между явными формулами и перечислительными алгоритмами в случае плотных графов. Кроме того, в случае $g = 4$ значение плотности, при котором временные затраты становятся сопоставимы, несколько меньше, чем при ограничении $2g$.

Для совсем неплотных графов с обхватами 4, 6 и 8 (плотность $1/4$ и меньше), особенно с большим количеством вершин, подсчёт циклов путём перечисления, безусловно, существенно эффективнее вычисления их количества по явным формулам. Во всех проведённых расчётах наблюдалось одно и то же соотношение перечислительных методов по эффективности: быстрее всего — версия алгоритма «цепей» с дополнительным уровнем дерева, затем — «оригинальный» вариант этого алгоритма, самый медленный — поиск с возвратом. Для первой пары алгоритмов отношение временных затрат составляло от 2 до 10, а для второй пары — от 8 до 32.

Дальнейшее существенное продвижение в реализации явных формул, как по значению обхвата, так и по длине подсчитываемых циклов, крайне затруднительно. В обоих случаях резко растёт количество форм, что осложняет сам вывод формул. Кроме того, как сообщалось в разделе 3, при расширении границы длин циклов до значения $4g$ появляются не только шестикратные, но восьмикратные и десятикратные суммы.

Перечислительные и другие известные «универсальные» методы формально применимы для произвольных значений длины цикла, однако в случае плотных графов имеют сложность существенно большего порядка по сравнению с явными выражениями. На практике такие методы нельзя рекомендовать для подсчёта циклов длиной $3g$ и более, если плотности графов становятся близки к максимальной для заданного обхвата, а количество вершин начинает исчисляться сотнями.

ПРИЛОЖЕНИЕ

А. КОЛИЧЕСТВО КОРОТКИХ ЦИКЛОВ В ТЕСТОВЫХ ГРАФАХ

Таблица 3. Количество циклов длиной 14 и 16 в графах $PG_2(q)$

q	c_{14}	c_{16}
2	24	0
3	98 280	292 734
4	9 443 520	79 969 680
5	274 350 000	4 372 627 500
7	37 917 455 184	1 379 902 150 620
8	259 733 859 840	12 809 706 989 184
9	1 403 569 477 440	89 866 540 305 360
11	24 453 527 307 000	2 415 117 876 251 850
13	260 848 919 775 600	36 666 008 152 657 884
16	4 893 705 303 552 000	1 059 325 386 621 465 600
17	11 499 152 279 503 104	2 820 539 144 493 034 560
19	55 043 809 484 073 480	16 963 353 037 140 424 470
23	807 647 229 059 412 720	367 495 260 397 947 427 284

Для чисел циклов длиной с 6 по 12 в графах $PG_2(q)$ известны явные формулы [10, 1].

Таблица 4. Количество циклов длиной с 16 по 22 в графах $W(q)$

q	c_{16}	c_{18}	c_{20}	c_{22}
2	1 620	3 840	7 524	8 280
3	2 028 240	14 129 280	92 013 408	567 492 480
4	237 945 600	3 142 579 200	40 761 011 520	522 293 040 000
5	8 961 030 000	190 535 280 000	4 042 665 900 000	85 691 913 840 000
7	2 029 966 747 200	86 737 580 966 400	3 733 813 080 224 640	161 679 613 678 732 800

Таблица 5. Количество циклов длиной с 6 по 16 в графах LDPC-кодов с обхватом 6

	c_6	c_8	c_{10}	c_{12}	c_{14}	c_{16}
96.3.967	133	1 553	9 518	84 211	634 620	5 179 488
96.44.443	1 815	25 230	407 831	6 648 871	108 620 083	1 766 740 975
120.64.3.110	129	978	7 283	54 101	408 084	3 107 706
204.33.484	162	1 271	10 196	82 084	681 340	5 723 309
204.55.153	8 524	214 348	6 050 187	175 411 969	5 168 067 490	153 621 448 624
273.82.3.353	1 017	13 293	188 778	2 731 565	40 194 050	596 211 217
273.82.4.2763	9 240	239 727	6 934 646	205 458 616	6 179 501 101	187 126 882 342
271.127.3.113	213	1 750	15 386	136 013	1 233 724	11 333 801
252.252.3.252	169	1 312	10 052	83 007	699 526	6 024 212
<i>PEGirReg252x504</i>	13 244	420 609	13 567 791	465 487 228	16 020 264 939	557 644 424 788

Количество циклов длиной с 8 по 14 в графах $W(q)$ можно найти в [1].

В графе *PEGirReg252x504* (обхват 8) количество циклов длиной 8, 10, ..., 22 равно 802, 11 279, 86 791, 723 426, 6 271 645, 55 013 858, 485 849 376, 4 319 577 317.

СПИСОК ЛИТЕРАТУРЫ

1. Воропаев А.Н. Учёт обхвата при подсчёте коротких циклов в двудольных графах. *Информационные процессы*, 2011, том 11, № 2, стр. 225–252. <http://www.jip.ru/2011/225-252-2011.pdf>.
2. Tarjan R. Enumeration of the Elementary Circuits of a Directed Graph. *SIAM Journal on Computing*, 1973, vol. 2, no. 3, pp. 211–216.
3. Nguyen D.V., Chilappagari S.K., Marcellin M.W., Vasic B.V. LDPC Codes from Latin Squares Free of Small Trapping Sets. *CoRR*, 2010, vol. abs/1008.4177, <http://arxiv.org/abs/1008.4177>.
4. Харари Ф. *Теория графов*. М.: Мир, 1973. (Harary F. *Graph Theory*. Reading: Addison-Wesley, 1969.)
5. Воропаев А.Н. Вывод явных формул для подсчёта циклов фиксированной длины в неориентированных графах. *Информационные процессы*, 2011, том 11, № 1, стр. 90–113. <http://www.jip.ru/2011/90-113-2011.pdf>.
6. The nauty page. <http://cs.anu.edu.au/~bdm/nauty>.
7. Encyclopedia of Sparse Graph Codes. <http://www.inference.phy.cam.ac.uk/mackay/codes/data.html>.
8. Караваев А.М., Воропаев А.Н. Эффективность распараллеливания явных формул для подсчёта коротких циклов в графе. *Международная научная конференция «Параллельные вычислительные технологии» (ПАВТ'2010)*. Челябинск: Изд. центр ЮУрГУ, 2010, стр. 486–497.
9. Explicit Formulae: FlowProblem. <http://flowproblem.ru/cycles/explicit-formulae>.
10. Lazebnik F., Mellinger K.E., Vega O. On the Number of k -gons in Finite Projective Planes. *Note di Matematica*, 2009, vol. 29, no. 1, pp. 135–152.