Задача об аддитивных цепочках и ее обобщения

С. Б. Гашков

В последние несколько десятков лет возрос интерес к алгоритмической стороне математики. Он проявляется, в частности, в постановке задач теоретико-сложностного характера в разнообразных областях, начиная с классического анализа и алгебры (например, [2,7,9,10,13,16,17,20–23,25]). Эти задачи отличаются друг от друга как по постановке, так и по методам, однако их большой круг объединяется возможностью формулировки в терминах схемной сложности вычисления функций.

1. Аддитивные цепочки и возведение в степень

Самой старой из относящихся к этой тематике задач является задача об аддитивных цепочках, рассматривавшаяся в тридцатые годы А. Шольцем и А. Брауэром. Идеи, ведущие к аддитивным цепочкам, в [9] прослежены до времен древних Египта и Индии. В последние двадцать лет аддитивные цепочки нашли применение в криптографических алгоритмах (см., например, [15]) и число публикаций на эту тему возросло настолько, что требует отдельного обзора (такой обзор уже написан [18]). Мы ограничимся некоторыми примерами.

Аддитивные цепочки возникают, например, в следующей олимпиадной задаче.

За какое наименьшее количество взвешиваний на чашечных весах можно отвесить один килограмм сахарного песку, если имеется лишь одна однограммовая гирька?

На первый взгляд кажется, что единственный способ решения этой задачи — отвесить один грамм, положить в эту же чашку гирьку, отвесить в другой чашке два грамма, переложить гирьку в нее, и так далее, добавляя по одному грамму, после тысячного взвешивания отмерить наконец-то килограмм.

Но есть и более быстрый способ. Нужно лишь заметить, что если мы научились отвешивать за k взвешиваний m грамм, то, сделав еще одно взвешивание, можно, даже не используя гирьку, отвесить еще m грамм и, ссыпав обе порции вместе, получить $2m$ грамм за $k+1$ взвешивание. А если при этом взвешивании положить на одну из чашек гирьку, то за $k+1$ взвешивание можно отвесить $2m \pm 1$ грамм.

Если нужно отмерить n грамм, то можно записать n в двоичном виде $(a_m \dots a_1)$, где $2^{m-1} \leqslant n < 2^m$, $a_m = 1$ и воспользоваться формулой

$$n = a_m 2^{m-1} + \dots + a_2 2 + a_1 = (\dots((2a_m + a_{m-1})2 + a_{m-2})\dots)2 + a_1,$$

последовательно отвешивая по

$$b_1 = a_m,\ b_2 = 2b_1 + a_{m-1},\ b_3 = 2b_2 + a_{m-2},\ \dots,\ b_m = b_{m-1}2 + a_1 = n$$

граммов (можно использовать и двоичную запись с отрицательными цифрами).

В используемой формуле читатели увидят схему Горнера. Она будет встречаться у нас и далее.

Идея, лежащая в основе этого метода взвешивания, стара, как сама математика. Ее применяли и древние египтяне, и древние индусы, но конечно, не для взвешивания, а для умножения — алгоритм умножения столбиком был придуман не сразу. А до этого умножение сводилось к сложению и удвоению. Такой метод умножения дожил почти до нашего времени, он удобен при вычислениях на счетах. Сейчас он никому не нужен — счеты вытеснены калькуляторами. Но как возвести на калькуляторе число a в тысячную степень, если у него нет специальной операции возведения в степень? Умножать 999 раз не нужно, а можно применить совершенно тот же прием, последовательно вычисляя

$$a^3 = a^2 a,\ a^7 = (a^3)^2 a,\ a^{15} = (a^7)^2 a,\ a^{31} = (a^{15})^2 a,$$
$$a^{62} = (a^{31})^2,\ a^{125} = (a^{62})^2 a, a^{250} = (a^{125})^2,\ a^{500} = (a^{250})^2,\ a^{1000} = (a^{500})^2.$$

Если вспомнить, что 1000 имеет двоичную запись 1111101000, то можно заметить, что если отбросить старший бит (равный единице), то каждому следующему биту соответствует операция возведения в квадрат, если он нулевой, или возведение в квадрат с последующим умножением на число a— основание степени. Число a не нужно каждый раз заново набирать на клавиатуре. Можно в начале вычислений занести его в память, и когда нужно, после нажатия кнопки для умножения, вызывать его из памяти и потом нажимать кнопку «равно». Посчитаем общее число операций умножения в рассмотренном вычислении. Число возведений в квадрат на единицу меньше длины двоичной записи показателя степени, а число умножений общего вида на единицу меньше суммы цифр двоичной записи.

Для любого n обозначим $\lambda(n)$ уменьшенную на единицу длину двоичной записи числа n, а $\nu(n)$ — ее сумму цифр (т. е. число единиц в ней). Тогда в общем случае число операций умножения, использованных в этом методе[1)] возведения в степень n, равно $\lambda(n) + \nu(n) - 1$.

[1)] Он называется *бинарным*.

Покажем, что меньшим числом операций обойтись нельзя, если только не обновлять содержимое ячейки памяти. Для удобства будем рассматривать изменение не самих степеней, а их показателей, которые после каждой операции будут складываться. Обозначим показатель степени у числа, находящегося в регистре, через a_i (в начальный момент $a_0 = 1$), тогда изменяться он может одним из двух следующих способов: $a_{i+1} = 2a_i$ или $a_{i+1} = a_i + 1$ (в первом случае содержимое регистра возводится в квадрат, во втором случае оно умножается на содержимое ячейки памяти, а результат опять записывается в регистр). Очевидно, что в первом случае $\nu(a_{i+1}) = \nu(a_i)$, $\lambda(a_{i+1}) = \lambda(a_i) + 1$, откуда

$$\nu(a_{i+1}) + \lambda(a_{i+1}) = \nu(a_i) + \lambda(a_i) + 1.$$

Во втором случае $\nu(a_{i+1}) \leqslant \nu(a_i) + 1$ (равенство возможно, только если a_i четно), $\lambda(a_{i+1}) \leqslant \lambda(a_i) + 1$ (равенство возможно, только если a_i нечетно, точнее, когда $a_i = 2^{\lambda(a_i)} - 1$), поэтому одновременно эти равенства не выполняются; значит, во втором случае

$$\nu(a_{i+1}) + \lambda(a_{i+1}) \leqslant \nu(a_i) + \lambda(a_i) + 1,$$

причем равенство возможно лишь когда a_i четно или $a_i = 1$. Так как $\lambda(1) + \nu(1) = 1$, то отсюда с помощью индукции выводится, что $\nu(a_l) + \lambda(a_l) \leqslant l+1$, причем равенство возможно только когда прибавление единицы всегда выполняется после одного или нескольких удвоений (можно считать, что на первом шаге всегда происходит удвоение), при этом после каждого прибавления единицы $\nu(a_i)$ увеличивается на единицу, значит $\nu(a_l) - 1$ равно числу выполненных при вычислении a_l прибавлений единицы. Значит,

$$\nu(n) + \lambda(n) = \nu(a_{l(n)}) + \lambda(a_{l(n)}) \leqslant l(n) + 1,$$

где $l(n)$ число операций умножения для возведения в n-ю степень. Поэтому $l(n) \geqslant \nu(n) + \lambda(n) - 1$. Аналогично можно показать, что число операций умножения регистра на постоянное число из памяти не меньше $\nu(n) - 1$. Далее, равенство $l(n) = \nu(n) + \lambda(n) - 1$ возможно лишь когда прибавление единицы всегда выполняется после одного или нескольких удвоений, при этом число прибавлений единицы равно $\nu(n) - 1$ и n представимо в виде

$$n = 2^{\lambda(n)} + 2^{\alpha_{\nu(n)-1}} + \ldots + 2^{\alpha_1},$$

где α_i строго возрастают. Такое представление возможно только одно, и поэтому существует только один способ возведения в n-ю степень в указанных условиях — указанный выше бинарный метод.

Но если разрешается обновлять содержимое ячейки памяти (используя для этого пересылки из других ячеек), то бинарный метод вычисления x^n в некоторых случаях можно улучшить. Для этого, например, можно применить *метод множителей*. Его идея заключается в следующем. Если

мы умеем возводить в степень n за $l(n)$ операций, и возводить в степень m за $l(m)$ операций, то можно, после того как закончено вычисление x^n, занести его в ячейку памяти, и далее вычислить $x^{nm} = (x^n)^m$ за $l(m)$ операций, используя тот же метод, что и для вычисления x^m. Тогда общее число операций будет равно $l(nm) = l(n) + l(m)$.

Например, вычисляя x^5 бинарным методом за 3 операции и применяя два раза метод множителей, получаем, что $l(125) = 3l(5) = 9$. Выполняя еще три возведения в квадрат, имеем $l(1000) = l(125) + 3 = 12$. Бинарный метод требовал $\lambda(1000) + \nu(1000) - 1 = 9 + 6 - 1 = 14$ операций.

Заметим еще, что существует и другой вариант бинарного метода, с тем же числом возведений в квадрат и с тем же общим числом умножений, но для его выполнения требуется большое число ячеек памяти.

Что же такое *аддитивная цепочка*? Это любая, начинающаяся с 1, последовательность натуральных чисел $a_0 = 1, a_1, \ldots, a_m$, в которой каждое число является суммой каких-то двух предыдущих чисел (или удвоением какого-то предыдущего числа). Обозначим $l(n)$ наименьшую длину аддитивной цепочки, заканчивающейся числом n. Длиной цепочки $a_0 = 1, a_1, \ldots, a_m$ называем число m. Например, $1, 2, 3, 5, 7, 14$ — минимальная цепочка для 14, т. е. $l(14) = 5$.

Аддитивные цепочки можно изображать в виде ориентированного графа, в котором в вершину a_i идут рёбра от вершин a_j, a_k, если $a_i = a_j + a_k$ (если такое представление неоднозначно, выбираем любое из них и рисуем только два ребра). Можно считать, что все числа в цепочке разные, просто удаляя из нее повторяющиеся числа, и располагать числа в цепочке в порядке возрастания.

Граф для предыдущего примера см. на рис. 1.

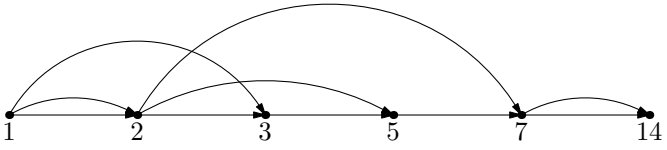

Рис. 1.

Очевидно, что наименьшее число умножений, необходимое для возведения в n-ю степень, равно $l(n)$.

Бинарный метод дает оценку $l(n) \leqslant \lambda(n) + \nu(n) - 1$. Методом множителей — оценку $l(nm) \leqslant l(n) + l(m)$. Справедлива и нижняя оценка $l(n) \geqslant \lambda(n)$. Из нее следует, что $l(2^n) = n$. Для доказательства оценки $l(n) \geqslant \lambda(n)$ достаточно заметить, что для любой аддитивной цепочки справедливы неравенства $a_i \leqslant 2^i$, в чем можно убедится, проведя индукцию.

База индукции очевидна, а индукционный шаг обосновывается неравенством

$$a_i = a_j + a_k \leqslant 2^j + 2^k \leqslant 2^{i-1} + 2^{i-1} = 2^i.$$

Из неравенства $n = a_{l(n)} \leqslant 2^{l(n)}$ следует, что $l(n) \geqslant \lceil \log_2 n \rceil = \lambda(n)$, символ $\lceil x \rceil$ в в последней формуле означает наименьшее целое число, не меньшее x. Более тонкие нижние оценки читатель может найти [9] (вместе с массой другой информации об аддитивных цепочках и не только о них), но они доказываются непросто, а по своей точности ненамного превосходят доказанную почти очевидную оценку.

Интересно, что бинарный метод был по существу известен древним индусам, потом был переоткрыт арабскими математиками, но задача о точном вычислении функции $l(n)$ появилась (согласно [9]) в одном французском журнале в 1894 г., потом заново была переоткрыта в тридцатые годы в Германии, и неоднократно переоткрывалась в дальнейшем[2)], но до сих пор в общем случае не решена. Не известно даже, существует ли алгоритм полиномиальной сложности[3)] для вычисления функции $l(n)$. Не решены также многие другие задачи об аддитивных цепочках. Например, неизвестно, верно ли равенство

$$l(2^n - 1) = n + l(n) - 1$$

(гипотеза Шольца). Некоторые естественные гипотезы об аддитивных цепочках оказались неверны. За информацией об всем этом мы отсылаем читателя к [9] — единственной пока монографии на русском языке, где есть раздел, посвященный аддитивным цепочкам.

Наилучшая из общих верхних оценок была доказана в тридцатые годы А. Брауэром и имеет вид

$$\lambda(n)\left(1 + \frac{1}{\lambda(\lambda(n))} + \frac{O(\lambda(\lambda(\lambda(n))))}{(\lambda(\lambda(n)))^2}\right).$$

Она вытекает из следующей теоремы, если в ней положить $k = \lambda(\lambda(n)) - 2\lambda(\lambda(\lambda(n)))$.

Теорема 1 (А. Брауэр). *При $k < \log_2 \log_2 n$ справедливо неравенство*

$$l(n) < (1 + 1/k)\lceil \log_2 n \rceil + 2^{k-1} - k + 2.$$

Доказательство. Представим n в двоичной записи

$$n = \sum_{i=0}^{m} \alpha_i 2^i,$$

[2)] Например, на рубеже 50-х и 60-х годов двадцатого века в Москве Р. Вальским.

[3)] Это означает, что время работы алгоритма ограничено некоторым полиномом от $\log n$.

Рис. 2. Альфред Брауэр

где $\alpha_i = 0$ или 1, $m = \lfloor \log_2 n \rfloor$. Выделим в наборе $(\alpha_0, \ldots, \alpha_m)$ не более чем $\lceil \frac{m+1}{k} \rceil$ непересекающихся блоков $A_0, \ldots, A_s$, $s < \lceil \frac{m+1}{k} \rceil$, следующим образом. Каждый блок состоит из не более чем из k подряд идущих цифр и заканчивается единицей, кроме блока A_s, состоящего из старших k цифр, а вне блоков стоят только нули. Числа $a_0, \ldots, a_s$, двоичными записями которых являются эти блоки, не превосходят $2^k - 1$ и все нечетны, кроме возможно a_s. Тогда n можно представить в виде

$$n = 2^{l_0}\left(2^{l_1}\ldots\left(2^{l_{s-2}}\left(2^{l_{s-1}}\left(2^{l_s}a_s + a_{s-1}\right) + a_{s-2}\right) + \ldots + a_1\right) + a_0\right),$$

где $l_s + l_{s-1} + \ldots + l_0 = m + 1 - k$. Все числа $a_0, \ldots, a_{s-1}$ содержатся в аддитивной цепочке $1, 2, 3, 5, 7, \ldots, 2^k - 1$ длины $2^{k-1} + 1$. Если число a_s не содержится в этой цепочке, то его можно поставить в ее конец, вычислив как $a_s = (a_s - 1) + 1$. Поэтому для вычисления n достаточно добавить к этой цепочке последовательность

$$a_s, 2a_s, 4a_s, \ldots, 2^{l_s}a_s, 2^{l_s}a_s + a_{s-1},$$
$$2\left(2^{l_s}a_s + a_{s-1}\right), 4\left(2^{l_s}a_s + a_{s-1}\right), \ldots,$$
$$2^{l_{s-1}}\left(2^{l_s}a_s + a_{s-1}\right), 2^{l_{s-1}}\left(2^{l_s}a_s + a_{s-1}\right) + a_{s-2}, \ldots,$$
$$2^{l_{s-2}}\left(2^{l_{s-1}}\left(2^{l_s}a_s + a_{s-1}\right) + a_{s-2}\right),$$

. . . . . .

$$2^{l_1}\left(\ldots 2^{l_{s-2}}\left(2^{l_{s-1}}\left(2^{l_s}a_s + a_{s-1}\right) + a_{s-2}\right) + \ldots + a_1\right) + a_0, \ldots,$$
$$2^{l_0}\left(2^{l_1}\left(\ldots 2^{l_{s-2}}\left(2^{l_{s-1}}\left(2^{l_s}a_s + a_{s-1}\right) + a_{s-2}\right) + \ldots + a_1\right) + a_0\right),$$

длина которой равна

$$l_s + l_{s-1} + \ldots + l_0 + s + 1 = m + 2 + s - k.$$

Поэтому

$$l(n) < 2^{k-1} + 1 + m + 2 + s - k < m + 2 + \left\lceil \frac{m+1}{k} \right\rceil + 2^{k-1} - k.$$

Можно считать, что $n \neq 2^m$, тогда $m + 1 = \lceil \log_2 n \rceil$ и

$$l(n) < \lceil \log_2 n \rceil (1 + 1/k) + 2^{k-1} - k + 2.$$

Понятие аддитивной цепочки имеет некоторые естественные обобщения. Например, изучались цепочки с вычитаниями. Можно рассматривать цепочки с различными ограничениями, например цепочки, в которых запрещены удвоения, или цепочки, в которых разрешается сложения только следующего типа $a_{i+1} = a_i + a_k$, но удвоения разрешены. Такие цепочки называются линейными, о них в [9] доказаны интересные теоремы. Можно рассматривать также различные меры сложности аддитивных цепочек, отличные от их длины. Одно из естественных обобщений аддитивных цепочек будет рассмотрено в следующем разделе.

2. Сложность вычисления сумм

Введем следующее определение. *Векторная аддитивная цепочка* — это последовательность векторов, начинающаяся с единичных базисных векторов $(1, 0, \ldots, 0)$, $\ldots$, $(0, \ldots, 0, 1)$, в которой каждый следующий вектор получается сложением двух предыдущих векторов (не обязательно непосредственно предшествующих), или удвоением какого-то предыдущего вектора (т. е. прибавлением его к самому себе). Сложностью системы векторов называется длина (без учета базисных векторов) кратчайшей цепочки, содержащей эту систему (если в системе некоторые вектора совпадают, то требуется,чтобы в цепочке содержались все различные вектора, встречающиеся в системе). Обозначим $L(p, q, N)$ наибольшую сложность системы, состоящих из p векторов размерности q, компоненты которых принадлежат множеству $0, 1, \ldots, N-1$. Эта величина совпадает с наибольшей *аддитивной сложностью* систем p линейных форм от q переменных с натуральными коэффициентами, меньшими N, и с наибольшей *мультипликативной сложностью* систем из p одночленов от q переменных,входящих в них в степенях, меньших N. Под аддитивной и мультипликативной сложностью здесь понимается сложность реализации соответствующей вектор-функции схемами из функциональных элементов в базисах $\{x + y\}$ и $\{x \cdot y\}$ соответственно (см. соответствующие определения в следующем разделе).

Н. Пиппенджер [24] показал, что при $w < N^{o(v)}$, где $w = \max(p, q)$, $v = \min(p, q)$,

$$L(p, q, N) = v \log N + \frac{H}{\log H}\left(1 + O\left(\frac{\log\log H}{\log H}\right)^{1/2}\right) + O(w),$$

где $H = pq \log N$. Отсюда следует, в частности, довольно неожиданная теорема Яо о том, что длина кратчайшей аддитивной цепочки, содержащей несколько заданных чисел, асимптотически равна длине кратчайшей цепочки, вычисляющей наибольшее из них, если только этих чисел не слишком много (см. [9]).

Оценки Пиппенджера в [4] дополнены и уточнены следующим образом: $L(p, q, N) + p = L(q, p, N) + q$, если $w \geqslant N^v - v - 1$, то $L(p, q, N) = N^v - v - 1 + (w - p)$, если $w < N^v - v - 1$, то

$$L(p, q, N) = v \log N + \frac{H}{\log H}\left(1 + O\left(\frac{\log\log H}{\log H}\right)^{1/2}\right) + \left(2 + \frac{1}{\log H}\right) w - p.$$

причем при $v \log N \geqslant \dfrac{H \log\log H}{\log^2 H}$ справедливо равенство

$$L(p, q, N) = v \log N + \frac{H}{\log H}\left(1 + O\left(\frac{\log\log H}{\log H}\right)\right),$$

если $w < N^v$, то

$$p + L(p, q, N) \leqslant \frac{H}{\log H}\left(1 + O\left(\frac{\log\log H}{\log H}\right)^{1/2}\right) + (2w + v \log N)\left(2 + \frac{1}{\log H}\right).$$

Частным случаем рассмотренной задачи является задача вычисления системы степеней одной переменной $x^{n_1}, \ldots, x^{n_p}$, поставленная Д. Кнутом [9]. Обозначим $l(n_1, \ldots, n_p)$ соответствующую сложность вычисления системы линейных форм от одной переменной. Положим $N = \max n_i$, $H = \log(n_1 \ldots n_p)$. Тогда приведенные выше результаты можно уточнить следующим образом [4]:

$$l(n_1, \ldots, n_p) \leqslant \log N + \frac{H}{\log H}\left(1 + O\left(\frac{\log\log H}{\log H}\right)^{1/2}\right) + O(p).$$

В книге для подготовки к математическим олимпиадам [12] в 13-й главе имеется подборка задач «Сложность суммирования». В ней идет речь о сложности вычисления линейных преобразований с булевыми матрицами над полем $\{0, 1\}$ в базисе, состоящем из одной операции $x \oplus y$ — сложения по модулю два. Повторять формулировки этих задач вряд ли здесь

уместно, но имеет смысл привести формулировки некоторых более общих утверждений из [4]. Величина $L(p,q,N)$ далее та же, что и выше.

Справедливы неравенства

$$L(p,q,N) \leqslant L(p,q\lceil\log_2 N\rceil,2),$$

и для любого $s \leqslant q$

$$L(p,q,N) \leqslant (s-1)p + sN^{\lceil q/s\rceil}.$$

Второе неравенство доказывается применением вентильной конструкции Лупанова [11]. Формально этого неравенства в [11] нет, там есть подобные неравенства для сложности вентильных схем, но из оценки для вентильных схем вытекает оценка для рассматриваемых схем в силу почти очевидного неравенства $C(p,q,N) \geqslant L(p,q,N) + p$, и конструкция для вентильных схем без существенных изменений переносится и на рассматриваемые нами схемы. Чтобы не отвлекаться в сторону, мы не будем здесь рассматривать вентильные схемы. В чем состоит эта конструкция, можно понять, посмотрев в [4] или [11], но можно и прочитать решения соответствующих задач из [12]. Строго говоря в [11] рассматривался случай $N = 2$ и рассматривались, так сказать, дизъюнктивные вентильные схемы, а не вентильные схемы по модулю два, которые более подходят к рассматриваемой ситуации, но принципиального значения это не имеет. В случае $N = 2$ из указанного неравенства при выборе $s = q/(\log_2 p - \log_2\log_2 p) + O(1)$ имеем

$$L(p,q,2) \leqslant \frac{pq}{\log_2 p}\left(1 + \frac{O(\log_2\log_2 p)}{\log_2 p}\right).$$

Если $q > p$, то в силу равенства $L(p,q,2)+p = L(q,p,2)+q$ это неравенство можно усилить до

$$L(p,q,2) \leqslant \frac{pq}{\log_2 q}\left(1 + \frac{O(\log_2\log_2 q)}{\log_2 q}\right).$$

В общем случае, если положить $w = \max(p,q), v = \min(p,q)$, то

$$L(p,q,2) \leqslant \frac{wv}{\log_2 w}\left(1 + \frac{O(\log_2\log_2 w)}{\log_2 w}\right).$$

Заметим, что из упомянутой выше теоремы Пиппенджера следует асимптотически точный результат

$$L(p,q,2) = v + \frac{H}{\log H}\left(1 + O\left(\frac{\log\log H}{\log H}\right)^{1/2}\right) + O(w),$$

где $H = pq$.

2.1. Лемма о транспонировании матриц

Простейший вариант этой леммы содержится в формулировке задачи 16 из цикла «Сложность суммирования» 13-й главы книги [12]. Сформулировать ее можно следующим образом.

Пусть A — матрица из нулей и единиц, все m строк и n столбцов которой ненулевые. Обозначим $L(A)$ сложность вычисления определяемого этой матрицей линейного преобразования AX в базисе $\{x+y\}$. Тогда $L(A) + m = L(A^T) + n$, где A^T — транспонированная матрица.

История многократно переоткрывавшейся леммы о транспонировании[4)] восходит, как утверждают французские авторы, к работам Теллегена об электрических цепях. Исторический обзор имеется в [19]. В нем выражается сомнение в правомерности приписывания чести открытия этой леммы Теллегену, и указывает, что вероятно, она принадлежит Фидуччиа (1973 г.), который ее распространил и на схемы для билинейных преобразований. Разные варианты леммы о транспозиции и некоторые их применения были указаны в 1980 – 1981 годах в работах Оливоса и Кнута – Пападимитриу, изложение которых можно найти в [9]. В 1981 году лемму о транспозиции получил также в [14] А. Ф. Сидоренко.

Укажем некоторые применения леммы о транспозиции (имеющиеся в [4, 12]). Первое заключается в выводе равенства $L(p, q, N) + p = L(q, p, N) + q$ с помощью которого в [4] получены некоторые уточнения результатов Пиппенждера [24], причем так называемый «трудный случай» теоремы Пиппенджера сводится к несложно доказываемой методом Брауэра оценке Страуса (см. [4] или [9])

$$L(p, q, N) \leqslant p\left(1 + \frac{q}{t}\right)\lceil \log_2 N\rceil + 2^t q + pq.$$

Та же идея позволяет легко доказать упоминавшуюся выше теорему Яо. Фактически так же, но без явного применения леммы о транспозиции, теорема Яо доказана в [9].

Второе применение леммы о транспозиции, указанное в [4], связано с выводом равенства $L(B_n) = 2^{n+1} - 2(n+1)$, где B_n есть $(n, 2^n - 1)$-матрица, столбцы которой образованы всевозможными различными наборами 0 и 1 длины n. Для транспонированной матрицы B_n^T равенство $L(B_n^T) = 2^n - n - 1$ очевидно. Нужное нам равенство следует из него с помощью леммы о транспозиции. Матрица B_n интересна тем, что она определяет матрицу оператора поиска ошибки в коде Хэмминга (если ее рассматривать над полем из двух элементов). Если из из матрицы B_n выбросить все столбцы с одной единицей, то получится матрица кодирования для кода Хэмминга. Ее сложность находится аналогично, и она равна $2^{n+1} - 3n - 2$.

[4)] И автор этой статьи тоже ее переоткрыл в 1985 г.

3. Схемная сложность вычисления функций

Здесь будут дано общее определение понятия схемной сложности и указаны некоторые конкретные примеры.

Базисом называется произвольное множество B функций (операций) $\omega : E^n \to E$, чьи аргументы и значения принадлежат любому заданному множеству E (содержащему хотя бы два элемента).

Схемой в базисе B называется произвольная последовательность функций $f_1(X), \dots, f_{L+n}(X)$, $X = (x_1, \dots, x_n)$, в которой первые n функций определяются равенствами $f_i(X) = x_i$, а каждая следующая функция $f_l(X)$ вычисляется через некоторые из предыдущих с помощью одной из базисных операций w_k, $k = k(l)$. Схемы иногда называют *неветвящимися программами*. *Сложность* схемы — это число L. Функция f *реализуется* схемой S, если f равна какой-то из функций f_i схемы S. *Сложностью* (*схемной реализации*) функции f назовем число $L_B(f)$, равное наименьшей из сложностей схем, реализующих f. Все введенные определения естественно распространяются на случай реализации *вектор-функций* (*операторов* $E^n \to E^m$). Можно также перенести эти определения на случай базисов с произвольными неотрицательными весами элементов. Сложностью схемы тогда называется сумма весов, входящих в схему базисных элементов.

Случай $E = \{0, 1\}$ относится к алгебре логики, где впервые в массовом количестве появились задачи о сложности вычисления функций (см., например, [10]). Функции, о которых в этом случае идет речь, называются булевыми. В случае $E = \{0, 1, \dots, k-1\}$ при $k > 2$ речь идет о сложности вычисления функций многозначной логики.

В случае $E = [0, 1]$ или $E = \mathbb{R}$ можно рассматривать и базисы, состоящие из непрерывных функций. Если в качестве B взять $\{x-y, x+y, xy\} \cup \mathbb{R}$, то в терминах сложности схем в этом базисе можно сформулировать многие результаты алгебраической теории сложности, например результаты о сложности вычисления многочленов, в том числе и об аддитивной и мультипликативной сложности, о сложности так называемых параллельных вычислений, о сложности умножения матриц и т. д. (см., например, [2, 9, 20, 22, 23]).

Очевидно, что понятие схемы в случае базиса, состоящего из операции сложения и единицы, превращается в понятие аддитивной цепочки.

Приведем примеры задач, на первый взгляд не связанных с рассматриваемой постановкой, но которые допускают переформулировку в ее терминах. Некоторые из этих задач фактически были известны довольно давно.

Таковой является известная в фольклоре задача о построении последовательно-параллельных схем (так называемых П-схем) из единичных резисторов, имеющих заданное сопротивление и содержащих минимально

возможное их число. Эта задача сводится к задаче о вычислении меры сложности $L_B(r)$ для $r \in \mathbb{Q}_+$ и $B = \{x + y, 1/(1/x + 1/y)\}$. Распространенное мнение, будто она легко решается разложением r в цепную дробь, неверно; надо использовать так называемые *ветвящиеся цепные дроби*, и все же явно вычислить эту меру сложности не удается. Интересные результаты по этой задаче получил О. М. Касим-Заде [8] (см. также [3, 5]).

Другим примером является задача о нахождении минимального числа прямых и окружностей, которые надо провести циркулем и линейкой (или только одним циркулем, как в построениях Мора – Маскерони), чтобы выполнить данное планиметрическое построение. Эта задача была поставлена в 19-м веке Лемуаном, но в определенном смысле ее история начинается в Древней Греции. При некоторых естественных ограничениях на проводимые построения она сводится к вычислению (с точностью до порядка) меры сложности $L_B(F)$ вектор-функции F в базисе $B = \{x \pm y, xy, x/y, \sqrt{x}\}$. Например, если задан единичный отрезок, и надо построить одну точку, то в качестве F можно взять вектор, составленный из двух положительных констант. Если положить веса рациональных операций в базисе B равными нулю, то получим задачу о так называемой иррациональной сложности[5)].

В заключение рассмотрим вопрос о сложности геометрических построений более подробно.

4. Сложность геометрических построений

Допустим, что на плоскости дано некоторое множество точек, прямых и окружностей, которое обозначим M_0. Назовем *построением циркулем и линейкой* при заданном M_0 любую последовательность множеств M_0, $M_1, \ldots, M_L$, начинающуюся с M_0, и такую, что каждое следующее множество M_{i+1} получается из предыдущего множества M_i добавлением либо некоторой прямой, проходящей через какие-то две точки из множества M_i, либо окружности с центром в какой-то из точек множества M_i и радиусом, равным длине некоторого отрезка с концами в точках из M_i, а также всех точек пересечения добавленной линии со всеми линиями из множества M_i. Число L назовем *сложностью этого построения. Сложностью построения множества* M точек, отрезков и окружностей и прямых при заданном M_0 назовем минимальную сложность такого построения M_0, $M_1, \ldots, M_L$, для которого множество M_L содержит все прямые и окружности из M,

[5)] Однако высказанное в [6] утверждение о тождестве иррациональной сложности и наименьшего числа применений циркуля в планиметрическом построении неверно, так как согласно теореме Штейнера любое построение можно выполнить одной линейкой, если задана окружность с центром.

все точки из M и концы всех отрезков из M. Аналогично определяется *сложность построения одним циркулем.*

Большая подборка задач о сложности геометрических построений приведена в [5]. Далее приводятся некоторые задачи из [5].

Задача 1. Пусть дан единичный отрезок (точнее, даны только его концы). Для любого натурального n можно построить одним циркулем отрезки длины n и $1/n$ сложностью не более

$$3\log_3 n + \frac{\log_3 n}{\log_3\log_3 n}\left(1 + \frac{2\log_3\log_3\log_3 n + O(1)}{\log_3\log_3 n}\right).$$

Сложность построения отрезка длины n одним только циркулем не меньше $\lfloor \log_\varphi (n+1/2)\rfloor$, где $\varphi = \dfrac{\sqrt{5}+1}{2}$. Можно разделить отрезок на n равных частей одним циркулем со сложностью не больше

$$n + 3\log_3 n + \frac{\log_3 n}{\log_3\log_3 n}\left(1 + \frac{2\log_3\log_3\log_3 n + O(1)}{\log_3\log_3 n}\right).$$

Сложность разделения отрезка одним циркулем на n равных частей не меньше $n-1$.

Указание. Доказательство основано на использовании метода аддитивных цепочек, троичной системы и следующей леммы:

можно построить со сложностью не более $4n$ одном циркулем такое множество точек, что среди отрезков с концами в этом множестве найдутся отрезки с длинами $1, 2, \ldots, n^2$.

Заметим, что если отрезок дан полностью (а не только его концы), то в оценке сложности задачи разделения его на n частей можно первое слагаемое n заменить на $n/2$.

В случае n, равного степени двойки, утверждение задачи 1 можно существенно усилить.

Задача 2. Пусть задан единичный отрезок. Можно построить циркулем отрезок длины 2^{-n} со сложностью не более

$$2\log_2 n + \frac{\log_2 n}{\log_2\log_2 n} + \frac{(2\log_2 n)\,(\log_2\log_2\log_2 n + O(1))}{(\log_2\log_2 n)^2}.$$

При $n = 2^m$ этот отрезок можно построить со сложностью $2\log_2 n + 4$. Можно разделить данный отрезок на 2^n равных частей одним циркулем со сложностью не более $2^{n-1} + 9$, если дан весь отрезок, а не только его концы.

Указание. Доказательство основано на использовании аддитивных цепочек и следующей леммы:

если дан отрезок длины $x < 1$, то можно построить со сложностью не более $4n$ одним циркулем такое множество точек, что среди отрезков с концами в этом множестве найдутся отрезки длиной x, x^2, $\ldots$, x^{n^2}.

При использовании линейки сложность построения, указанная в задаче 1, может быть уменьшена.

Задача 3. Можно построить циркулем и линейкой отрезки длины n и $1/n$ со сложностью не более

$$\frac{\log_2 n}{2\log_2\log_2 n}\left(1+\frac{3\log_2\log_2\log_2 n+O(1)}{\log_2\log_2 n}\right)$$

и разделить отрезок циркулем и линейкой на n равных частей со сложностью не более

$$\frac{n}{2}+\frac{\log_2 n}{2\log_2\log_2 n}\left(1+\frac{3\log_2\log_2\log_2 n+O(1)}{\log_2\log_2 n}\right).$$

Сложность разделения отрезка циркулем и линейкой на n равных частей не меньше $\lceil (n-1)/2 \rceil$. Для некоторой бесконечной последовательности чисел n сложность построения отрезка длины n циркулем и линейкой больше

$$\frac{1}{6}\frac{\log_2 n}{\log_2\log_2 n}.$$

Указание. Доказательство верхней оценки основано на использовании того факта что сложность вычисления произвольного числа n в базисе $\{x+y, xy, 1\}$

$$\tau(n) \leqslant \frac{\log_2 n}{\log_2\log_2 n}\left(1+\frac{3\log_2\log_2\log_2 n+O(1)}{\log_2\log_2 n}\right),$$

и следующей леммы:

можно построить со сложностью не более $4n$ *одном циркулем такое множество точек, что среди отрезков с концами в этом множестве найдутся отрезки с длинами* $1, 2, \ldots, n^2$.

Доказательство нижней оценки основано на мощностных соображениях.

В [1] часть приведенных выше результатов была независимо получена (в несколько более слабом виде), причем один из них был предложен в качестве задачи[6)] на Всероссийской олимпиаде школьников. Но в [1] получены и несколько новых интересных результатов, из которых, в частности, следует, что оценка, указанная в задаче 2, по порядку неулучшаема. А именно, там показано существование такой константы c, что если отрезок длины a построен со сложностью l, то $a < 2^{c^l}$. Если добавить к рассуждениям [1] использование леммы Ландау – Миньотта, то можно получить оценку $L(a) < 2^{c^l}$, где $L(a)$ — сумма модулей коэффициентов минимального многочлена с целыми коэффициентами, корнем которого является число a. Фактически, это утверждение в [1] есть, но доказательство

[6)] Второй автор — А. Я. Белов — задачу на олимпиаду предлагал, а первый автор — М. В. Алехнович — будучи во время олимпиады школьником, ее решал.

не приведено (но доказано, что степень этого многочлена не больше 2^l). Если далее воспользоваться известными в теории трансцендентных чисел теоремами о точности аппроксимации числа π алгебраическими числами данной степени и высоты, то можно доказать, что если $|\sqrt{\pi} - a| < \epsilon$, то $l > \Theta(\log_2 \log_2 1/\epsilon)$, т. е. сложность приближенного решения квадратуры круга по порядку не меньше двойного логарифма от $1/\epsilon$, где ϵ — относительная погрешность построения стороны квадрата, равновеликого данному кругу. Используя алгоритм вычисления π из [21], можно выполнить приближенное решение квадратуры круга со сложностью $O(\log_2 \log_2 1/\epsilon)$ (см. [5]). Значит, эта оценка по порядку неулучшаема.

Список литературы

[1] Алехнович М. В., Белов А. Я. *Сложность алгоритмов при построении циркулем и линейкой* // Фундаментальная и прикладная математика. Т. 7, №2, 2001. С. 597–614.

[2] Ахо А., Хопкрофт Д., Ульман Д. *Построение и анализ вычислительных алгоритмов*. М.: Мир, 1979.

[3] Гашков С. Б. *Алгоритм Евклида, цепные дроби, числа Фибоначчи и квадрирование прямоугольников* // Математическое просвещение. Сер. 2, вып. 6, 2002. С. 93-115.

[4] Гашков С. Б., Кочергин В. В. *Об аддитивных цепочках векторов, вентильных схемах и сложности вычисления степеней* // Методы дискретного анализа в теории графов и сложности. Т. 52, 1992. С. 22–40. [Анг. пер.: Gashkov S., Kochergin V. On addition chains of vectors, gate circuits, and the complexity of computation of power, Syberian Advances in Mathematics, 1994, v.4, no 4, 1–16.]

[5] Гашков С. Б., Чубариков В. Н. *Арифметика. Алгоритмы. Сложность вычислений.* 1е изд. М.: Наука, 1996, 2е изд. М.: Высшая школа, 2000, 3е изд. М.:Дрофа, 2005.

[6] Григорьев Д. Ю. *Нижние оценки в алгебраической сложности вычислений* // Теория сложности вычислений. I., Записки научных семинаров ЛОМИ. Т. 118, 1982. С. 25–82.

[7] Карацуба А. А. *Сложность вычислений* // Труды Математического ин-та РАН. Т. 211, 1995. С. 1–17.

[8] Касим-Заде О. М. *О сложности схем из единичных сопротивлений и о некоторых свойствах чисел Фибоначчи* // Труды матем. института им. Стеклова. Т.218. М.: Наука, 1997. С. 233–248.

[9] Кнут Д. *Искусство программирования* Т. 2. Изд. Вильямс, 2000.

[10] Лупанов О. Б. *Асимптотические оценки сложности управляющих систем*. М.: Изд. МГУ, 1984.

[11] Лупанов О. Б. *О вентильных и контактно-вентильных схемах* // ДАН СССР, т. 111, №6, 1956. С. 1171–1174.

[12] *Математика в задачах*. М.: МЦНМО, 2009.

[13] Ноден П., Китте К. *Алгебраическая алгоритмика*. М.: Мир, 1999.

[14] Сидоренко А. Ф. *Сложность аддитивных вычислений семейств целочисленных линейных форм* // Записки научных семинаров ЛОМИ. Т. 105, 1981. С. 53–61.

[15] Смарт Н. *Криптография*. М.: Техносфера, 2005.

[16] Трауб Дж., Вожьняковский Х. *Общая теория оптимальных алгоритмов*. М.: Мир, 1983.

[17] Трауб Дж., Вожьняковский Х., Васильковский Г. *Информация, неопределенность, сложность*. М.: Мир, 1988.

[18] Bernstein D. J. *Pippenger's exponentiation algorithm*. `http://cr.yp.to/papers.html#pippenger`

[19] Bernstein D. J. *The transposition principle*. `http://cr.yp.to/transposition.html`.

[20] Blum L., Cucker F., Shub M., Smale S. *Complexity and real computation*. Springer-Verlag, 1998.

[21] Borwein J. M., Borwein R. P. *Pi and the AGM*. Wiley, New York, 1987.

[22] *Handbook of theoretical computer science. Algorithms and Complexity*. Elsevier-MIT Press, 1990.

[23] Gathen, von zur, J., Gerhard J. *Modern computer algebra*. Cambridge University Press, 1999.

[24] Pippenger N. *On the evaluation of powers and monomials* // SIAM J. Comput. Vol. 980, № 9, 1980. P. 230–250.

[25] `www.ccas.ru/personal/karatsuba/algen.htm`

С. Б. Гашков, механико-математический факультет МГУ им. М. В. Ломоносова
E-mail: `sergey.gashkov@lsili.ru`, `sbgashkov@gmail.com`