О СХЕМНОЙ СЛОЖНОСТИ СТАНДАРТНОГО МЕТОДА УМНОЖЕНИЯ ЧИСЕЛ И МЕТОДА КАРАЦУБЫ

Сергеев И.С.

(ФГУП "Научно-исследовательский институт "Квант", Москва, Россия)

Целью настоящей работы является получение аккуратных оценок сложности основных применяемых на практике методов умножения чисел: стандартного (школьного) метода и метода Карацубы [1]. Под сложностью мы понимаем число элементов в реализующей метод логической схеме над базисом битовых операций $\{AND, OR, XOR, NAND, NOR, XNOR\}$.

Известные оценки сложности методов приведены, например, в [2]. Стандартный метод умножения n-разрядных двоичных чисел имеет сложность $M(n) \leq 6n^2 - 8n$. Сложность $K(n)$ метода Карацубы при $n = 2^k$ определяется из рекуррентного соотношения $K(2n) \leq 3K(n) + 49n - 8$ как $K(2^k) \leq 26\frac{2}{9} \cdot 3^k - 49 \cdot 2^k + 4$.

Мы покажем, что указанные оценки могут быть улучшены при помощи результата работы [3], который заключается в том, что сумма n бит может быть вычислена за $4{,}5n$ операций вместо $5n$ операций в тривиальном методе, а также некоторых дополнительных соображений.

Вспомогательные схемы. Схемы умножения строятся из следующих подсхем: полусумматоров HA, $HA^{\pm}$, (3, 2)-компрессоров FA_3, FA_3^-, FA_3^0, SFA_3, SFA_3^-, (5, 3)-компрессоров $MDFA$, $MDFA^-$, которые реализуют функции соответственно:

$HA: (x_1, x_2) \to (u; v)$, где $x_1 + x_2 = 2u + v$;

$HA^{\pm}: (x_1, x_2) \to (u; v)$, где $x_1 - x_2 = -2u + v$;

$FA_3: (x_1, x_2, x_3) \to (u; v)$, где $x_1 + x_2 + x_3 = 2u + v$;

$FA_3^-: (x_1, x_2, x_3) \to (u; v)$, где $x_1 + x_2 - x_3 = 2u - v$;

$FA_3^0: (x_1, x_2, x_3) \to (u; v)$, где $x_1 + x_2 - x_3 = 2u + v$, если $x_1 + x_2 - x_3 \geq 0$;

$SFA_3: (x_1, x_1 \oplus x_2, x_3) \to (u; v)$, где $x_1 + x_2 + x_3 = 2u + v$;

$SFA_3^-: (x_1, x_1 \oplus x_2, x_3) \to (u; v)$, где $x_1 - x_2 + x_3 = 2u - v$;

$MDFA$: $(x_1, x_1 \oplus y_1, x_2, x_2 \oplus y_2, z) \to (u_1, u_1 \oplus u_2; v)$, где $x_1 + y_1 + x_2 + y_2 + z = 2(u_1 + u_2) + v$;

$MDFA^-$: $(x_1, x_1 \oplus y_1, x_2, x_2 \oplus y_2, z) \to (u_1, u_1 \oplus u_2; v)$, где $x_1 - y_1 + x_2 - y_2 + z = 2(u_1 - u_2) + v$.

Перечисленные подсхемы изображены на рис. 1 (схема $MDFA$ предложена в [3]). Элементы AND, OR, XOR обозначены символами ∧, ∨, ⊕ соответственно. Инвертированные входы элементов отмечены кружками.

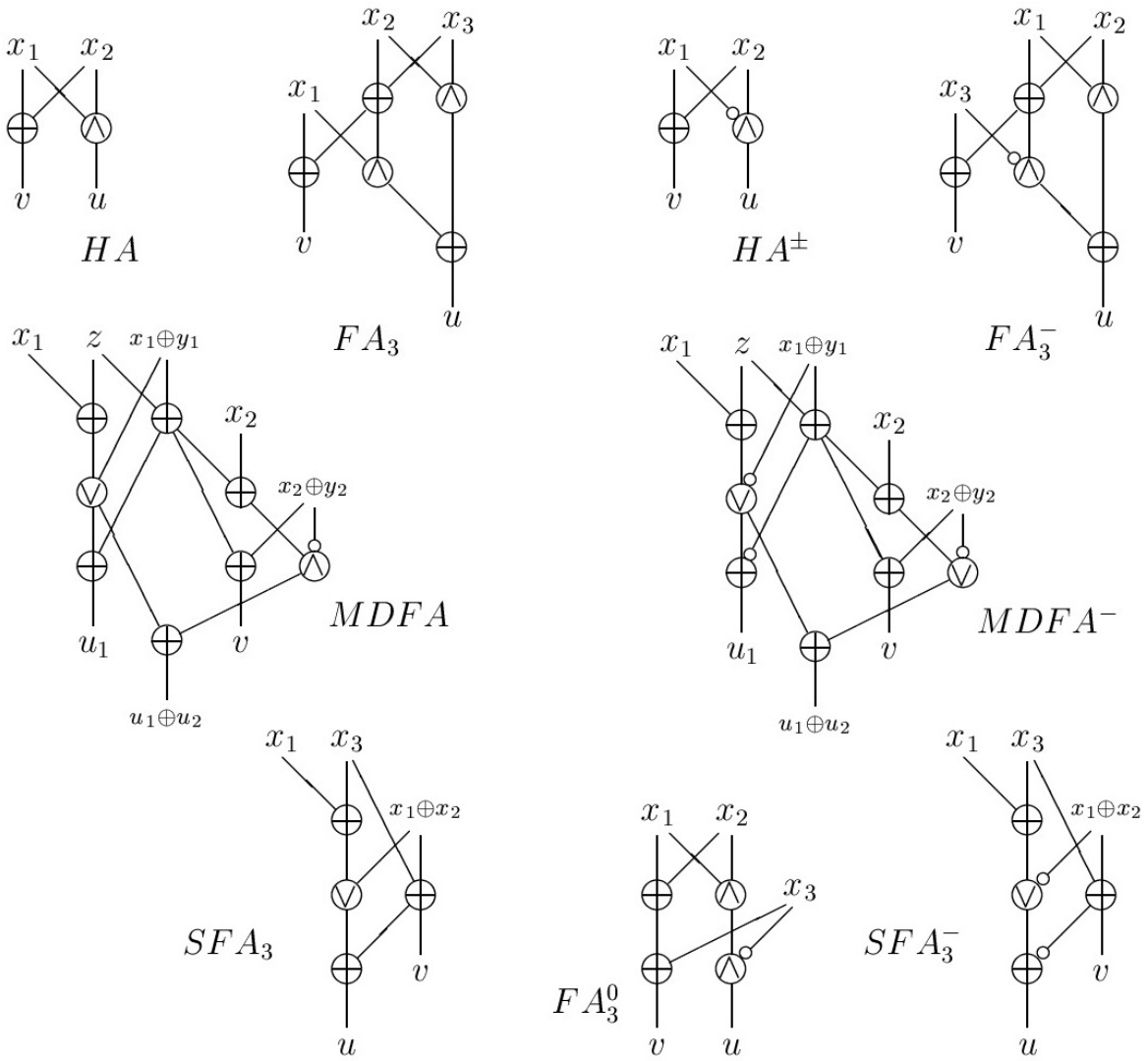

Рис. 1

Стандартный метод. Первая фаза стандартного метода умножения n-разрядных чисел состоит в выполнении n^2 битовых умножений. Следующая фаза — многократное сложение — реализуется последовательными сложениями по столбцам. Столбец — это множество битов одинакового разряда. Сложения выполняются при помощи

перечисленных схем-компрессоров. Результат сложения в столбце: бит суммы и множество переносов в следующий разряд.

Нумеруя столбцы с единицы, начиная с младших разрядов, имеем перед выполнением сложений по $n - |n - k|$ битов в k-м столбце, $k = 1, \dots, 2n - 1$. Сложение в столбце выполняем по правилу: если есть 5 битов-слагаемых, то используем $MDFA$; иначе, если есть 3 бита, то используем FA_3; иначе, если есть 2 бита, используем HA.

Определим $h(k)$ — число битов в k-м столбце после выполнения сложений в столбцах с меньшими номерами. Ясно, что $h(1) = 1$. Проверим по индукции, что $h(k) = 2k - 2$ для $2 \leq k \leq n$. Очевидно, это верно для $k = 2$. Предположим, что это верно для $k = t$, и рассмотрим сложение в t-м столбце: сумма $2t - 2$ битов, согласно заявленной стратегии, вычисляется при помощи $\lfloor t/2 \rfloor - 1$ схем $MDFA$, одной схемы HA и при нечетном t еще одной схемы FA_3. Это дает $t - 1$ переносов в следующий разряд, откуда заключаем, что $h(t + 1) = t + 1 + t - 1 = 2(t + 1) - 2$.

Аналогичным образом выясняем, что $h(n + 1) = 2n - 2$ и $h(2n - k) = 2k + 1$ при $0 \leq k \leq n - 2$. Для сложения в $(n + 1)$-м столбце используется то же множество схем, что и при сложении в n-м. При сложении в $(2n - k)$-м столбце используем $\lfloor k/2 \rfloor$ схем $MDFA$ и при нечетном k еще одну схему FA_3, но при $k = 1$ вместо FA_3 используем схему SFA_3, так как при сложении в предыдущем столбце используется $MDFA$.

Использование $MDFA$ требует перехода к специальному кодированию битов $(x, y) \to (x, x \oplus y)$. Так как закодированные этим способом выходы всех $MDFA$, за исключением финального, в $(2n - 2)$-м столбце, подаются также на входы $MDFA$, то для выполнения всех сложений дополнительно требуется $q + 1$ элементов XOR, где q — число $MDFA$.

Таким образом, если $n \geq 4$, то для выполнения второй фазы умножения используются n схем HA, $n - 3 + 2(n \bmod 2)$ схем FA_3, одна схема SFA_3, $q = (n^2 - 3n)/2 + 1 - (n \bmod 2)$ схем $MDFA$ и $q + 1$ элементов XOR (число $MDFA$ легко установить, исходя из числа (3, 2)-компрессоров, так как $MDFA$ уменьшает число битов-слагаемых на 2, FA_3 или SFA_3 уменьшает число слагаемых на 1; число битов перед выполнением сложений n^2, после — $2n$). Суммируя сложности подсхем, для сложности $M(n)$ всей схемы умножения получаем формулу

$$M(n) = 5{,}5n^2 - 6{,}5n - 1 + (n \bmod 2),$$

которая верна и для $2 \leq n \leq 3$.

Метод Карацубы. В методе Карацубы перемножаемые m-разрядные числа представляются в виде $A_1 2^n + B_1$ и $A_2 2^n + B_2$, где $n = \lceil m/2 \rceil$, $B_i < 2^n$, $A_i < 2^{m-n}$. Произведение вычисляется по формуле

$$A_1 A_2 2^{2n} + \big((A_1 + B_1)(A_2 + B_2) - A_1 A_2 - B_1 B_2\big) 2^n + B_1 B_2.$$

Схема состоит из двух подсхем, вычисляющих суммы $A_1 + B_1$ и $A_2 + B_2$, трех подсхем умножений $(n+1)$-разрядных, $(m-n)$-разрядных и -разрядных чисел, и схемы, выполняющей финальное сложение с вычитаниями. Структура заключительного сложения показана на рис. 2: символами "+" и "−" обозначены биты слагаемых и вычитаемых соответственно, снизу подписаны номера разрядов (номер i соответствует разряду при 2^i), пары битов в скобках отсутствуют, если m нечетно.

$$\begin{array}{lllllllllll}
(+ & +) + \ldots + & + & + & + & + & + \ldots + & & & & A_1 A_2 2^{2n} \\
 & & + & + & + & + & + \ldots + & + & \ldots & + & (A_1 + B_1)(A_2 + B_2) 2^n \\
 & & & & (- & -) & - \ldots - & - & \ldots & - & -A_1 A_2 2^n \\
 & & & & - & - & - \ldots - & - & \ldots & - & -B_1 B_2 2^n \\
 & & & & & & & + & \ldots & + \ldots + & B_1 B_2 \\
4n-1 & & & 3n-1 & & & & 2n-1 & & n \quad 0 &
\end{array}$$

Рис. 2

Сложения-вычитания выполняем последовательно по столбцам, используя перечисленные выше компрессоры, по правилу: если есть 3 бита-слагаемых и 2 бита-вычитаемых, то используем $MDFA^-$; иначе, если есть 3 бита, то используем подходящий компрессор из семейства FA_3; иначе, если есть 2 бита, используем подходящий полусумматор типа HA.

Обозначим через $h^+(k)$ и $h^-(k)$ число слагаемых и вычитаемых битов в k-м столбце после выполнения сложений-вычитаний в столбцах с меньшими номерами. Легко проверить, что $h^+(n) = h^-(n) = 2$, $h^+(n+1) = h^-(n+1) = 3$ и $h^+(k) = 3$, $h^-(k) = 4$ для $n + 2 \le k \le 2m - n - 1$: в n-м столбце используем одну схему FA_3^- и одну схему HA; в $(n+1)$-м — одну схему $MDFA^-$ и одну схему $HA^\pm$; в последующих столбцах вплоть до $(2m - n - 1)$-го — по одной схеме $MDFA^-$ и одной схеме FA_3^-.

При нечетном m имеем $h^+(k) = h^-(k) = 3$ при $k = 3n - 1, 3n - 2$: в соответствующих столбцах используем по одному $MDFA^-$ и $HA^\pm$.

Далее $h^+(3n) = 3$, $h^-(3n) = 2$ (используем $MDFA^-$), $h^+(3n+1) = 3$, $h^-(3n+1) = 1$ (используем SFA_3^- и $HA^\pm$), $h^+(3n+2) = 2$, $h^-(3n+2) = 1$ (используем FA_3^0, так как разность $(A_1 + B_1)(A_2 + B_2) - A_1 A_2 - B_1 B_2$ неотрицательна). Далее $h^+(k) = 2$, $h^-(k) = 0$ при $3n + 3 \le k \le 2m - 1$: везде используем HA и только в старшем разряде XOR, поскольку перенос вычислять не нужно.

Как и в стандартном методе, для перехода к вспомогательной кодировке пар битов требуется дополнительно использовать $q+1$ элементов XOR, где q — число $MDFA^{-}$.

Оценим сложность $K(m)$ схемы. Сложение n-разрядного числа с n-разрядным выполняется со сложностью $5n-3$, а с $(n-1)$-разрядным — со сложностью $5n-6$ стандартным методом сложения (см., например, [2]). Если $m \geq 10$, то финальный сумматор состоит из $n-1$ схем HA и $HA^{\pm}$, $2m-2n-1$ схем FA_3^{-}, одной схемы SFA_3^{-}, одной схемы FA_3^{0}, $2n$ схем $MDFA^{-}$ и $2n+2$ элементов XOR.

Заметим, что часть операций можно сэкономить: в столбцах с номерами $n+i$ и $2n+i$, где $i=0,\dots,n-1$, присутствуют одинаковые пары битов (из слагаемых B_1B_2, $-A_1A_2 2^n$ и $-B_1B_2 2^n$, $A_1A_2 2^{2n}$ соответственно). Организуем вычисления так, что эти биты являются входами компрессора FA_3^{-} в столбцах с номерами n и $n+2,\dots,2m-n-1$, а в прочих столбцах (т.е. с номером $n+1$ и при нечетном m с номерами $3n-2$, $3n-1$) входами $MDFA^{-}$ в кодировке $(x, x \oplus y)$. Поэтому при $i=0$ и $2 \leq i \leq n-1-2(m \bmod 2)$ можно сэкономить по две операции XOR и $NAND$, если в младших разрядах использовать компрессор из рис. 3а, а в старших — из рис. 3б (компрессоры разные, потому что знаки слагаемых битов в младших и старших разрядах различны). При $i=1$ и $n-2(m \bmod 2) \leq i \leq n-1$ экономится по одной операции XOR.

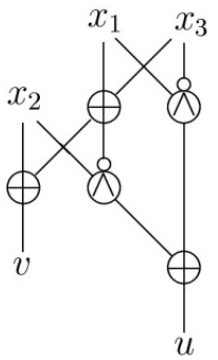

Рис. 3а

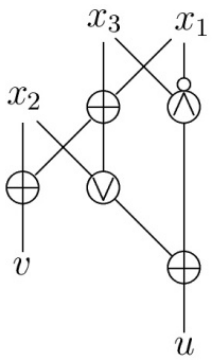

Рис. 3б

Получаем рекуррентные соотношения:

$$K(2n-1) \leq K(n+1) + K(n) + K(n-1) + 38n - 16, \qquad n \geq 6,$$

$$K(2n) \leq K(n+1) + 2K(n) + 38n - 2, \qquad n \geq 5.$$

Итерации метода Карацубы выгодно использовать при $m=16$ и $m \geq 18$. Значения битовой сложности $L(m)$ умножения m-разрядных чисел при $m \leq 18$ сведены в таблицу 1 (символом * отмечены значения, полученные методом Карацубы).

Для удобства сопоставления получим оценку сложности метода Карацубы в замкнутой форме для $m = 2^k$. Обозначим

$$X_k = \begin{bmatrix} K(2^k+2) \\ K(2^k+1) \\ K(2^k) \end{bmatrix}, \qquad A_k = \begin{bmatrix} 1 & 2 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 2 \end{bmatrix}, \qquad b_k = \begin{bmatrix} 38\cdot 2^k + 36 \\ 38\cdot 2^k + 22 \\ 38\cdot 2^k - 2 \end{bmatrix}.$$

Из приведенных выше рекуррентных соотношений следует, что $X_{k+1} \le AX_k + b_k$ при $k \ge 4$. Решая это неравенство стандартным способом (начальное значение X_4 берется из таблицы 1), получаем оценку:

$$K(2^k) \le 25\frac{83}{405}\cdot 3^k - 38\cdot 2^k - \frac{81}{5}\varphi_{k+2} - \frac{37}{5}\varphi_{k+1} + 20,$$

где $\{\varphi_k\}$ — последовательность чисел Фибоначчи: $\varphi_1 = \varphi_2 = 1$, $\varphi_{k+2} = \varphi_{k+1} + \varphi_k$.

Таблица 1: Сложность умножения m-разрядных чисел.

m	1	2	3	4	5	6	7	8	9
$L(m)$	1	8	30	61	105	158	224	299	387
m	10	11	12	13	14	15	16	17	18
$L(m)$	484	594	713	845	986	1140	1287*	1479	1598*

В заключение отметим, что построенные модификации схем для стандартного метода умножения и метода Карацубы имеют примерно на $1/12$ и на 4% меньшую сложность, чем (достаточно экономные) схемы, упомянутые во введении. Работа выполнена при поддержке РФФИ, проект № 14-01-00671а.

ЛИТЕРАТУРА

1. Карацуба А.А., Офман Ю.П. Умножение многозначных чисел на автоматах. ДАН СССР. 1962. Т. 145(2), с. 293-294.
2. Гашков С.Б. Занимательная компьютерная арифметика. Быстрые алгоритмы операций с числами и многочленами. М.: Либроком, 2012. 224 с.
3. Demenkov E., Kojevnikov A., Kulikov A., Yaroslavtsev G. New upper bounds on the Boolean circuit complexity of symmetric functions. Inf. Proc. Letters. 2010. V. 110(7), p. 264-267.