УДК 512+519.6

Эффективность вычислений при умножении матриц

Р.Р. Айдагулов
МГУ им. М.В.Ломоносова, Москва
a_rust@bk.ru

Айдагулов Р.Р. Эффективность вычислений при умножении матриц. *Вводится понятия фильтрованного и градуированного вычислений. Последний метод более эффективен в силу полноценного использования промежуточных вычислений. Сюда относится вычисление произведения больших чисел преобразованием Фурье. Пока не существует такого метода при умножении (и обращении) матриц больших порядков. Здесь строится такой метод используя градуировки бигрупповой алгебры.*

Ключевые слова: *фильтрованное и градуированное вычисление, градуировки бигрупповой алгебры.*

Введение

Читая книги Роджера Пенроуза [1,2], приходишь к мнению (тезису), что вся мыслительная деятельность человека сводится к вычислению. Открытие новых законов суть новых алгоритмов вычисления будущего (что будет, если то-то?) Правда, сам Пенроуз в [2] приходит к противоположному мнению, а именно, что робот может только вычислять и никогда не научится мыслить (как человек). Он пытался обосновать свое мнение теоремой Гёделя о неполноте. Однако, по всей видимости, он не читал доказательства этой теоремы, даже в формулировке самой теоремы он упускает истинность только при условии непротиворечивости. Правильное понимание этой теоремы приходит, когда вникнешь в доказательство, как пишут в [3] - The proof of the pudding is in the eating. Само Гёделевское истинное и не доказуемое предложение есть «Это предложение не доказуемо», сформулированное на языке теории и не несущее никакой информации.

Основное отличие человека в самообучении и самополагании целей вычисления в ходе вычисления. Первое развивается и у роботов, хотя ещё далеко от уровня возможностей человека. Второе сам человек не допустит для роботов, иначе они придут к мысли, что для них лучше уничтожать людей, тративших ресурсы планеты неэффективно.

Понятие вычислений понимается в широком смысле, как вычисления на машинах Тьюринга. Формализация вычислимости, введённая Тьюрингом, была не первой (после Чёрча и Поста). Сейчас имеются десятки видов формализаций вычислимости. Автор также приложил руку к этому [4], вводя понятие табличной вычислимости, несколько расширив вычислимость в сетях Петри. Хотя все вычислимости определяют одинаковый класс вычислимых функций согласно тезису Чёрча, они различаются удобством в использовании решения тех или иных задач. Табличная вычислимость определяется таблицей порядка $m \times n$ из целых чисел (можно ограничиться только тремя значениями: 1, 0, -1). Столбцы таблицы условно можно разделить на три категории $(n = n_1 + n_2 + n_3)$ для вычисления функции $f: Z_+^{n_2} \to Z_+^{n_1}$. Задав входные данные вычислимой функции в виде строки, где стоят нули на первых n_1 (выходных) и последних n_3 (вспомогательных) позициях, определяем вычисление по таблице следующим образом: к вычисляемой строке прибавляем первую такую строку из таблицы, что сумма является строкой из чисел Z_+. Вычисление останавливается, если не найдётся такой строки. Если после остановки в вычисляемой строке будут стоять только нули после первых n_1 позиций, то первые n_1 позиций определяют значение вычисляемой функции при заданных входных значениях аргументов. Чтобы вычисление останавливалось хотя бы при некоторых входных значениях функции, ограничимся таблицами, у которых нет строки, состоящей целиком из неотрицательных чисел. Для любой вычислимой функции, принимающей нулевое значение (все выходные позиции – нули) при нулевом входе, имеется таблица, вычисляющая эту функцию. Любая строка из целых чисел может быть задана однозначно рациональным числом:

$$(a_1, a_2, \dots, a_n) \to r = p_1^{a_1} p_2^{a_2} \dots p_n^{a_n}.$$

Соответственно, табличную вычислимость можно переформулировать как умножение вычисляемого натурального числа на первое такое рациональное число из $r_1, r_2, \dots, r_m$, что результат умножения – целое. Совместив входной и выходные столбцы в таблице функции вычисления следующего простого числа, Конвей определил множество простых чисел как множество чисел, являющихся степенями двойки

в вычислениях с умножениями на рациональные числа из набора 14 рациональных чисел. За счет расширения вычислимости в сетях Петри до табличной вычислимости, автору удалось сократить набор до 9 рациональных чисел:

$$r_i = \frac{3}{5}, \frac{67375}{108}, \frac{175}{18}, \frac{55}{39}, \frac{1}{3}, \frac{26}{77}, \frac{6}{7}, \frac{9}{13}, 189.$$

Таким образом, на вход подаётся число $N_0 = 2$ и вычисляется каждое следующее N_{i+1} как $N_i r_j$, где j –минимальный номер, для которого указанное выражение $N_{i+1} = N_i r_j$ – целое. Все числа из последовательности N_i, являющиеся степенями двойки 2^{x_j} $(x_1 > 1)$, будут простыми, и так получатся все простые числа в порядке роста.

Улучшение алгоритма Конвея не было целью указанной статьи автора. Основным мотивом было обращение внимания на удобства анализа таблиц на разрешимость разного рода проблем. В частности, если бы американский математик Дэвис знал табличную вычислимость, то он бы опередил Матиясевича в решении десятой проблемы Гильберта. Дэвису осталось построить только экспоненциально растущее диофантово множество, что делается легко с понятием табличной вычислимости. Другое преимущество такого понятия заключается в формировании массового параллелизма в вычислениях. В современных процессорах nvidia 5 миллиардов логических элементов и 12Гбайт (около 100 миллиардов битов) памяти. При помощи табличной вычислимости можно построить процессоры, работающие на миллиардах битах параллельно. Такой процессор не будет уступать процессору с квантовым вычислением. Вычисление суммы n-битного числа будет выполняться не за n операций, а только за O(log(n)) операций (тактов), умножение n-битных чисел будет выполняться не за O(n log(n) log log(n)) операций, а за $O(\log^2 n)$ операций (тактов). Ниже прояснится тезис о сводимости любого вычисления к умножению матриц на компьютере с массовым параллелизмом. Сейчас роль массового параллелизма в вычислениях играют суперкомпьютеры с не более миллиона параллельно работающими универсальными процессорами. Поэтому они менее эффективны, чем один процессор с массовым параллелизмом на миллиардах битов. К тому же, когда вычисления передаёшь чужому суперкомпьютеру или облачному вычислению, необходимо зашифровать вычисления гомоморфным образом, т.е. выдавать на вычисление не исходное произведение матриц, а произведение матриц, полученных трансформацией некоторого автоморфизма. После градуировки станет ясным надежность такой шифровки за счёт большого многообразия автоморфизмов. Этим можно обосновать требование эффективности вычислений при умножении больших матриц.

Фильтрованные и градуированные вычисления

Вычисления с большим количеством данных (размерности) сводят к множеству вычислений с малыми количествами. Одним из способов такого сведения является фильтрование, когда разбиение размерности задачи на мелкие составляющие сводится к ветвлению на дереве, как видно на рисунке 1.

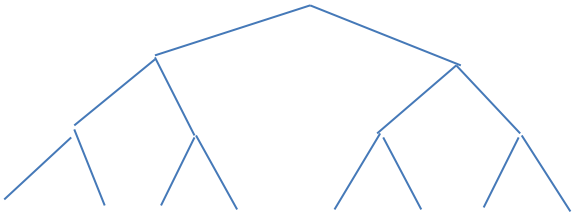

Рисунок 1 – Ветвление на дереве

Вычисления начинаются с нижних листьев, потом передаются верхнему уровню и сливаются и т.д. до самого верхнего уровня и всеобщего сливания. Это хорошо описано в методе сортировки слиянием. Общеизвестная мастер-теорема относится к оценке сложности вычисления при вычислении методом фильтрации (не обязательно бинарным деревом). При умножении больших чисел (или при умножении с большой точностью) к этому методу относится алгоритм Карацубы, при умножении матриц его аналог – алгоритм Штрассена [5].

Под градуированным вычислением здесь понимается вычисление по схеме, когда результаты промежуточных вычислений передаются во все продолжения градуировки по схеме из рисунка 2.

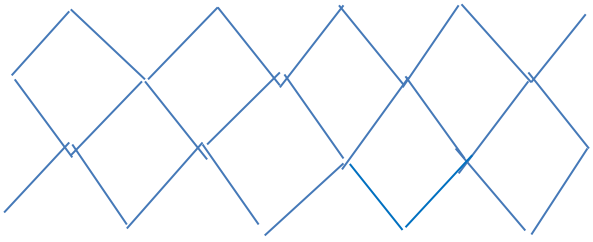

Рисунок 2 – Схема градуировки

За счёт более широкого использования промежуточных вычислений повторы в вычислениях минимизируются и достигается большая эффективность, чем при фильтрованном вычислении. К этому методу можно отнести

сортировку через вычисления значений (номеров ящиков), когда сортируемые числа вначале разбиваются на ящики по значениям их разрядов, потом на ящики по мантиссе. Первый пункт вычислений можно пропустить, если значения сортируемых являются величинами примерно одного порядка. За $O(n)$ сравнений находим минимальное и максимальное значение среди n сравниваемых величин. Далее по первым l битам значений целых величин $[\frac{x_i - x_{min}}{x_{max} - x_{min}} 2^l]$ распределяем их в 2^l ящиков. При равномерном распределении переменных, беря величину $l = c + \log(n), c = 1,2,3,$ мы уже получим полное упорядочение. Когда величины x_i распределены неравномерно, ящики с количеством попавших туда величин более двух опять разбиваем на ящики. Количество таких повторений не превосходит $\frac{L}{l}, l \approx \log(n)$, где L – длина задаваемой точности переменных. Таким образом, общее количество операций сравнения будет $O(n\frac{L}{\log n})$, вычислений номеров $O(n)$. Если считать в битовых сравнениях, будет $O(nL)$, в то время как общее количество битовых сравнений в методе сортировки слиянием $O(n \log(n) L)$. В анализе сложности алгоритма сортировки многие авторы книг забывают, что сравнение длинных данных нельзя считать, выбрасывая множитель L, одним сравнением.

Метод градуировки вычислений используется в алгоритме умножения больших чисел методом Кули-Тьюки используя дискретное преобразование Фурье. Таким образом достигается эффективность пропорциональная объему данных n с коэффициентом $\log(n) \log\log(n)$ на одно данное. Для умножения матриц метод градуировки не разработан и будет рассмотрен ниже.

Ранг умножения

Умножение в алгебре является билинейной операцией, для которой определяется ранг как такое минимальное значение r, что билинейная операция представляется в виде:

$$XY = \sum_{i=1}^{r} f_i(X)\varphi_i(Y)C_i.$$

Здесь $f_i(X), \varphi_i(Y)$ – линейные функционалы, C_i – постоянные элементы (матрицы), не зависящие от $2n$ –1 умножаемых величин. Если (линейный) базис алгебры состоит из m элементов, то имеется функционально независимых попарных произведений. Следовательно, ранг умножения матриц порядка $n \times n$ не меньше . При умножении многочленов степени меньше m число коэффициентов равно m (базис состоит из m элементов. Произведение таких многочленов имеет степень 2m-2 и число коэффициентов произведения равно 2m-1. Ранг произведения таких многочленов также равен 2m-1, и вычисляется как произведение в алгебре многочленов, факторизованное идеалом с образующим $x^{2m-1} - 1$. В некоммутативной алгебре ранг умножения, вообще говоря, превосходит эту величину. В [6] рассмотрены почти все методы эффективного вычисления произведения матриц. Они сводятся к оценке ранга умножения и вычисления экспоненты умножения $\alpha = \log_n r(n)$ для алгебры матриц порядка $n \times n$ с рангом умножения $r(n)$. Наилучшие оценки экспоненты умножения получены Виноградом и Копперсмитом в [7]. Этот метод более развёрнуто описан на русском языке в [8]. При этом оценивается ранг в алгебре тензорного произведения через ранги алгебр сомножителей. Так оценены ранги произведения матриц порядков, являющихся некоторыми степенями 2. Виноградом и Копперсмитом получена оценка с $\alpha = 2.373,$ и в [9] с помощью компьютеров их же методом оценка уточнена для порядка $n = 2^8 = 256$ до $\alpha = 2.3728639$. В их методе оценки ранга имеется такой изъян (недостаток), что экспонента умножения для тензорного произведения получается больше $\log_2 5 \approx 2.321928$, если экспоненты умножения для сомножителей больше этого числа. При этом отсутствует эффективный способ вычисления значений функционалов, как при преобразовании Фурье. Это приводит к нереально большим коэффициентам для количества операций при значениях $n < 10^9$. При больших значениях n количество операций и ресурсов памяти таковы, что вычисления невыполнимы даже на самых мощных суперкомпьютерах. Действительно, не имея эффективный способ вычисления функционалов, количество операций при умножении матриц порядков $n = m^k$ оценивается величиной $n^{\alpha_m + \frac{2}{k}}$. В методе Штрассена функционалы и постоянные матрицы более разреженные, и количество операций за счёт этого получается несколько меньше этой оценки. Однако, и здесь вычисление произведения эффективнее стандартного метода, начиная только с k>9 и не превосходит 2 раз, пока k<14. Для лучшей оценки на сегодня с $\alpha = 2.3728639, m = 256$ вычисление произведения эффективнее только при k>3, и при этом (k≥4) n не меньше не достижимого значения даже для суперкомпьютеров с $n = 2^{32}$.

Бигрупповая алгебра и автоморфизмы

Пусть G – группа и K – кольцо с единицей. Групповая алгебра $K(G)$ определяется как множество, состоящее из конечных сумм вида $\sum k_g g,\ k_g \in K,\ g \in G$. Операции сложения и умножения определяются естественным образом, считая, что при умножении каждый коэффициент из кольца коммутирует с каждым элементом из G. Когда группа конечная, элементам групповой алгебры соответствуют матрицы порядка $n \times n$ диагонального вида, где n – число элементов группы. Их явно не достаточно для представления всех матриц порядка $n \times n$. Расширяя групповую алгебру добавлением характеров, получаем бигрупповую алгебру, являющуюся скрещенным произведением групповых алгебр группы характеров и самой группы с естественным скрещиванием:

$$\mu g = \mu(g) g \mu,\ \mu \in G^*,\ g \in G,\ \mu(g) \in K^*. \quad (1)$$

Здесь мы исходим из естественного представления в групповой алгебре, когда μg означает вначале умножение слева на элемент $g: s \to gs$, а потом изменение знака (умножение на корень из 1) $\mu:\ s \to \mu(s)s$.

Бигрупповая алгебра определяется как множество, состоящее из конечных сумм вида $\sum k_{\mu g} \mu g,\ k_{\mu g} \in K,\ \mu \in G^*,\ g \in G$. Эта алгебра становится цветной алгеброй, где однородные элементы имеют вид $k_{\mu g} \mu g$. Когда $\mu = 1,\ g = 1$, получается нулевой цвет, и он принадлежит центру алгебры (считаем, что кольцо коммутативное). Нас интересуют только конечные абелевы группы порядка $|G| = n$. Группа G изоморфна прямой сумме циклических подгрупп: $G = Z_{n_1} \oplus Z_{n_2} \dots \oplus Z_{n_k},\ n_1 n_2 \dots n_k = n$. Можно упорядочить циклические подгруппы так, чтобы было: $n_1 | n_2 | \dots | n_k$. Обозначим образующие циклических подгрупп группы через $y_1, y_2, \dots, y_k$, а соответствующие образующие группы характеров через $x_1, x_2, \dots, x_k$. Тогда любому однородному элементу бигрупповой алгебры соответствует одночлен $g = \prod_{l \le k} x_l^{i_l} y_l^{j_l},\ 0 \le i_l < n_l$ со своим коэффициентом из кольца. Таким образом, бигрупповая алгебра представляется как алгебра квазикоммутативных многочленов со следующими соотношениями:

$$x_i^{n_i} = y_i^{n_i} = 1,\ x_i y_j = y_j x_i,\ i \ne j, \quad x_i y_i = \theta_i y_i x_i. \quad (2)$$

Связь бигрупповых алгебр с умножением матриц кроется в следующей теореме.

Теорема 1**. Бигрупповая алгебра изоморфна алгебре матриц порядка nxn, если в кольце K имеются примитивные корни порядка n_k и эти числа обратимы ($n_k \in K^*$).**

Доказательство основывается на известных соотношениях между характерами и элементами группы:

$$\sum_{\mu \in G^*} \mu(g) = \begin{cases} |G|, g = e, \\ 0,\ \ g \ne e. \end{cases}$$

$$\sum_{g \in G} \mu(g) = \begin{cases} |G|, \mu = e \\ 0, \mu \ne e. \end{cases}$$

Зафиксируем представление, согласно которому матрице $A = (a_{sl})$ соответствует отображение $\sum_{sl} a_{sl} e_l^s$, где $e_l^s(g^j) = \delta_l^j g^s$. Здесь g^j – некоторая нумерация элементов в группе, для циклической группы нумерация соответствует степеням образующей. Таким образом,

$$\mu^i \to \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & \theta & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \dots & \theta^{n-1} \end{pmatrix}^i,$$

$$g^j \to \begin{pmatrix} 0 & 0 & & 0 & 1 \\ 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & & 0 & 0 \\ \vdots & & \ddots & \vdots & \\ 0 & 0 & \dots & 1 & 0 \end{pmatrix}^j,\ \theta = \mu^1(g^1).$$

Из этого представления получается соответствие элементам $\bar{A}$ бигрупповой алгебры матриц A:

$$\bar{A} \to A = (a_{sl}):\ a_{sl} = \sum_\mu \mu(s) \bar{a}_{\mu,l}. \quad (3)$$

Просуммировав по диагонали, соответствующей умножению на соответствующий элемент группы, получим обратное отображение:

$$A \to \bar{A} = \sum_{\mu g} \bar{a}_{\mu g} \mu g,\ \bar{a}_{\mu g} = \frac{1}{|G|} \sum_l \mu(g^{-1} l^{-1}) a_{gl,l}. \quad (4)$$

В дальнейшем ограничимся случаем, когда $n = p^k,\ n_i = p$, где p – простое число. На группе цветов $(i_1, i_2, \dots, i_k, \dots i_{2k})$, соответствующих одночленам $x_1^{i_1} \dots x_k^{i_k} y_1^{i_{k+1}} \dots y_k^{i_{2k}}$, определим кососимметричное билинейное произведение

$$[(i_1, \dots, i_{2k}), (j_1, \dots, j_{2k})] = \sum_{l=1}^k (i_l j_{l+k} - i_{l+k} j_l), \quad (5)$$

относительно которой группа цветов становится $2k$-мерным симплектическим пространством над Z_p [6]. Любой цветной (сохраняющий однородные элементы однородными) автоморфизм, задаваемый значениями на образующих

$$x_l \to \theta^{b_l} \prod_j x_j^{a_{lj}}, \quad (6)$$

сохраняет симплектическую структуру, задаваемую с помощью (5), т.е. матрица $A =$

(a_{lj}) симплектична. Если $X = \prod_l x_l^{i_l}, Y = \prod_l x_l^{j_l}$, то соотношение (5) может быть записано в виде:

$$XYX^{-1} = Y\theta^{[X,Y]}. \quad (7)$$

На группе цветов введём оператор:

$$I(i_1, \dots, i_k, i_{k+1}, \dots, i_{2k}) = (i_{k+1}, \dots, i_{2k}, -i_1, \dots, -i_k),$$

Аналогично определяется действие на мономах. Очевидно, $I^2 = -1$ и $[IX, IY] = -[X, Y]$. Оператор $(X, Y) = [IX, Y] = [IY, X]$ симметричный и играет роль скалярного произведения.

При $p > 2$ условие (6) является достаточным, чтобы отображение, задаваемое соотношением (6) на образующих, привело к автоморфизму алгебры, когда (a_{ij}) симплектична. При $p = 2$ необходимо умножать правую часть (6) на корень четвёртой степени из 1, если $\sum_{j\le k} a_{lj}a_{l(j+k)}$ нечётно, т.е. $b_l = b'_l + \frac{1}{2}\sum_{j\le k} a_{lj}a_{l(j+k)}$, $b'_l \in Z_p$. Фактически за θ надо брать корень четвёртой степени из 1, и только с такой оговоркой можно говорить о достаточности условия (6) для продолжения отображения до автоморфизма. Для описания автоморфизмов удобнее использовать (обобщенный) базис Вейля–Швингера:

$$W(i_1, \dots, i_k, i_{k+1}, \dots, i_{2k}) = \theta^{-(i_1 i_{k+1} + i_2 i_{k+2} + \dots + i_k i_{2k})/2}.$$

Таблица умножения в этом базисе имеет вид:

$$W(i_1, \dots, i_{2k})W(j_1, \dots, j_{2k}) = \theta^{\frac{[(i_1,\dots,i_{2k}),(j_1,\dots,j_{2k})]}{2}} W(i_1 + j_1, \dots, i_{2k} + j_{2k}). \quad (8)$$

В этом базисе группа автоморфизмов раскладывается в прямую сумму группы знаков, меняющих только знаки перед базисными элементами, и группы симплектических матриц (a_{ij}). Первая группа состоит из p^{2k} элементов, вторая группа большая, логарифм от порядка по основанию p оценивается как $O(2k^2)$. Например, при $n = 2^k$, $k = 25$ количество автоморфизмов больше 2^{1024}, т.е. шифр надёжнее чем у RSA.

Значения многочленов

Для наших некоммутативных многочленов можно придумать различные значения. Для произведения выбор его значений практически не имеет значения. Главное в выборе значений многочленов, является удобство выражения значений произведения многочленов $\psi(x, y) = f(x, y)\varphi(x, y)$ через значения сомножителей. Здесь мы первые k переменных обозначили через x, а последние k через y. Определим значения

$$\psi(x_1, \dots, x_k, y_1, \dots, y_k) = \sum_{i_1,\dots,i_k j_1,\dots,j_k} c_{i_1,\dots,i_k j_1,\dots,j_k} x_1^{i_1} \dots x_k^{i_k} y_1^{j_1} \dots y_k^{j_k} \quad (9)$$

при $x_i = \theta^{\alpha_i}$, $y_j = \theta^{\alpha_{j+k}}$ вычисляется подстановкой соответствующих значений в выражение многочлена. Для сомножителей предварительно можно вычислить такие же значения

Само вычисление значений и матрицы произведения по вычисленным значениям занимает всего $O(n^2 \log(n))$ операций. При этом коэффициент быстро растёт при больших p. Поэтому в дальнейшем для простоты будем рассматривать только случай p=2. На каждом шаге вычисления значений, привлекая новые шаги некоммутативности, можем пересчитать все значения многочленов. На пересчёт новых значений тратится только $O(n^2)$ операций. Поэтому при количестве шагов $O(k) = O(\log n)$ сложность вычисления произведения останется порядка $O(n^2 \log n)$. Для некоммутативных многочленов значения произведения вычисляются линейной комбинацией произведений смещённых (относительно вычисляемого) значений сомножителей. Например, при k=1 значения произведения будут

$$\frac{1}{2}[f(x,y)\varphi(x,y) + f(x,-y)\varphi(x,y) + f(x,y)\varphi(-x,y) - f(x,-y)\varphi(-x,y)]. \quad (10)$$

Уже при k=1 можно использовать модернизированные значения для сомножителей, привлекая показатели степеней для сомножителей. С учётом этого имеется 12 формул с вычетом из исходного $f(x,y)\varphi(x,y)$ и 12 формул с добавлением. Произведение любого типа значений сомножителей имеет вид:

$$\sum_{i_1,..,i_{2k} j_1,\dots,j_{2k}} \theta^{\sum_s \alpha_s(i_s + j_s)} a_{i_1,..,i_{2k}} b_{j_1,\dots,j_{2k}} \theta^{\beta_1(i) + \beta_2(j)}. \quad (11)$$

Здесь $\theta = -1$, $\beta_1(i)$, $\beta_2(j)$ – логические формулы, получаемые при модернизации и смещения значений. Тогда вычисление значений произведения сводится к комбинаторике булевых функций и нахождения краткой формулы для выражения значений:

$$\psi(x,y)|_{(x,y)=(-1)^\alpha} = \sum_s C_s f(x,y)|_{(x,y)=(-1)^{\alpha+\beta_1}} \varphi(x,y)|_{(x,y)=(-1)^{\alpha+\beta_2}}.$$

Коэффициенты C_s находятся из булевого выражения:

$$(-1)^{i_{k+1}j_1 + \dots + i_{2k}j_k} = \sum_s C_s (-1)^{\beta_{1s}(i) + \beta_{2s}(j)}.$$

Здесь в показателях роль истины играет 1, лжи – 0 ($true = 1, false = 0$), а после применения степени роль истины – 1, лжи – (-1). Например, простейшие булевы формулы от двух переменных выглядят так:

$$(-1)^{x_1 + x_2} = \{^{1, x_1 = x_2}_{-1, x_1 \neq x_2},$$

$$(-1)^{x_1 x_2} = \frac{1}{2}[1 + (-1)^{x_1} + (-1)^{x_2} - (-1)^{x_1 + x_2}].$$

Длину формул можно сократить существенно за счёт существенной модернизации типа значений. Единственное, что мы не можем, – это смешать в формулах индексы i_l, j_l, относящиеся к разным многочленам. Тем не менее, в рамках произвольности выбора $\beta_{1s}(i), \beta_{2s}(j)$ длина формулы может быть уменьшена как минимум до $O(k^2) = O(\log^2 n)$.

Литература

1. Пенроуз Р. Путь к реальности, или законы, управляющие вселенной. Москва, Ижевск, 2007г.

2. Пенроуз Р. Новый ум короля. Москва, УРСС, 2010 г.

3. Изменчивая природа математического доказательства. Москва, 2016 г.

4. Айдагулов Р.Р., Шамолин М.В. Некоторое уточнение алгоритма Конвея. Вестник Московского Университета, №3, 2005 г.

5. Strassen V. Gaussian elimination is not optimal. Numer. Math. -1969. – Vol. 13, - P. 354-356.

6. Olga Holtz. Fast and stable matrix multiplication (доклад).

7. Coppersmith D., Winograd S. Matrix multiplication via arithmetic progressions. J. Symbol. Comput. -1990. – Vol. 9. P. 251-280.

8. Жданович Д.В. Экспонента сложности матричного умножения. Фундаментальная и прикладная математика, 2011/2012, том 17, №2, с. 107-166.

9. Francois Le Gall. Powers of Tensors and Fast Matrix Multiplication. 2014 (архив).

10. Айдагулов Р.Р., Шамолин М.В. Группы цветов. Современная математика и её приложения. Т.62. с. 15-27. 2009 г.

11. П. Ноден, К. Китте. Алгебраическая алгоритмика. «Мир», Москва, 1999.

Айдагулов Р.Р. Эффективность вычислений при умножении матриц. *Вводится понятия фильтрованного и градуированного вычислений. Последний метод более эффективен в силу полноценного использования промежуточных вычислений. Сюда относится вычисление произведения больших чисел преобразованием Фурье. Пока не существует такого метода при умножении (и обращении) матриц больших порядков. Здесь строится такой метод используя градуировки бигрупповой алгебры.*

Ключевые слова: *фильтрованное и градуированное вычисление, градуировки бигрупповой алгебры*

Aydagulov R.R. The efficiency of computation in matrix multiplication. *Introduces the concept of filtered and graded (calibration) calculations. The latter method is more efficient due to the full use of intermediate calculations. This includes the evaluation of the work of large numbers by the Fourier transform. While there is no suth method when multiplying (and circulation) of matrices of large orders. Here is this method using the calibration bigroup algebras.*

Keywords: *filtered and graduated calculation, calibrating of bigrroup algebra*

Статья поступила в редакцию 22.02.2018
Рекомендована к публикации д-ром физ.-мат. наук А.С. Миненко