is correct and selfadjoint if and only if c_1 is a real nonzero constant.
The unique solution of (3.16), for each $f \in L_2(0,1)$, is given by the formula

$$\begin{aligned} u(t) =& \widehat{A}^{-3} f(t) + \frac{c_1}{9\pi^3}[\sin 3\pi t + 9\sin \pi t + 3ic_1(3\cos \pi t + \cos 3\pi t) + 9c_1^2(\sin \pi t \\ &+ \sin 3\pi t)] \int_0^1 (\sin \pi x + \sin 3\pi x) f(x)dx + [3\cos \pi t + \cos 3\pi t - 3ic_1(\sin \pi t \\ &+ \sin 3\pi t)] \int_0^1 (3\cos \pi x + \cos 3\pi x) f(x)dx + (\sin \pi t \\ &+ \sin 3\pi t) \int_0^1 (9\sin \pi x + \sin 3\pi x) f(x)dx, \end{aligned} \tag{3.17}$$

where $\widehat{A}^{-3} f(t)$ is defined by (3.12).
And finally $\dim R(B_3 - \widehat{A}^3) = 3$.

References

[1] I.N. Parasidis and P.C. Tsekrekos, *Correct selfadjoint and positive extensions of non-densely defined symmetric operators.* Abstract and Applied Analysis, Hindawi Publishing Corporation, **7** (2005), 767–790.

[2] I.N. Parasidis and P.C. Tsekrekos, *Correct and selfadjoint problems with cubic operators.* submitted to the Conference "Computer Algebra" in St. Petersburg, Russia 2009.

Th. Lokkas
Technological Educational Institution of Larissa
41110 Greece
e-mail: thlokkas@teilar.gr

Вычислительные алгортмы для полиномиальных матриц с использованием ДПФ

А.О. Лапаев

Аннотация. Рассматривается новый подход к выполнению кольцевых операций в алгоритмах вычисления определителя, характеристического полинома и присоединенной матрицы в кольце $Z[x]$. Арифметические действия с полиномами заменяются на соответствующие действия с их образами, полученными при дискретном преобразовании Фурье. Проводится теоретическое и экспериментальное сравнение такого подхода с классическим, использующим Китайскую теорему об остатках как для полиномов, так и для их коэффициентов.

1. Введение

Получим теоретические выражения сложности для алгоритмов вычисления определителя, характеристического полинома и присоединенной матрицы для матрицы над полиномами одной переменной.

Пусть $A = (a_{ij}(x))$ – матрица над кольцом $\mathbf{Z}[x]$ размера $n \times n$, $a_{ij}(x) = \sum_{k=0}^{s_{ij}-1} a_{ij}^k x^k$. Пусть $\max_{i,j,k} |a_{ij}^k| \le \alpha$ и $\deg a_{ij} < s$.

Сложность вычисления дискретного преобразования Фурье по алгоритму быстрого преобразования Фурье для n^2 полиномов на $N = 2^{\lceil \log_2 ns \rceil}$ точках при использовании r модулей равна

$$n^2 r(7s\lceil \log_h \alpha \rceil + 9N \log_2 N).$$

2. Вычисление определителя

Будем применять алгоритм прямого хода [?] для вычисления определителя матрицы.

Оценим количество простых 32-битных модулей r, достаточное для получения определителя по его образу при дискретном преобразовании Фурье. Определитель матрицы A может быть вычислен по формуле:

$$\det A = \sum_{(j_1,\ldots,j_n)} (-1)^t a_{1j_1} a_{2j_2} \ldots a_{nj_n}, \tag{1}$$

где $(j_1, \ldots, j_n)$ – перестановка числел от 1 до n, t – четность этой перестановки.

Формула (1) содержит ровно $n!$ слагаемых. Оценив по ней максимальный коэффициент $\det A$ и воспользовавшись формулой Стирлинга для верхней оценки значения $n!$, получим, что количество простых модулей r будет равно

$$r = \lceil \log_h 2(\sqrt{2\pi n}\left(\frac{n}{e}\right)^n e^{\frac{1}{12n}} s^{n-1}\alpha^n)\rceil,$$

где $h = 2^{32}$.

Каждая кольцевая операция над образами полиномов состоит из rN арифметических операций над словами и rN вычислений остатка от деления. Будем считать, что вычисление остатка от деления занимает столько же времени, как 7 сложений слов. Тогда количество арифметических операций над образами полиномов исходной матрицы A при использовании алгоритма прямого хода равно

$$8n^3 rN.$$

Для восстановления полинома по его образу необходимо выполнить r обратных преобразований Фурье и восстановить ns чисел. Таким образом, число операций для получения полинома по его образу ДПФ будет

$$9rN\log_2 N + 2r^2 sn.$$

Общее количество операций в алгоритме вычисления определителя с использованием ДПФ равно

$$n^2 r(7s\lceil\log_h \alpha\rceil + 9N\log_2 N) + 8n^3 rN + 9rN\log_2 N + 2r^2 sn. \tag{2}$$

Приведем выражение сложности алгоритма вычисления определителя, использующего КТО как для полиномов, так и для их коэффициентов:

$$n^2 r(7s\lceil\log_h \alpha\rceil + ns^2) + 8n^4 sr + 2n^2 s^2 r + 2r^2 sn \tag{3}$$

$N = 2^{\lceil\log_2 ns\rceil}$ в (2),(3).

Если ns равно точной степени числа 2, то оценки (2) и (3) будут иметь вид (4) и (5):

$$n^2 r(7s\lceil\log_h \alpha\rceil + 9ns\log_2(ns)) + 8n^4 rs + 9ns\log_2(ns)\cdot r + 2r^2 sn \tag{4}$$

$$n^2 r(7s\lceil\log_h \alpha\rceil + ns^2) + 8n^4 rs + 2n^2 s^2\cdot r + 2r^2 sn \tag{5}$$

3. Вычисление характеристического полинома

Известен алгоритм вычисления характеристического полинома матрицы [?]. Данный алгоритм по кольцевым операциям также имеет сложность $O(n^3)$. Отличие находится только в оценке значения r и в том, что необходимо восстанавливать n полиномов. Для данного алгоритма $r_1 = \log_h 2n^n s^{n-1}\alpha^n$. Сложность этого алгоритма будет

$$n^2 r_1(7s\lceil\log_h \alpha\rceil + 9N\log_2 N) + 8n^3 r_1 N + n(9r_1 N\log_2 N + 2r_1^2 sn). \tag{6}$$

Классический подход с использованием КТО как для полиномов, так и для чисел будет иметь сложность

$$n^2 r_1(7s\lceil\log_h \alpha\rceil + ns^2) + 8n^4 r_1 s + n(2n^2 s^2 r_1 + 2r_1^2 sn). \tag{7}$$

Если ns равно точной степени числа 2, то оценки (6) и (7) будут иметь вид (8) и (9):

$$n^2 r_1(7s\lceil\log_h \alpha\rceil + 9ns\log_2(ns)) + 8n^4 r_1 s + n(9ns\log_2(ns)\cdot r + 2r_1^2 sn) \tag{8}$$

$$n^2 r_1(7s\lceil\log_h \alpha\rceil + ns^2) + 8n^4 r_1 s + n(2n^2 s^2\cdot r_1 + 2r_1^2 sn) \tag{9}$$

4. Вычисление присоединенной матрицы

Алгоритм вычисления присоединенной матрицы [?] по кольцевым операциям также имеет сложность $O(n^3)$. Необходимое количество модулей будет

$$r = \log_h 2(\sqrt{2\pi n}\left(\frac{n}{e}\right)^n e^{\frac{1}{12n}} s^{n-1}\alpha^n).$$

Необходимо восстанавливать n^2 полиномов по их образам. Таким образом, общая сложность алгоритма будет

$$n^2 r(7s\lceil\log_h \alpha\rceil + 9N\log_2 N) + 8n^3 rN + n^2(9rN\log_2 N + 2r^2 sn). \tag{10}$$

Также приведем выражение сложности классического метода:

$$n^2 r(7s\lceil\log_h \alpha\rceil + ns^2) + 8n^4 rs + n^2(2n^2 s^2 r + 2r^2 sn). \tag{11}$$

Если ns равно точной степени числа 2, то оценки (10) и (11) будут иметь вид (12) и (13):

$$n^2 r(7s\lceil\log_h \alpha\rceil + 9ns\log_2(ns)) + 8n^4 rs + n^2(9ns\log_2(ns)\cdot r + 2r^2 sn) \tag{12}$$

$$n^2 r(7s\lceil\log_h \alpha\rceil + ns^2) + 8n^4 rs + n^2(2n^2 s^2\cdot r + 2r^2 sn) \tag{13}$$

5. Экспериментальное сравнение алгоритмов

По выражениям сложности алгоритмы с использованием ДПФ близки к тем, которые используют КТО для полиномов и КТО для коэффициентов. Разница лишь в том, что сложность вычисления остатков от деления полиномов на простые полиномиальные модули $x, x+1, \ldots, x+ns$ составляет ns^2 и восстановление полинома по его остаткам от деления на простые модули требует $2n^2s^2$ операций, а получение образа полинома при ДПФ и восстановление полинома из его преобразования Фурье требует $9N\log_2 N$ операций. Так как $N \sim ns$, то использование ДПФ имеет преимущество. По теоретическим оценкам выигрыш будет наиболее заметен в случае, если ns – точная степень числа 2.

Приведем результаты вычислительных экспериментов. Сравнивались два алгоритма вычисления характеристического полинома матрицы над кольцом $Z[x]$: алгоритм Данилевского с применением КТО для полиномиальных простых модулей и Новый алгоритм [3], в котором операции над полиномами заменены операциями над их образами преобразования Фурье. Для восстановления коэффициентов полинома в результате в обоих алгоритмах использовалась КТО. Эксперименты проводились на ЭВМ следующей конфигурации: Intel Core 2, @1.46 GHz, 1 Gb ОЗУ, ОС Fedora Linux 10. Указанные выше алгоритмы были реализованы на языке программирования Java. Эксперименты ставились для матриц размера 8x8 и 16x16, степеней полиномов от 2 до 64 с 32-битными коэффициентами. Полученные результаты приведены в Таблице 1,2,3 и на Рисунке 1, 2.

s	Алгоритм Данилевского, ms	Новый алгоритм с ДПФ, ms
2	**83.0**	85.4
6	269.6	**136.8**
10	639.8	**274.6**
14	1169.2	**277.4**
16	1705.0	**761.0**
20	2316.0	**597.0**
24	3269.0	**586.0**
28	4352.0	**584.0**
32	5586.0	**1181.0**
36	7005.0	**1163.0**
40	9497.0	**1301.0**
44	11234.0	**1304.0**
48	13314.0	**1307.0**
52	15574.0	**1300.0**
56	18096.0	**1304.0**
60	20755.0	**1339.0**
64	23204.0	**2671.0**

Таблица 1. Результаты экспериментов с матрицами 8x8 над полиномами с 32-битными коэффициентами

s	Алгоритм Данилевского, ms	Новый алгоритм с ДПФ, ms
2	**1085.0**	1779.0
6	5107.0	**3614.0**
10	11977.0	**7125.0**
14	21896.0	**7155.0**
16	27208.0	**14415.0**
20	41001.0	**14031.0**
24	60078.0	**14678.0**
28	80925.0	**15045.0**
32	102761.0	**29928.0**
36	129246.0	**29991.0**
40	158378.0	**29814.0**
44	189182.0	**29755.0**
48	223105.0	**29920.0**
52	261548.0	**29958.0**
56	301388.0	**30011.0**
60	345024.0	**30095.0**
64	387278.0	**60845.0**

Таблица 2. Результаты экспериментов с матрицами 16x16 над полиномами с 32-битными коэффициентами.

n	Алгоритм Данилевского, ms	Новый алгоритм с ДПФ, ms	k
4	63.0	**23.0**	2.74
8	415.0	**143.0**	2.90
16	7655.0	**3619.0**	2.12
32	162543.0	**99562.0**	1.63

Таблица 3. Результаты экспериментов с матрицами 4x4, 8x8, 16x16 и 32x32 над полиномами степени 7 с 32-битными коэффициентами. Столбец k – коэффициент выигрыша Нового алгоритма

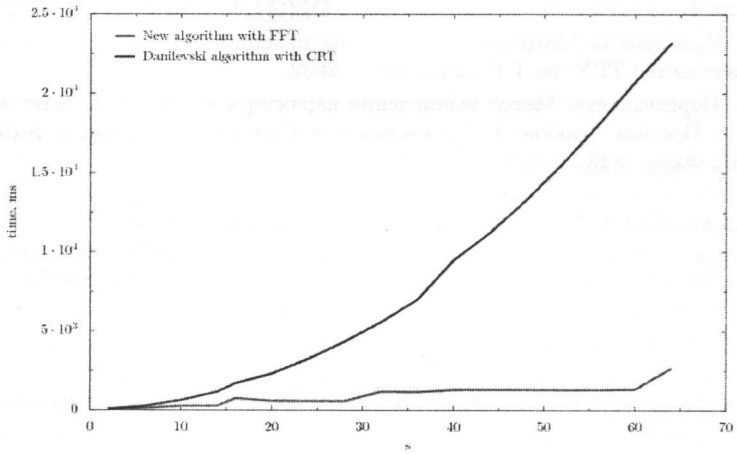

Рисунок 1. Размер матриц 8x8, полиномы с 32-битными коэффициентами. Ось x - степень полиномов, ось y - время выполнения алгоритмов.

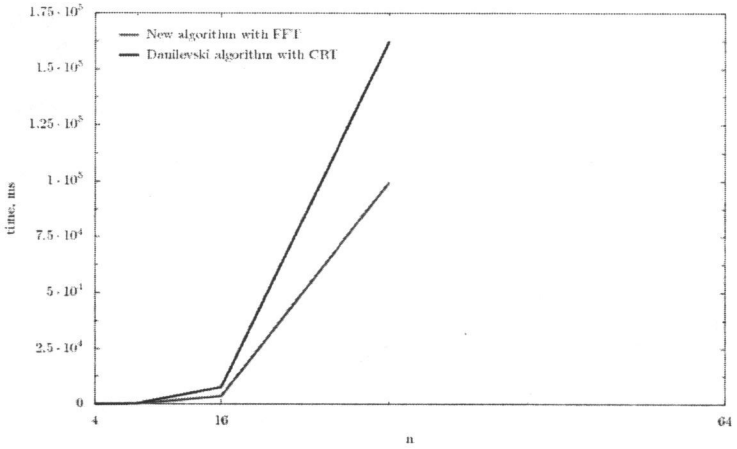

Рисунок 2. Размер матриц 4x4, 8x8, 16x16, 32x32, полиномы степени 7 с 32-битными коэффициентами. Ось x - размер матриц, ось y - время выполнения алгоритмов.

Также планируется экспериментальное сравнение алгоритмов вычисления определителя и присоединенной матрицы. Результаты будут представлены в докладе.

Список литературы

[1] А.О. Лапаев. О применении ДПФ в алгебраических вычислениях. // Вестник Тамбовского университета. Серия естественные и технические науки: 2009 – т.14. – Вып. 1.

[2] Г.И. Малашонок. Матричные методы вычислений в коммутативных кольцах. Издательство ТГУ им. Г.Р. Державина, 2002.

[3] О.Н. Переславцева. Метод вычисления характеристического полинома матрицы. // Вестник Тамбовского университета. Серия естественные и технические науки: 2008 – т.13. – Вып. 1. – с. 131-133.

А.О. Лапаев

Algorithms for the functional decomposition of symbolic polynomials

Stephen M. Watt

Abstract. Earlier work has presented algorithms to factor and compute GCD of symbolic Laurent polynomials, that is multivariate polynomials whose exponents are themselves integer-valued polynomials. More recently we have extended the notion of univariate polynomial decomposition to symbolic polynomials and have presented an algorithm to compute these decompositions. For example, the symbolic polynomial $f(x) = 2x^{n^2+n} - 4x^{n^2} + 2x^{n^2-n} + 1$ can be decomposed as $f = g \circ h$ where $g(x) = 2x^2 + 1$ and $h(x) = x^{n^2/2+n/2} - x^{n^2/2-n/2}$. This decomposition of symbolic Laurent polynomials relies on a functional decomposition of the usual multivariate Laurent polynomials with specific integer exponents. This paper shows this reduction and shows that the required multivariate Laurent polynomial decomposition is about the same cost as the decomposition of multivariate polynomials.

Stephen M. Watt
University of Western Ontario,
Canada
e-mail: smwatt@gmail.com