Об обращении разностных операторных матриц

С. А. Абрамов*
Вычислительный центр им. А.А.Дородницына ФИЦ ИУ РАН
ул. Вавилова, 40, Москва, 119333, Россия
sergeyabramov@mail.ru

Аннотация

Обсуждается алгоритм обращения матриц, элементами которых служат скалярные линейные разностные операторы над некоторым разностным полем $\mathbb{K}$. Алгоритм имеет меньшую сложность в сравнении с известными алгоритмами. Рассматриваются отличия обсуждаемого алгоритма от его дифференциального аналога.

1 Введение

Для матриц над некоторым полем или кольцом проверка обратимости и фактическое построение обратной матрицы тогда, когда она существует, являются известными математическими задачами, которые ниже рассматриваются применительно к операторным матрицам. В данном случае элементами матриц служат скалярные линейные разностные операторы над разностным полем — некоторым полем $\mathbb{K}$ с автоморфизмом (сдвигом) σ; по предположению поле $\mathbb{K}$ имеет характеристику 0. Обсуждаются новые алгоритмы решения названных задач. Оговоримся, что эти задачи могут решаться известными алгоритмами, предназначенными для решения более общих задач, и об этом еще будет сказано ниже. Но обсуждаемые новые алгоритмы имеют меньшую сложность. В обозримом будущем предполагается эти алгоритмы опубликовать, в настоящей же расширенной аннотации мы только приводим оценки их сложности.

Обычно в случаях операторных матриц вместо "обратимая матрица" употребляется термин *унимодулярная* матрица. Мы также будем пользоваться этим термином.

Алгоритмы проверки унимодулярности и построения обратной матрицы для дифференциального случая, когда $\mathbb{K}$ является дифференциальным полем характеристики 0 с дифференцированием ∂ и когда элементы матриц суть скалярные линейные операторы над $\mathbb{K}$, рассматривались автором в [1]. Как дифференциальные, так и обсуждаемые ниже разностные алгоритмы основаны на рассмотрении размерности пространства решений V_L соответствующей системы уравнений, компоненты решений принадлежат расширению Пикара-Вессио ([2, 3, 4]) поля $\mathbb{K}$ для L. Операторная матрица L полного ранга (строки L независимы над кольцом скалярных линейных операторов) является унимодулярной если и только если $\dim V_L = 0$, т.е. само пространство V_L является нулевым ([5]).

*Частичная поддержка РФФИ, грант 16-01-00174-а.

2 Отличие от дифференциального случая

Рассматриваемый здесь разностный случай отличается от разобранного в [1] дифференциального довольно существенно.

1. Простая замена ∂ на σ в алгоритмах для дифференциального случая не приводит к алгоритмам для разностного случая, здесь требуются дополнительные ухищрения. Уже говорилось, что основой алгоритмов является определение значения $\dim V_L$. Ситуация такова, что система $\sigma y = Ay$, где A — это $n \times n$-матрица с элементами из $\mathbb{K}$, имеет пространство решений размерности n если и только если матрица A невырождена, а в дифференциальном случае $y' = Ay$ эта невырожденность не требуется. Однако это вычисление размерности и в разностном случае может быть проведено для произвольной матрицы L алгоритмически ([5]).

2. В разностном случае возможно раздельное рассмотрение сложностей по числу арифметических операций и по числу сдвигов, подобно алгоритмам сортировки, для которых раздельно рассматривают сложности по числу сравнений и по числу перемещений элементов. В дифференциальном случае это раздельное рассмотрение двух сложностей невозможно, так как применение ∂ слева к скалярному дифференциальному оператору требует дополнительных сложений (предполагаем, что для операторов используется стандартное представление в виде полиномов от ∂): например, $\partial(a\partial + b) = a\partial^2 + (a' + b)\partial + b'$. Игнорирование операций дифференцирования может привести к неправильным оценкам числа арифметических операций. В разностном случае мы имеем дело с автоморфизмом σ, что открывает возможность раздельного рассмотрения сложностей.

3. Пусть разностное поле $\mathbb{K}$ является полем рациональных функций от x и применение σ к произвольной рациональной функции получается подстановкой $x + 1$ вместо x с последующим приведением результата к канонической записи (для σ^{-1} подставляется $x - 1$). Тогда естественно считать, что сдвиг σ имеет ту же сложность, что и σ^k при любом целом k. Принятие этой точки зрения может повлиять на сложность алгоритма по числу сдвигов. В дифференциальном случае отождествление сложностей ∂ и ∂^k было бы безосновательным.

3 Основные результаты

Кольцо $\mathbb{K}[\sigma, \sigma^{-1}]$ — это кольцо скалярных разностных операторов над разностным полем $\mathbb{K}$ с перестановочным правилом $a\sigma^r \cdot b\sigma^s = a(\sigma^r b)\sigma^{r+s}$. Кольцо $n \times n$-матриц с элементами в некотором кольце R обозначается через $\mathrm{Mat}_n(R)$. Соответственно $\mathrm{Mat}_n(\mathbb{K}[\sigma, \sigma^{-1}])$ есть кольцо операторных $n \times n$-матриц. Пусть $L \in \mathrm{Mat}_n(\mathbb{K}[\sigma, \sigma^{-1}])$, тогда L можно записать в развернутом виде $L = A_l\sigma^l + A_{l-1}\sigma^{l-1} + \cdots + A_t\sigma^t$, где $A_l, A_{l-1}, \ldots, A_t \in \mathrm{Mat}_n(\mathbb{K})$, при этом матрицы A_l, A_t ненулевые. В этом случае $\mathrm{ord}\, L = l - t$ называется *порядком* операторной матрицы L.

3.1 Произвольное разностное поле $\mathbb{K}$

Предполагается, что n и d — целые числа, $n > 0$, $d \geqslant 0$, при этом исходная операторная матрица L такова, что

$$L \in \mathrm{Mat}_n(\mathbb{K}[\sigma, \sigma^{-1}]), \quad \mathrm{ord}\, L = d. \tag{1}$$

Алгоритмам решения рассматриваемых матричных задач сопоставляются две сложности, являющиеся функциями от n и d:

- *арифметическая сложность*, т.е. число арифметических операций в поле $\mathbb{K}$ в худшем случае для заданных n, d,
- *сдвиговая сложность*, т.е. число применений операций σ, σ^{-1} к элементам поля $\mathbb{K}$ в худшем случае для заданных n, d.

В связи с понятием сдвиговой сложности поясним, что вычисление $\sigma^k a$ для $k \in \mathbb{Z}$, $a \in \mathbb{K}$ по предположению требует $|k|$ применений операции сдвига.

Теорема 1. *(i) Существует алгоритм проверки обратимости произвольной операторной матрицы L, для арифметической и сдвиговой сложностей которого справедливы соответственно оценки $O(n^3d^2)$ и $O(n^2d^2)$. Результатом применения алгоритма является "да" (матрица обратима) или "нет" (матрица необратима).*

(ii) Существует алгоритм проверки обратимости произвольной операторной матрицы L и построения обратной матрицы в случае ее существования, для арифметической и сдвиговой сложностей которого справедливы соответственно оценки $O(n^4d^2)$ и $O(n^4d^3)$. Результатом применения алгоритма является L^{-1} или сообщение о том, что матрица L не является унимодулярной.

3.2 Поле рациональных функций в роли $\mathbb{K}$

Пусть элементы поля $\mathbb{K}$ представлены выражениями, в которые входит переменная x, а сдвиг σ есть замена x на $x+1$ с возможным последующим упрощением выражения; например, $\mathbb{K}$ является полем рациональных функций. Тогда затраты на применение σ и σ^k, $k \in \mathbb{Z}$, можно считать одинаковыми. Это меняет представление о сдвиговой сложности. Хотя, с одной стороны, по-прежнему сдвиговая сложность для заданных n и d — это затраты на выполнение сдвигов в худшем для этих n и d случае. Но с другой стороны, затраты теперь определяются по-другому: в разделе 3.1 мы считали, что при $k \in \mathbb{Z}, a \in \mathbb{K}$ сдвиговые затраты на вычисление $\sigma^k a$ равны $|k|$, а теперь мы их считаем равными 1. В следующей теореме сдвиговая сложность понимается именно в этом смысле.

Теорема 2. *Пусть $\mathbb{K} = K(x)$, где K — поле характеристики 0 и x — переменная. Пусть $\sigma R(x) = R(x+1)$ для любой рациональной функции $R(x) \in K(x)$. Тогда существует такой алгоритм проверки унимодулярности произвольной операторной разностной матрицы и построения обратной матрицы в случае ее существования, что для его как арифметической, так и сдвиговой сложности справедлива оценка $O(n^4d^2)$.*

Аналогичный результат можно получить для q-разностного случая ([6, 7]). В наиболее простом его варианте $\mathbb{K} = K(q, x)$, где q — еще одна переменная и автоморфизм σ определяется посредством $x \to qx$.

Для разработанного алгоритма проверки унимодулярности с указанными в теореме 1(i) оценками арифметической и сдвиговой сложностей сделанное предположение об эквивалентности затрат на применение σ и σ^k не имеет существенного значения и оценки сохраняются прежними.

Если применение σ^k к какому-то $a \in \mathbb{K}$ потребовало нахождения $\sigma a, \ldots, \sigma^k a$, то эти значения можно запомнить для дальнейшего использования. В то же время такое вычисление $\sigma^k a$, как мы обсуждаем здесь, не даст нам $\sigma a, \ldots, \sigma^{k-1} a$, и если нам в дальнейшем понадобится, например, $\sigma^{k-1}a$, то потребуется его вычисление. Поэтому сложность какого-либо алгоритма при обсуждаемом здесь взгляде на связанные с применением σ^k затраты не обязательно будет меньше, чем при трактовке сдвиговой сложности в разделе 3.1.

4 Другие подходы

Связанные с проверкой унимодулярности и построением обратной матрицы задачи могут решаться разными алгоритмами. Например, могут использоваться алгоритмы построения форм Джекобсона и Эрмита заданной операторной матрицы; определения этих форм можно найти в [9, 10]. В [9] сложность предложенного там алгоритма построения формы Джекобсона рассмотрена как функция трех переменных, и две из них — это наши n, d (в [9] привлекались другие обозначения). Значение третьей переменной в худшем случае равно nd, и для сложности как функции переменных n, d можно получить оценку $\Theta(n^9d^9)$. Формой Эрмита унимодолярной матрицы служит единичная матрица, и в этом случае матрица U преобразования является обратной для исходной матрицы. Приведенная в [10, Thm 5.5] оценка сложности имеет в наших обозначениях вид $O(n^7d^3\log(nd))$. Похоже, что последняя оценка является точной. (Конечно, алгоритмы из [9, 10] решают более общие задачи, и обсуждавшиеся выше алгоритмы имеют некоторые преимущества только для проверки унимодулрности и построения обратной операторной матрицы.) В [9, 10] алгоритмы описаны на языке колец некоммутативных полиномов Оре ([11]), что делает эти алгоритмы применимыми в дифференциальном и разностном случаях.

Нелишне заметить, что мы обсуждаем алгоритмы со сложностной точки зрения. Алгоритм, который выглядит лучшим в этом смысле, не обязательно окажется лучшим в вычислительной практике.

5 Открытые вопросы, гипотезы

Неясно, существует ли алгоритм проверки унимодулярности со сложностью, допускающей оценку $O(n^\alpha d^\beta)$, где α, β суть вещественные числа и $\alpha < 3$. Для матриц, элементами которых являются обычные коммутативные полиномы из $K[x]$, известен опубликованный в [12] алгоритм построения обратной матрицы со сложностью $O(n^3\rho)$, где ρ является максимальной степенью элементов данных матриц (строго говоря, алгоритм из [12] предназначен для обращения только матриц “общего положения”). Неясно также, сводится ли задача построения обратной матрицы к задаче умножения матриц, аналогично сводимости в случае, когда элементы матриц принадлежат полю ([13, Sect. 16.4], [14, гл. 6]). Сводимость здесь понимается в том смысле, что если существует алгоритм умножения операторных матриц со сложностью (арифметической или сдвиговой) $T(n, d)$, то существует алгоритм обращения со сложностью $O(T(n, d))$. Предположение об этой сводимости вызывает сомнения, но при этом сводимость в другую сторону доказывается так же, как для матриц над полем.

Возвращаясь к матрицам с полиномиальными элементами, заметим, что существует алгоритм умножения матриц со сложностью $O(n^\omega\rho\, f(\log n \log\rho))$, где ω - показатель матричного умножения, $2 < \omega \leqslant 3$, f — некоторый полином ([15]). Однако алгоритм с такой сложностью для обращения матриц, скорее всего, не существуют, и похоже, что, все-таки, задача обращения матриц не сводится к задаче умножения матриц ни в полиномиальном, ни в операторном случае.

Но сказанное — это не более, чем предположение. Автор не располагает сведениями об алгоритмах, которые бы решали, например, задачу проверки унимодулярности с меньшей, чем указана в теореме 1(i), сложностью. Поиск в литературе не дал положительного результата, но, конечно, это не исключает существования такого алгоритма. Возможно, например, что с использованием идей, на которых построены алгоритмы быстрого матричного умножения над полем ([16, 17, 18]), и идей алгоритмов быстрого умножения скалярных линейных операторов ([19, 8, 20]), можно предложить алгоритм для быстрого умножения операторных матриц, а затем получить соответствующий алгоритм для проверки унимодулярности.

Многие недавние работы направлены на выяснение роста размера принадлежащих $\mathbb{K}$ коэффициентов, когда, например, $\mathbb{K} = \mathbb{Q}(x)$ (см. [21], [10] и т.д.). Было бы интересно исследовать битовую сложность алгоритмов проверки унимодулярности. Другой путь — рассматривать сложность как функцию трех переменных: n, d и ρ, где ρ таково, что все полиномы, участвующие в L как числители и знаменатели коэффициентов элементов, имеют степени, не превосходящие ρ. Сложность является числом операций в поле $\mathbb{Q}$ в худшем случае.

Список литературы

[1] Abramov S.A. On the differential and full algebraic complexities of operator matrices transformations. In Proc. of CASC'2016 (2016), 1–11.

[2] M. van der Put, M. F. Singer. Galois Theory of Differential Equations, Grundlehren der mathematischen Wissenschaften, **328**, Springer, 2003.

[3] Хованский А.Г. Топологическая теория Галуа. Разрешимость и неразрешимость уравнений в конечном виде. М.: МЦНМО, 2008.

[4] van der Put M., Singer M. F. Galois Theory of Difference Equations, LNM, **1666**, Springer, Heidelberg, 1997

[5] Abramov S., Barkatou M. On the dimension of solution spaces of full rank linear differential systems. In: Proc. CASC'2013. LNCS, **8136**, Springer, Heidelberg (2013), 1–9.

[6] Кац В.Г., Чен П. Квантовый анализ. М: МЦНМО, 2005.

[7] Andrews G.E. q-Series: Their Development and Application in Analysis, Number Theory, Combinatorics, Physics, and Computer Algebra. CBMS Regional Conference Series, **66**, AMS, R.I., 1986.

[8] Benoit A., Bostan A., van der Hoeven J. Quasi-optimal multiplication of linear differential operators. Proc. FOCS '12, New Brunswick, USA (2012), 524–530.

[9] Middeke J. A polynomial-time algorithm for the Jacobson form for matrices of differential operators. Tech. Report No. 08-13 in RISC Report Series, 2008

[10] Giesbrecht M., Sub Kim M. Computation of the Hermite form of a Matrix of Ore Polynomials. J. Algebra **376** (2013), 341–362.

[11] Ore O. Theory of non-commutative polynomials. Annals of Mathematics, **34** (1933), 480–508.

[12] Jeannerod C.-P., Villard G. Essentially optimal computation of the inverse of generic polynomial matrices. J. Complexity **21**(1) (2005), 72–86.

[13] Bürgisser P., Clausen M., Shokrollahi M.A. Algebraic Complexity Theory. Grundlehren der mathematischen Wissenschaften, **315** Springer, Heidelberg, 1997.

[14] Ахо А., Хопкрофт Дж., Ульман Дж. Построение и анализ вычислительных алгоритмов. М.: Мир, 1979.

[15] Bostan, A., Schost, E.: Polynomial evaluation and interpolation on special sets of points. J. Complexity **21**(4) (2005), 420–446.

[16] Strassen V. Gaussian elimination is not optimal. Numer. Math., **13** (1969), 354–356.

[17] Coppersmith D., Winograd S. Matrix multiplication via arithmetic progressions. J. Symb. Comput. **9**(3) (1990), 251–280.

[18] Vassilevska Williams, V.: Multiplying matrices faster than Coppersmith–Winograd. In: STOC'2012 Proceedings (2012), 887–898

[19] van der Hoeven J. FFT-like multiplication of linear differential operators. J. Symb. Comput. **33** (2002), 123–127.

[20] Bostan A., Chyzak F., Le Roix N. Products of ordinary differential operators by evaluating and interpolation. In: Proc. ISSAC'2008 (2008), 23–30.

[21] Beckermann B., Cheng H., Labahn G. Fraction-free row reduction of matrices of Ore polynomials. J. Symbolic Comput. **41** (2006), 513–543.