УДК 519.6
DOI: 10.20310/1810-0198-2016-21-6-2042-2046

USING THE SMITH FORM FOR THE EXACT MATRIX INVERSION

© G. I. Malaschonok

Tambov State University named after G.R. Derzhavin
33 Internatsionalnaya St., Tambov, Russian Federation, 392000
E-mail: malaschonok@ya.ru

We discuss the problem of constructing an effective algorithm for computing the inverse matrix for an integer matrix. One of the way, for obtaining the inverse matrix, is based on the matrix Smith form. Known probabilistic algorithm can find the Smith form with the computational bit complexity which has cubic dependence of the matrix sizes. We propose a deterministic extension of this approach to calculating the inverse matrix.
Key words: Smith form; Exact Computations; Matrix Inversion

Introduction

We discuss the problem of constructing an effective algorithm for the integer matrix inversing.

It is known that the inverse of an integer matrix and other problems of linear algebra over a commutative domain are performed with the complexity of matrix multiplication. If you solve these problems in integers using modular arithmetic, the complexity in the bit-operations increased by n times. Where n - is the size of the matrix [1-5].

In recent years, have been actively develop probabilistic algorithms. The best probabalistic algorithm for the integer matrix inversing was proposed by Arne Storiohanom [6]. This algorithm has complexity $\tilde{n}^3(log(||A|| + log||A^{-1}||))$ He proposed a probabilistic algorithm that with probability at least $1/2$ computes the inverse matrix for the non-singular integral matrix A size $n\ timesn$ and uses

$$\sim n^3(log||A|| + log||A^{-1}||)$$

bit operations. Here $||A|| = \max_{i,j}|A_{ij}|$ is the biggest coefficient of the matrix A , symbol sim is a missing factor

$$a\ log(n)^b(log\ log(||A||))^c,$$

and the numbers a , b , c – is some positive constants.

The best algorithm, which was known before, has a bit complexity

$$\sim n^{w+1}log||A||.$$

Random matrix with a high probability is well-conditioned and has $||A|| \approx ||A^{-1}||$. However, it may be that $||A^{-1}|| \approx n||A||$ for ill-conditioned matrix, for example, when $\det(A) = 1$. Thus, Storiohana algorithm allows us to calculate the inverse matrix faster for well-conditioned matrix ($\sim n^3 log||A||$). And this algorithm does not improve in the case of ill-conditioned matrix ($\sim n^4 log||A||$).

The central idea of this algorithm is to calculate the Smith form of the initial matrix as a sum of matrices of rank one.

Let $\mathbf{r} = \mathrm{rank}(A)$

$$\mathbf{snf}(A) = S = PAQ = Diag(s_1, s_2, ..., s_{\mathbf{r}}, 0, 0, ..., 0)$$

– is the Smith form of matrix A, P and Q – are unimodular matrices and $\forall_i\ s_i|s_{i+1}$. Then the expansion of Smith form can be written as follows

$$A = (s_1)c_1r_1 + (s_2)c_2r_2 + ... + (s_{\mathbf{r}})c_{\mathbf{r}}r_{\mathbf{r}}. \tag{1}$$

here $c_ir_i,\ i=1,2,..,r$ is an outer product of the column c_i by row r_i.

The existence of such an expansion follows directly from the following matrix identity.

Let $s_1 = gcd(A)$, w_1 and h_1 – is a row and column, satisfying the equation $w_1Ah_1 = s_1$ and let $r_1 = w_1A/s_1$, $c_1 = Ah_1/s_1$. Then we have the following matrix identity:

$$\begin{bmatrix} 1 & w_1 \\ -c_1 & I_n - c_1w_1 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & A \end{bmatrix} \begin{bmatrix} 1 & -r_1 \\ h_1 & I_n - h_1r_1 \end{bmatrix} = \begin{bmatrix} s_1 & 0 \\ 0 & A - s_1c_1r_1 \end{bmatrix}. \tag{2}$$

On the left side of equality both factors are the unimodular matrices. Hence the matrix A and $\mathbf{diag}(s_1, A - s_1c_1r_1)$ have the same Smith form. And we can easy to find them using this recursive identity: first, the matrix A, then the matrix $A_2 = A - s_1c_1r_1$, and so on.

As was shown in [1] for $\mathbf{r} = n$ the algorithm requires a $\sim n^3(log||A||)$ bit operations. We must each time taking random vector h_i, then the vector w_i we must find using the extended Euclidean algorithm, calculating $gcd(A_ih_i)$. The equality $gcd(A_ih_i) = gcd(A_i)$ will be true with very high probability.

The following is an algorithm for computing the inverse matrix, which has roughly the same complexity in operations on integer coefficients. Its bit complexity we have to evaluate in the future.

Computation of the inverse matrix

Suppose we have already constructed the decomposition of Smith form of the matrix A and calculated all the components s_i, w_i, h_i, r_i, c_i ($i = 1,2,..,\mathbf{r}$) in decomposition (1).

We will introduce other notations. We further denote by w_i, r_i, h_i, c_i matrix of size $n \times n$, all of whose elements are zero except for one i-th row, which is equal to w_i or r_i, or one i th column, which equals h_i or c_i, respectively. In the new notation Smith decomposition will be recorded in the same manner as in (1).

T H E O R E M. *Let* $A = (s_1)c_1r_1 + (s_2)c_2r_2 + ... + (s_{\mathbf{r}})c_{\mathbf{r}}r_{\mathbf{r}}$ *– be Smith decomposition for matrix* A, *of size* $n \times n$, $\mathbf{r} = \text{rank}(A)$, $r_1 = w_1A/s_1$, $c_1 = Ah_1/s_1$. *Let* S'_i *– be* $n \times n$ *matrix, which is different from zero only one diagonal element in the* i th *row, which is equal* s_i, *and let* $S_i = S'_1 + S'_2 + .. + S'_i$ ($1 \le i \le \mathbf{r}$) *and* $S_0 = 0$.

Let $F_i = I_n - c_iw_i$, $G_i = I_n - h_ir_i$, $A_i = (s_i)c_ir_i + ... + (s_{\mathbf{r}})c_{\mathbf{r}}r_{\mathbf{r}}$,

$$U_i = \begin{bmatrix} I_n & w_i \\ -c_i & F_i \end{bmatrix} \text{ and } V_i = \begin{bmatrix} I_n & -r_i \\ h_i & G_i \end{bmatrix} \quad (i = 1,..,\mathbf{r}.)$$

Then the following matrix identities hold for any integer k, $1\ lek\ le\ bfr(A) : k$, $1 \le k \le \mathbf{r}(A)$:

$$\begin{bmatrix} I_n & w_k \\ -c_k & F_k \end{bmatrix} \begin{bmatrix} S_{k-1} & 0 \\ 0 & A_k \end{bmatrix} \begin{bmatrix} I_n & -r_k \\ h_k & G_k \end{bmatrix} = \begin{bmatrix} S_k & 0 \\ 0 & A_{k+1} \end{bmatrix}, \tag{3}$$

$$U_kU_{k-1}\cdots U_1 = \begin{bmatrix} \mathbf{L}_k & \mathbf{W}_k \\ -\mathbf{C}_k & \mathbf{F}_k \end{bmatrix}, \ V_1V_2\cdots V_k = \begin{bmatrix} \mathbf{M}_k & -\mathbf{R}_k \\ \mathbf{H}_k & \mathbf{G}_k \end{bmatrix}, \tag{4}$$

$$\mathbf{W_r}A\mathbf{H_r} = S_{\mathbf{r}}, \tag{5}$$

in which the the notation used

$$\mathbf{F}_k = F_kF_{k-1}\cdots F_1, \quad \mathbf{G}_k = G_1G_2\cdots G_k$$

$\mathbf{W}_k = w_1 + w_2\mathbf{F}_1 + .. + w_k\mathbf{F}_{k-1}$, $\mathbf{C}_k = c_k + F_{k-1}(c_{k-1} + F_{k-2}(c_{k-2} + .. + F_1(c_1)..)$, $\mathbf{H}_k = h_1 + + \mathbf{G}_1 h_2 + .. + \mathbf{G}_{k-1}h_k$, $\mathbf{R}_k = (..(r_1)G_1 + .. + r_{k-2}G_{k-2}) + r_{k-1})G_{k-1} + r_k$, $\mathbf{L}_k = \mathbf{I}_n - (w_2\mathbf{C}_1 + + w_3\mathbf{C}_2 + .. + w_k\mathbf{C}_{k-1})$, $\mathbf{M}_k = \mathbf{I}_n - (\mathbf{R}_1 h_2 + \mathbf{R}_2 h_3 + .. + \mathbf{R}_{k-1}h_k)$

P R O O F. The identity (2), which is used in the first step of calculating Smith decomposition, can be written in the form in which it will look for the step i . At the same time, we extend it zero and unit elements. And besides, we will add a diagonal matrix S_{i-1} and $S_i = S_{i-1} + S_i'$ to the left and the right side. Here, obviously, the identity is retained since $c_i S_{i-1} = 0$. As a result, come to identity (3). We prove (4) by induction. For k = 1 the assertion is obvious. Suppose it is true for some $k \geq 1$. Let us prove the following equality

$$\begin{bmatrix} I_n & w_{k+1} \\ -c_{k+1} & F_{k+1} \end{bmatrix} \begin{bmatrix} \mathbf{L}_k & \mathbf{W}_k \\ -\mathbf{C}_k & \mathbf{F}_k \end{bmatrix} = \begin{bmatrix} \mathbf{L}_{k+1} & \mathbf{W}_{k+1} \\ -\mathbf{C}_{k+1} & \mathbf{F}_{k+1} \end{bmatrix}$$

Matrices $\mathbf{L}_k$ and $\mathbf{W}_k$ differ from the unit and zero matrices, respectively, only in the first k rows therefore $c_{k+1}\mathbf{L}_k = c_{k+1}$ and $c_{k+1}\mathbf{W}_k = 0$. This implies that: $\mathbf{F}_{k+1} = F_{k+1}\mathbf{F}_k$, $\mathbf{C}_{k+1} = c_{k+1} + + F_{k+1}\mathbf{C}_k$, $\mathbf{W}_{k+1} = \mathbf{W}_k + w_{k+1}\mathbf{F}_k, \mathbf{L}_{k+1} = \mathbf{L}_k - w_{k+1}\mathbf{C}_k$.

The second of the identities (4) can be proved similarly.

To prove the identity (5) applies to the original matrix A k times the equation (3) and use (4). A result we get

$$\begin{bmatrix} \mathbf{L}_k & \mathbf{W}_k \\ -\mathbf{C}_k & \mathbf{F}_k \end{bmatrix} \begin{bmatrix} 0 & 0 \\ 0 & A_k \end{bmatrix} \begin{bmatrix} \mathbf{M}_k & -\mathbf{R}_k \\ \mathbf{H}_k & \mathbf{G}_k \end{bmatrix} = \begin{bmatrix} S_k & 0 \\ 0 & A_{k+1} \end{bmatrix}$$

When $k = \ bfr$ from this equation, we obtain (5).

T H E O R E M. *Let all the conditions of Theorem 1 are satisfied, and* $\text{rank}A = n$. *Then there is a factorization for the the inverse matrix:*

$$A^{-1} = \mathbf{H}_n S^{-1} \mathbf{W}_n. \tag{6}$$

Here

$$\mathbf{H}_k = h_1 + \mathbf{G}_1 h_2 + .. + \mathbf{G}_{k-1}h_k,\ \mathbf{G}_k = G_1 G_2 \cdots G_k,\ G_i = I_n - h_i r_i, \tag{7}$$

$$\mathbf{W}_k = w_1 + w_2\mathbf{F}_1 + .. + w_k\mathbf{F}_{k-1},\ \mathbf{F}_k = F_k F_{k-1} \cdots F_1,\ F_i = I_n - c_i w_i. \tag{8}$$

The proof is reduced to the inversion of equality (5).

Complexity. Let us find a product

$$(I_n - h_1 r_1)(I_n - h_2 r_2) \cdots (I_n - h_{n-1} r_{n-1})$$

$$G_1 G_2 = (I_n - h_1 r_1)(I_n - h_2 r_2) = (I_n - h_1(r_1 - \mu_{12} r_2) - h_2 r_2)$$

$$G_1 G_2 G_3 = (I_n - h_1(r_1 - q_1 r_2) - h_2 r_2)(I_n - h_3 r_3) =$$

$$(I_n - h_1(r_1 - q_1 r_2) - h_2 r_2) - (h_3 r_3 - h_1(r_1 - q_1 r_2) h_3 r_3 - h_2 r_2 h_3 r_3) =$$

$$I_n - h_1(r_1 - \mu_{12} r_2 - (\mu_{13} - \mu_{12}\mu_{23}) r_3) - h_2(r_2 - \mu_{23} r_3) - h_3 r_3 =$$

$$I_n - (h_1 q_1 q_2 .. q_{n-1} + h_2 q_2 q_3 .. q_{n-1} - h_3 q_3 q_4 .. q_{n-1} ... + h_{n-1}) r_{n-1}$$

$$q_i = r_i h_{i+1}$$

–is the value of the element (i, i + 1) in the matrix (i = 1..n-2) To calculate the matrix $\mathbf{H}_n$ need to calculate each of its columns in accordance with (7). Column number k is equal to

$$G_1 G_2 \cdots G_{k-1} h_k.$$

We calculate it from right to left. We calculate the last product:

$$G_{k-1} h_k = (I_n - h_{k-1} r_{k-1}) h_k = h_k - h_{k-1}(r_{k-1} h_k)$$

To do this, the result of the scalar product of vectors $(r_{k-1} h_k)$ multiply by a column vector h_{k-1} and subtract from column h_{k-1} . In total, we performed 2n multiplications and additions as well. Continuing to go on like this, we calculate the entire column with the number k using $2(k-1)n$ operations. Since the number of columns is equal to n , it would take n^3 of operations for all calculations.

Similarly we can calculate the matrix $\mathbf{W}_n$.

Thus, if we know Smith decomposition of matrix A then the inverse matrix factorization can be obtained for n^3 of operations over coefficients.

The question of the bit complexity of such an algorithm is still open.

REFERENCES

1. *Malaschonok G.I.* Effective matrix methods in commutative domains. In: D. Krob, A.A. Mikhalev, A.V. Mikhalev (eds.) Formal Power Series and Algebraic Combinatorics // Springer, Berlin, 2000. P. 506–517.

2. *Akritas A.G., Malaschonok G.I.* Computations in Modules over Commutative Domain // Computer Algebra in Scientific Computing. Springer, Berlin, 2007. P. 11–23.

3. *Malaschonok G.I.* On computation of kernel of operator acting in a module // Tambov University Review. Series: Natural and Technical Sciences, 2008. V. 13. Iss. 1. P. 129–131.

4. *Malaschonok G.I.* Generalized Bruhat decomposition in commutative domains // International Workshop on Computer Algebra in Scientific Computing, LNCS 8136. Springer, Berlin, Heidelberg, 2013. P. 231–242.

5. *Malaschonok G., Scherbinin A.* Triangular Decomposition of Matrices in a Domain // Computer Algebra in Scientific Computing. LNCS 9301, Springer, Switzerland, 2015. P. 290–304.

6. *Storjohann Arne.* On the Complexity of Inverting Integer and Polynomial Matrices // Computational Complexity. 2015. V. 24. P. 777–821. DOI 10.1007/s00037-015-0106-7

ACKNOWLEDGEMENTS: The work is partially supported by the Russian Fund for Basic Research (project № 16-07-00420).

Received 19 October 2016 г

Malaschonok Gennadi Ivanovich, Tambov State University named after G.R. Derzhavin, Tambov, the Russian Federation, Doctor of Physics and Mathematics, Professor of the Functional Analysis Department, e-mail: malaschonok@ya.ru

UDC 519.6
DOI: 10.20310/1810-0198-2016-21-6-2042-2046

ИСПОЛЬЗОВАНИЕ ФОРМЫ СМИТА ДЛЯ ТОЧНОГО МАТРИЧНОГО ОБРАЩЕНИЯ

© **Г. И. Малашонок**

Тамбовский государственный университет им. Г.Р. Державина
392000, Российская Федерация, г. Тамбов, ул. Интернациональная, 33
E-mail: malaschonok@ya.ru

Обсуждается проблема построения эффективного алгоритма обращения целочисленной матрицы. Один из способов вычисления обратной матрицы опирается на предварительное вычисление матрицы Смита. Известен вероятностный алгоритм вычисления матрицы Смита с кубической зависимостью числа бит-операций от размеров матрицы. Предлагается некоторое детерминистское продолжением этого подхода для вычисления обратной матрицы.

Ключевые слова: форма Смита; символьные вычисления; обращение матриц

СПИСОК ЛИТЕРАТУРЫ

1. *Malaschonok G.I.* Effective matrix methods in commutative domains. In: D. Krob, A.A. Mikhalev, A.V. Mikhalev (eds.) Formal Power Series and Algebraic Combinatorics // Springer, Berlin, 2000. P. 506–517.
2. *Akritas A.G., Malaschonok G.I.* Computations in Modules over Commutative Domain // Computer Algebra in Scientific Computing. Springer, Berlin, 2007. P. 11–23.
3. *Malaschonok G.I.* On computation of kernel of operator acting in a module // Вестник Тамбовского университета. Серия Естественные и технические науки. Тамбов, 2008. Т. 13. Вып. 1. С. 129–131.
4. *Malaschonok G.I.* Generalized Bruhat decomposition in commutative domains // International Workshop on Computer Algebra in Scientific Computing, LNCS 8136. Springer, Berlin, Heidelberg, 2013. P. 231–242.
5. *Malaschonok G., Scherbinin A.* Triangular Decomposition of Matrices in a Domain // Computer Algebra in Scientific Computing. LNCS 9301, Springer, Switzerland, 2015. P. 290–304.
6. *Storjohann Arne.* On the Complexity of Inverting Integer and Polynomial Matrices // Computational Complexity. 2015. V. 24. P. 777–821. DOI 10.1007/s00037-015-0106-7

БЛАГОДАРНОСТИ: Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований (проект № 16-07-00420)

Поступила в редакцию 19 октября 2016 г.

Малашонок Геннадий Иванович, Тамбовский государственный университет им. Г.Р. Державина, г. Тамбов, Российская Федерация, доктор физико-математических наук, профессор кафедры функционального анализа, e-mail: malaschonok@ya.ru

Информация для цитирования:
Malaschonok G.I. Using the Smith form for the exact matrix inversion. *Vestnik Tambovskogo universiteta. Seriya Estestvennye i tekhnicheskie nauki – Tambov University Review. Series: Natural and Technical Sciences*, 2016, vol. 21, no. 6, pp. 2042-2046. DOI: 10.20310/1810-0198-2016-21-6-2042-2046