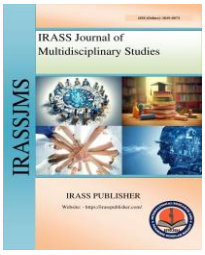
IRASS Journal of Multidisciplinary Studies
Abbriviate Title- IRASS J Mul Stud
ISSN *(Online) 3049-0073*
https://irasspublisher.com/journal-details/IRASSJMS
Vol-2, Iss-4 (April-2025)



Advanced Methods for Solving Linear Systems with Polynomial Coefficients: Algorithms and Complexity

Moumouni DJASSIBO WOBA*

Université Lédéa Bernard OUEDRAOGO (BURKINA FASO)

Corresponding Author **Moumouni DJASSIBO WOBA**

Université Lédéa Bernard OUEDRAOGO (BURKINA FASO)

Article History

Received: 20 / 12 / 2024

Accepted: 12 / 02 / 2025

Published: 02 / 04 /2025

Abstract: This article addresses the algorithmics of linear systems with polynomial coefficients in one variable, highlighting their similarities with the treatment of rational fractions. Emphasis is placed on the importance of efficiently utilizing fast matrix multiplication. We consider the linear system in the form $A(X)Y(X) = B(X)$, where A is an $n \times n$ matrix of polynomials with a non-zero determinant and B is a vector of polynomials. We establish clear notations to facilitate understanding of the concepts. The article also presents complexity estimates related to matrix multiplication and evaluation-interpolation. Key results include the computation of the series expansion of $A^{-1}B$ and the reconstruction of the coefficients of the rational fraction vector Y using Padé approximants. Newton's method is discussed for its efficiency in the case of polynomial matrices. Finally, a detailed analysis of resolution algorithms, including those of Storjohann, is provided, highlighting recent advances in computing the coefficients of rational solutions of linear systems. The algorithmics of linear systems with polynomial coefficients in one variable is very similar to that of rational fractions. Moreover, it is important to leverage fast matrix multiplication.

Keywords: Linear systems with polynomial coefficients; Fast matrix multiplication; Algorithmic resolution; Padé approximants; Polynomial matrices.

How to Cite: WOBA, M. D., (2025). Advanced Methods for Solving Linear Systems with Polynomial Coefficients: Algorithms and Complexity. *IRASS Journal of Multidisciplinary Studies*, *2(4),1-4.*

Introduction

The study of linear systems with polynomial coefficients in one variable presents unique challenges that require specific algorithmic approaches. These systems, often represented in the form $A(X)Y(X) = B(X)$, where A is an $n \times n$ matrix of polynomials, highlight complex algebraic structures and interactions between coefficients that differ from those of classical linear systems with rational coefficients. [1] One of the main similarities between these two domains lies in the treatment of rational fractions, where the development of efficient techniques has enabled significant advances. The importance of fast matrix multiplication is also emphasized, offering insights into the efficiency of resolution methods. In this article, we highlight essential notations and concepts to facilitate understanding of the solutions to these systems. We also propose complexity estimates concerning matrix multiplication and evaluation-interpolation techniques, which are crucial for optimizing the resolution process. The presented results include the computation of the series expansion of $A^{-1}B$ as well as the reconstruction of the coefficients of the rational fraction vector Y using Padé approximants, a method recognized for its ability to handle polynomial data. Furthermore, we will examine the efficiency of Newton's method in the context of polynomial matrices. Finally, a detailed discussion of resolution algorithms, including those developed by Storjohann, will be provided, highlighting recent advances that enable more efficient computation of the coefficients of rational solutions of linear systems with polynomial coefficients. We consider the linear system:

$$A(X)Y(X) = B(X)$$

where A and B are given and Y is unknown. A is an $n \times n$ matrix of polynomials, regular (with a non-zero determinant), and B is a vector of polynomials. Equivalently, one can view A (or B) as a polynomial with matrix (or vector) coefficients. To fix the notations, we assume $\deg A \leq d$ and $\deg B < d$. We denote:

- $\mathcal{M}_{m,k}(R)$: the set of $m \times k$ matrices with coefficients in R,
- $K[X]_d$: the set of polynomials of degree at most d with coefficients in the field K.

The function M is such that multiplication in $K[X]_d$ costs at most $M(d)$ operations in K. The exponent ω is such that multiplying two $n \times n$ matrices with coefficients in K costs $\mathrm{MM}(n) = O(n^\omega)$ operations in K. We will also need products of matrices in $\mathcal{M}_n(K[X]_d)$. In the most general case, this product is known to be executable in $\mathrm{MM}(nd) = O\big(n^\omega M(d)\big)$ operations in K, a bound that we will use in complexity estimates. When the field K contains enough points, this cost drops through evaluation-interpolation to $O(nd + n^2 M(d)\log d)$; under the same conditions, by choosing points in geometric progression, this complexity can be further reduced to $O\big(nd + n^2 M(d)\big)$.

Input

$A \in \mathcal{M}_n(K[X]_d)$, $B \in \mathcal{M}_{n,1}(K[X]_{d-1})$.

Output

The vector of rational fractions Y such that $AY = B$.

- Compute the series expansion of $A^{-1}B$ to precision $2nd$.
- Reconstruct the coefficients of Y using Padé approximants. [2]

From Series to Rational Solutions

A first observation is that the structure of the sought solution is given by Cramer's formulas.

lemma

The system has a solution whose coordinates are rational fractions. Its numerators and denominators have degrees bounded by $nd - 1$ and nd.

Proof. The system can be rewritten as:

$$A_1 y_1 + \cdots + A_n y_n = B,$$

where y_i is the i-th coordinate of Y and A_i is the i-th column of A. Cramer's formula:

$$\det(A_1, \ldots, A_{i-1}, B, A_{i+1}, \ldots, A_n) = y_i \det(A),$$

is obtained by substituting B with its value in the left-hand side and expanding the determinant. Thus, y_i is the quotient of determinants of matrices belonging to $\mathcal{M}_n(K[X]_d)$, which necessarily have a degree at most $nd - 1$ for the numerator and nd for the denominator. □

Algorithm 1 for resolution is justified by the quasi-optimal complexity of Padé approximants. In all subsequent algorithms, it is the search for a series solution that will dominate the complexity. [3]

Development as a Rational Fraction

Newton's method works for any invertible matrix of series. With this algorithm, computing the rational fraction requires $O\big(nM(nd)\big)$ operations in K. It is possible to improve efficiency when the matrix is a matrix of polynomials. The basis of this improvement is contained in the following lemma.

lemma

Let $A(X) \in \mathcal{M}_n(K[X]_d)$ and $B(X) \in \mathcal{M}_{n,m}(K[X]_{d-1})$, with A invertible. For all k, there exists a matrix $B_k \in \mathcal{M}_{n,m}(K[X]_{d-1})$ such that:

$$A^{-1}B = a_0 + a_1 X + \cdots + a_{k-1}X^{k-1} + X^k A^{-1} B_k,$$

where $a_i \in \mathcal{M}_{n,m}(K)$.

Input

A, B, k, and $S = A^{-1} \bmod X^k$.

Output

$a_0, \ldots, a_{k-1}$ and B_k defined by equation (11.2).

1. Compute $SB =: a_0 + \cdots + a_{k-1}X^{k-1} \bmod X^k$.
2. Compute $B_k = \big(B - A(a_0 + \cdots + a_{k-1}X^{k-1})\big)/X^k$.
3. Return $a_0, \ldots, a_{k-1}, B_k$.

Proof. If the a_i are the coefficients of the series expansion of $A^{-1}B$, then:

$$B - A(a_0 + \cdots + a_{k-1}X^{k-1})$$

is a matrix of $\mathcal{M}_{n,m}(K[X]_{d+k-1})$ which, by construction, is divisible by X^k, hence the lemma. □

This result translates into Algorithm 2. The following proposition estimates the cost of this algorithm. [4]

Proposition

Let $A \in \mathcal{M}_n(K[X]_d)$ with $A(0)$ invertible, then we can compute the first Nd coefficients of A^{-1} in $O(nNM(d)d)$ operations in K. For $B \in \mathcal{M}_{n,1}(K[X]_{d-1})$, we can compute the first Nd coefficients of $A^{-1}B$ in $O(n^2 NM(d)d)$ operations in K.

Proof. The algorithm first computes the inverse $S = A^{-1} \bmod X^d$ using Newton's algorithm, in $O\big(\mathrm{MM}(nd)\big) = O\big(nM(d)\big)$ operations in K. Then, we apply the above algorithm N/d times with $k = d$ to compute d coefficients and a new $B_{(i+1)d}$ at each iteration. If we start with $B_0 = I$, the result provides the first N coefficients of A^{-1}; if $B_0 = B$, then we obtain the coefficients of $A^{-1}B$. The two steps of the algorithm (with $k = d$) cost $\mathrm{MM}(nd) = O\big(nM(d)\big)$ operations in K if B has n columns, $O\big(n^2 M(d)\big)$ if it has only one. With this algorithm, computing the rational fraction solution of (1) requires $O\big(n^3 M(d)\big)$ operations in K. [5] □

Storjohann's Algorithm

Storjohann's algorithm allows computing the first N coefficients of the series expansion of $A^{-1}B$ in $O\big(n^{\omega-1} N \log NM(d)\big)$ operations in K. If $\omega < 3$, this quantity grows with n slower than the size of the inverse A^{-1}, which is therefore not computed entirely. Thus, solving the following linear system has a cost of $O\big(nM(d)\log(nd)\big)$ operations in K.

Input

A, B, k, and $S = [A^{-1}]_{2d}^{2k-2d+1}$.

Output

B_k defined by equation (11.2).

- Compute $U := [SB]_{k-d}^{d-1}$.
- Compute $B_k := \frac{[B - AU]_{k-d}^{d-1}}{X^k}$.
- Return B_k.

This method relies on a "divide and conquer" technique, and on grouping intermediate calculations to replace groups of n matrix-vector products with matrix-matrix products. To begin, we can modify the algorithm for developing $A^{-1}B$ from the previous section to compute B_k without calculating all the coefficients $a_0, \ldots, a_{k-1}$. The input required is smaller, and the complexity is lower when k is large compared to d. For a series $V = v_0 + v_1 X + \cdots$, we denote:

$$[V]_a^b := v_a X^a + \cdots + v_{a+b} X^{a+b},$$

with the convention that coefficients of negative index of V are zero.

Proof

To prove the correctness of this algorithm, we must ensure that series truncations are sufficient to compute the same polynomial B_k as before. For the first step of the algorithm, it suffices to observe that since B has degree at most $d-1$, the coefficient of X^i in $A^{-1}B$ depends only on $[A^{-1}]_{i-d+1}^{d+1}$, for all i. Therefore, the coefficients calculated by this first step are the same as the previous a_i, for $i = k-d, \dots, k-1$. Next, we need to compute $[X^k B_k]_{k-d}^{d-1}$. Extracting the coefficients from equation (2) gives:

$$[X^k B_k]_{k-d}^{d-1} = [B - A(a_0 + \cdots + a_{k-1}X^{k-1})]_{k-d}^{d-1},$$

which concludes the proof. An important observation regarding this algorithm is that the same polynomial S can be used to compute B_{i+k} for all i. The idea is then to group several vectors B_i and compute the corresponding B_{i+k} using matrix-matrix products. Note the resemblance between this idea and that of the Keller-Gehrig algorithm. Thus, if we know $B_0, B_{2m}, \dots, B_{2sm}$ and $[A^{-1}]_{2d}^{2k-2d+1}$, then computing the next vectors $B_{(2s+1)m}$ requires only $O\big(n^{\omega-1}sM(d)\big)$ operations in K. By iterating this idea, starting from B_0 and assuming known $[A^{-1}]_{2d}^{2k-2d+1}$ for $k = 0, \dots, \log(Nd) =: k_{\max}$, we first obtain $B_0, B_{2^{k_{\max}}}$, then at the next step $B_0, B_{2^{k_{\max}}-1}, B_{2^{k_{\max}}}, B_{3\cdot 2^{k_{\max}}-1}$, and so on until computing the entire sequence $B_0, B_d, B_{2d}, \dots$. [6]

Storjohann's Algorithm

Input

$S = [A^{-1}]_0^{d-1}$, $T = [A^{-1}]_{2k-2d+1}^{2d-2}$ and B_{2k-d} defined by (2) with $B = I$.

Output

B_{2k+1-d} and $[A^{-1}]_{2k+1-2d+1}^{2d-2}$.

1. Compute $[A^{-1}]_{2k+1-2d+1}^{2d-2} = [TB_{2k-d}]_{2k+1-2d+1}^{2d-2}$.
2. Compute $B_{2k+1-d} := [ATB_{2k-d}]_{2k+1-d}^{d-1}$.
3. Compute $[A^{-1}]_{2k+1-2d+1}^{d-1} = [X^{2k+1-d}B_{2k+1-d}S]_{2k+1-d}^{d-1}$.
4. Return B_{2k+1-d} and $[A^{-1}]_{2k+1-2d+1}^{2d-2}$.

Algorithm – Development of A^{-1} – Power-of-two Indices

We obtain B_{Nd} in $O(k_{\max}n^{\omega-1}NM(d)d)$ operations in K. By then multiplying these vectors by $[A^{-1}]_0^d$, we finally obtain the first N coefficients of $A^{-1}B$ for the same cost. It remains to see how to compute the $[A^{-1}]_{2k-2d+1}^{2d-2}$. Again, the starting point is the identity, with $B = I$, $k = m$, and $k = p$:

$$\begin{aligned} A^{-1} &= a_0 + \cdots + a_{m-1}X^{m-1} + X^m A^{-1}B_m, \\ &= a_0 + \cdots + a_{m-1}X^{m-1} \\ &\quad + X^m\big(a_0 + \cdots + a_{p-1}X^{p-1} + X^p A^{-1}B_p\big)B_m. \end{aligned}$$

The second line is obtained by substituting the value of A^{-1} given by the first line with $m = p$. These equations imply for all $m + p \le d$:

$$\begin{aligned} [A^{-1}]_{m+p}^{d-1} &= \big[A[A^{-1}]_{2d-2}^{p-2d+1}B_m\big]_{m+p}^{d-1}X^p, \\ &= \big[[A^{-1}]_{p+d-2}^{d-1}B_m\big]_{m+p}^{d-1}. \end{aligned}$$

The Algorithm follows using this identity with $m = 2k-d$, $p = 2k$, and $\ell = d-1$. To summarize, these algorithms lead to the following result. [7]

Theorem Storjohann, 2002

Let A be an $n \times n$ polynomial matrix of degree d with $A(0)$ invertible and B a polynomial vector of degree at most $d-1$, then we can compute the numerator and denominator of $A^{-1}B$ in $O\big(nM(d)\log(nd)\big)$ operations in K. [8]

Theorem Storjohann, 2005

Let A be an $n \times n$ invertible matrix containing integers of binary size at most b and let B be an $n \times 1$ vector containing integers of size $O(b)$. We can compute the numerator and denominator of $A^{-1}B$ in $O\big(n\log(n)M(d)\log(d)\big)$ binary operations, where $d = b + \log n$. [9]

Conclusion

This article explored the algorithmics of linear systems with polynomial coefficients in one variable, highlighting innovative and efficient approaches for their resolution. By considering the system of the form $A(X)Y(X) = B(X)$, we were able to establish significant links between these systems and the treatment of rational fractions, thus illustrating the relevance of these concepts in the field of computational algebra. We emphasized the crucial importance of fast matrix multiplication, as well as evaluation-interpolation techniques, which play a central role in optimizing resolution algorithms. The obtained results, such as the series expansion of $A^{-1}B$ and the reconstruction of the coefficients of the rational fraction vector Y using Padé approximants, demonstrate the power of the proposed methods. Furthermore, the discussion around Newton's method revealed its efficiency in the context of polynomial matrices, highlighting promising avenues for future research. The analysis of resolution algorithms, particularly those developed by Storjohann, highlights recent advances in the field and paves the way for further improvements in computing solution coefficients. In summary, we hope that the results and methods presented in this article will provide a solid foundation for future work and contribute to the advancement of algorithmic techniques in the treatment of linear systems with polynomial coefficients.

References

1. Dixon, J. D. (1982). Exact solution of linear equations using p-adic expansions. *Numerische Mathematik*, *40*(1), 137-141.
2. Giorgi, P., Jeannerod, C. P., & Villard, G. (2003, August). On the complexity of polynomial matrix computations. In *Proceedings of the 2003 international symposium on Symbolic and algebraic computation* (pp. 135-142).
3. Gupta, S., Sarkar, S., Storjohann, A., & Valeriote, J. (2012). Triangular x-basis decompositions and derandomization of linear algebra algorithms over K [x]. *Journal of Symbolic Computation*, *47*(4), 422-453.
4. Jeannerod, C. P., & Villard, G. (2005). Essentially optimal computation of the inverse of generic polynomial matrices. *Journal of Complexity*, *21*(1), 72-86.

5. Kaltofen, E., & Villard, G. (2005). On the complexity of computing determinants. *Computational complexity*, *13*(3), 91-130.

6. Moenck, R. T., & Carter, J. H. (1979). Approximate algorithms to derive exact solutions to systems of linear equations. In *Symbolic and Algebraic Computation: EUROSM'79, An International Symposium on Symbolic and Algebraic Manipulation, Marseille, France, June 1979 2* (pp. 65-73). Springer Berlin Heidelberg.

7. Pauderis, C., & Storjohann, A. (2012, July). Deterministic unimodularity certification. In *Proceedings of the 37th International Symposium on Symbolic and Algebraic Computation* (pp. 281-288).

8. Storjohann, A. (2002, July). High-order lifting. In *Proceedings of the 2002 international symposium on Symbolic and algebraic computation* (pp. 246-254).

9. Storjohann, A. (2003). High-order lifting and integrality certification. *Journal of Symbolic Computation*, *36*(3-4), 613-648.