Computing Reciprocals of Bivariate Power Series

Markus Bläser

Institut für Theoretische Informatik, Med. Universität zu Lübeck
Wallstr. 40, 23560 Lübeck, Germany
blaeser@tcs.mu-luebeck.de

Abstract. We consider the multiplicative complexity of the inversion and division of bivariate power series modulo the "triangular" ideal generated by all monomials of total degree $n+1$. For inversion, we obtain a lower bound of $\frac{7}{8}n^2 - O(n)$ opposed to an upper bound of $\frac{7}{3}n^2 + O(n)$. The former bound holds for all fields with characteristic distinct from two while the latter is valid over fields of characteristic zero that contain all roots of unity (like e.g. $\mathbb{C}$). Regarding division, we prove a lower bound of $\frac{5}{4}n^2 - O(n)$ and an upper bound of $3\frac{5}{6}n^2 + O(n)$. Here, the former bound is proven for arbitrary fields whereas the latter bound holds for fields of characteristic zero that contain all roots of unity.
Similar results are obtained for inversion and division modulo the "rectangular" ideal (X^{n+1}, Y^{n+1}).

1 Introduction

The question how many *essential* multiplications and divisions are needed to compute the reciprocal of a power series is an intriguing and challenging problem in algebraic complexity theory. Here, "essential" means that additions, subtractions, and scalar multiplications are free of costs. (We give a formal definition of multiplicative complexity below.) In the literature, mainly the inversion of *univariate* power series is discussed. In the present work, we investigate the multiplicative complexity of inversion of *bivariate* power series. More precisely, given a power series $A = \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} a_{i,j}X^iY^j$ (with indeterminate coefficients) we want to determine the number of essential operations necessary and sufficient to compute the coefficients $b_{i,j}$ with $i+j \le n$ of $A^{-1} = \sum_{i=0}^{\infty}\sum_{j=0}^{\infty} b_{i,j}X^iY^j$ from the coefficients $a_{i,j}$ with $i+j \le n$.

The reason why to investigate this problem is threefold: first, the problem of computing reciprocals of bivariate power series is an algebraic modelling of the inversion of univariate power series with arbitrary precision numbers as coefficients. The first indeterminate X refers to the original indeterminate whereas the second indeterminate Y models the precision management. Further details can be found in [8]. Second, the results obtained in the present work may shed a new light on the univariate case, a major open problem in algebraic complexity theory (see e.g. [4, Problem 2.7]). Third, our new results complement nicely the results in [2] on the multiplication of bivariate power series.

What is known in the univariate case? Let C_n denote the number of essential operations necessary and sufficient to compute the first $n+1$ coefficients of

$A^{-1} = b_0 + b_1X + b_2X^2 + \cdots$ from the first $n+1$ coefficients of the power series $A = a_0 + a_1X + a_2X^2 + \cdots$ (where we again consider $a_0, a_1, a_2, \ldots$ as indeterminates over some ground field k). If the characteristic of the ground field k differs from two, then we have

$$n - 2 + \lceil (n-1)/2 \rceil \le C_n \le 3\tfrac{3}{4}n. \tag{1}$$

If the characteristic of k is two, then

$$n + 1 \le C_n \le 3\tfrac{1}{4}n. \tag{2}$$

Kalorkoti [5] attributes the lower bound in (1) to Hanns-Jörg Stoß. To prove this bound, the problem of computing the lower $n-1$ coefficients of the square of a polynomial of degree $n-2$ is reduced to the problem of computing $b_0, \ldots, b_n$ using techniques due to Strassen [12]. (The lower bound for computing the lower $d+1$ coefficients of the square of a polynomial of degree d given in [5] can be improved by one, $d + \lceil (d+1)/2 \rceil$ is the true complexity of this problem.) The upper bound in (1) is due to Kung [6]. Kung actually states the upper bound $4n - \log n$, but a more careful analysis of his algorithm by Schönhage yields the stated upper bound, see again [5] for a discussion.

The lower bound in (2) follows from a standard linear independence argument, the upper bound is again due to Kung. The improvement compared to (1) follows from the fact that computing the square of a polynomial is cheap over fields of characteristic two.

Until today, the bounds in (1) and (2) have withstood any attempts of improvement (including those of the author, see however [3] for an improvement in a restricted setting).

1.1 Computations, Costs, and Complexity

In the present section, we briefly introduce the concept of multiplicative complexity. For a detailed description, the reader is referred to [4]. We start with giving a formal definition of computations.

Definition 1. *Let $x_1, \ldots, x_m$ be indeterminates over a field k and $f_1, \ldots, f_n \in K = k(x_1, \ldots, x_m)$.*

1. *A sequence $\beta = (w_1, \ldots, w_\ell)$ of rational functions $w_1, \ldots, w_\ell \in K$ is called a computation over K, if for all $1 \le \lambda \le \ell$ there are $i, j < \lambda$ such that $w_\lambda = w_i \circ w_j$ with $\circ \in \{+, -, *, /\}$ and $w_j \ne 0$ if $\circ = /$ or $w_\lambda = \alpha \cdot w_i$ with $\alpha \in k$ or $w_\lambda \in k \cup \{x_1, \ldots, x_m\}$.*
2. *The sequence is called a computation for $f_1, \ldots, f_n$, if in addition $f_1, \ldots, f_n \in \{w_1, \ldots, w_\ell\}$.*

The next step is to define the costs of a computation.

Definition 2. *Let k be a field, $x_1, \ldots, x_m$ indeterminates over k, and $\beta = (w_1, \ldots, w_\ell)$ a computation over $K = k(x_1, \ldots, x_m)$.*

1. *The costs γ_λ in the λth step of β are defined as*

$$\gamma_\lambda = \begin{cases} 0 & \text{if there are } i, j < \lambda \text{ such that } w_\lambda = w_i \pm w_j, \\ & \text{or } w_\lambda = \alpha \cdot w_i \text{ with } \alpha \in k, \\ & \text{or } w_\lambda \in k \cup \{x_1, \ldots, x_m\}, \\ 1 & \text{otherwise.} \end{cases}$$

2. *The* costs *$\Gamma(\beta)$ of β are $\Gamma(\beta) = \sum_{\lambda=1}^{\ell} \gamma_\lambda$.*

In the above definition, we count only those steps where the rational function w_λ can solely be expressed as the product or quotient of two elements with smaller index. Such a step is called an *essential* multiplication or division, respectively. This measurement of costs is also called *Ostrowski measure.*

We proceed with defining the complexity of a set of rational functions.

Definition 3. *Let $x_1, \ldots, x_m$ be indeterminates over a field k and $f_1, \ldots, f_n \in K = k(x_1, \ldots, x_m)$. The* multiplicative complexity *$C(f_1, \ldots, f_n)$ of $f_1, \ldots, f_n$ is defined by $C(f_1, \ldots, f_n) = \min\{\Gamma(\beta) \mid \beta$ is a computation for $f_1, \ldots, f_n\}$.*

1.2 Our Results

After settling the model of computation, we are able to formulate our problem precisely: let $A(X, Y) = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} a_{i,j} X^i Y^j$ be a bivariate power series with indeterminates over some ground field k as coefficients. Let $B(X, Y) = \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} b_{i,j} X^i Y^j$ be the power series defined by $A \cdot B = 1$. (Note that A is invertible, since $a_{0,0}$ is an indeterminate.) The coefficients of B are rational functions in the coefficients of A and only $a_{0,0}$ occurs in the denominators of these rational functions. If we write $A = a_{0,0} - A_1$, then

$$B = \frac{1}{a_{0,0}} + \frac{1}{a_{0,0}^2} A_1 + \frac{1}{a_{0,0}^3} A_1^2 + \cdots \quad . \tag{3}$$

Power series are infinite objects but in our model (and in today's computers) we can only deal with a finite number of rational functions. In the univariate case, computing modulo (X^{n+1}) is the canonical choice. In the bivariate case, however, there are more than one meaningful choice. We here mainly consider the case of computing modulo the "triangular" ideal $I_n := (X^{n+1}, X^n Y, X^{n-1} Y^2, \ldots, Y^{n+1})$ generated by all monomials of degree $n + 1$. In this case, we only have to consider terms up to $A_1^n / a_{0,0}^{n+1}$ in (3). Another possibility is to take the "rectangular" ideal (X^{n+1}, Y^{n+1}). We will discuss this case in Section 5 briefly.

In the above language, our problem reads as follows: find good upper and lower bounds for $C(\{b_{i,j} \mid i + j \leq n\})$. Note that the rational functions $b_{i,j}$ with $i + j \leq n$ only depend on the $a_{i,j}$ with $i + j \leq n$. Therefore, inverting power series modulo I_n is equivalent to computing the reciprocal of the generic element $\sum_{i+j\le n} a_{i,j}X^iY^j$ in the local k-algebra $\mathcal{T}_n := K[X,Y]/I_n$ where $K = k(a_{i,j} \mid i+j \le n)$. For convenience, we do not always want to choose names for the coefficients of the power series explicitly. Therefore, we assume the following convention about the fields k and K:

The field k denotes the underlying ground field and multiplication with elements from k is for free. On the other hand, the field K is obtained by adjoining to k the coefficients (which are indeterminates over k) of the power series we want to manipulate.

So if for instance we speak about the multiplicative complexity of the multiplication in $K[X,Y]/I_n$, we mean the multiplicative complexity of computing the coefficients of the product AB of two "truncated" power series $A = \sum_{i+j\le n} a_{i,j}X^iY^j$ and $B = \sum_{i+j\le n} b_{i,j}X^iY^j$ from the coefficients of A and B. The field K is obtained by adjoining all $a_{i,j}$ and $b_{i,j}$ with $i+j \le n$ to k. (Note that the coefficients of AB are rational functions in the coefficients of A and B.)

Our main results concerning inversion are subsumed in the following theorems:

Theorem 1. *For fields k of characteristic zero that contain all roots of unity, the multiplicative complexity of inverting in $\mathcal{T}_n$ is at most $\frac{7}{3}n^2 + O(n)$.*

Theorem 2. *The multiplicative complexity of inversion in $\mathcal{T}_n$ has the lower bound $\frac{1}{2}n^2 + \frac{3}{2}n + 1$. Over fields of characteristic distinct from two this can be improved to $\frac{7}{8}n^2 - O(n)$.*

Building up on these results, we obtain the following bounds for the division of power series:

Theorem 3. *Let k be a field of characteristic zero that contains all roots of unity. The multiplicative complexity of division in $\mathcal{T}_n$ is at most $3\frac{5}{6}n^2 + O(n)$.*

Theorem 4. *The multiplicative complexity of division in $\mathcal{T}_n$ is bounded from below by $\frac{5}{4}n^2 - O(n)$.*

For the upper bound in Theorem 1, we use generalizations of the fast multiplication algorithms for $\mathcal{T}_n$ developed in [2]. These algorithms only work over fields of characteristic zero that contain all roots of unity. (More precisely, we only need that the multiplicative complexity in the algebra $K[X]/(X^N - 1)$ equals N for certain values of N. The preceding more restrictive formulation avoids some lengthy case discussions.) Therefore, we make the following convention:

Throughout the remainder of this work, we assume that the underlying ground field k has characteristic zero and contains all roots of unity.

So for instance, $k = \mathbb{C}$ would be an appropriate choice. All lower bounds proven in this work remain valid over fields with characteristic distinct from two, in the case of inversion, and over arbitrary fields, in the case of division. All upper bounds concerning inversion and division can be carried over to arbitrary fields by utilizing the more expensive evaluation–multiplication–interpolation scheme for multiplication in $\mathcal{T}_n$ at the price of larger constants.

2 Multiplying Bivariate Power Series

The inversion of power series is usually done by an iterative process. In the nonscalar model, the costing operations in each iteration are caused by polynomial multiplications of various kinds. In this section, we therefore study the multiplication in $\mathcal{T}_n$ extensively.

2.1 Multiplication in $K[X]/(X^N - 1)$

In what follows, we will reduce the multiplication in $\mathcal{T}_n$ to the multiplication in the algebra $\mathcal{U}_N := K[X]/(X^N - 1)$ for some suitable N. Therefore, we first examine the problem of multiplying two univariate polynomials modulo $X^N - 1$ and state some well known results concerning its complexity (see also [4, Chap. 2.1]). We again assume that the coefficients of the elements we want to multiply are indeterminates over k and that K is obtained from k by adjoining these indeterminates.

Theorem 5 (see [4, Chap. 2.1]). *The multiplicative complexity of the multiplication in $\mathcal{U}_N$ equals N.*

Theorem 6. *Suppose we know M coefficients of the product $AB \mod X^N - 1$ of two polynomials A and B with indeterminate coefficients in advance. Then the coefficients of $AB \mod X^N - 1$ can be computed with $N - M$ essential multiplications.*

A standard linear independence argument show that the number $N - M$ of multiplications is also necessary.

2.2 Multiplication in $\mathcal{T}_n$

Let us turn to the multiplication in $\mathcal{T}_n$. Assume we are given two elements $A, B \in \mathcal{T}_n$. One way to multiply A and B is to evaluate both A and B at $2n^2 + 3n + 1$ suitably chosen points, which is free of costs, perform that many pointwise costing multiplications, obtain the coefficients of the (untruncated) product by interpolation, which is again free of costs, and finally throw away all coefficients belonging to monomials of total degree greater than n. By the Alder-Strassen theorem [1], the foregoing method is optimal (w.r.t. multiplicative complexity) if we want to compute the product of two univariate polynomials modulo X^{n+1}. But surprisingly, we can do better in the bivariate case, as is shown in [2]. In the remainder of this section, we present useful variations on and generalizations of this result.

The main idea of [2] is to reduce the multiplication of A and B to the multiplication of two elements A' and B' in $\mathcal{U}_N$ for some suitable N. We encode both factors A and B as long univariate polynomials by substituting $Y \mapsto X^{2n+2}$, i.e., $A'(X) = A(X, X^{2n+2})$ and $B'(X) = B(X, X^{2n+2})$. This substitution groups together the coefficients of all monomials with the same degree in Y and inserts some additional zero coefficients between these groups. If we now set

$N = n(2n+2) + n + 1$, we can read off the coefficients of $R = AB$ from the coefficients of the result $R' = A'B' \mod X^N - 1$. Borrowing the language of [8], we can depict this fact by a pattern

$$\begin{array}{l} \text{d d d d o o o o ' d d d o o o o o ' d d o o o o o o ' d o o o} \\ \hfill \text{for the factors } A', B' \\ \text{d d d d} * * * \text{o ' d d d} * * * \text{o o ' d d} * * * \text{o o o ' d} * * * / \\ \text{o o o o} * * * \text{o ' o o o o} * * \text{o o ' o o o o} * \\ \hfill \text{for the result } R', \end{array}$$

here for $n = 3$, $N = 28$. Lower coefficients are standing on the left, higher ones on the right. An entry "d" (data) marks a coefficient of the original polynomials A, B, and R, respectively, an entry "o" stands for a zero coefficient, and an "$*$" indicates "garbage", that is, a coefficient that is not needed to recover R. The quotes separate groups of monomials with the same degree in Y. The second line of the result is the "wraparound" produced by the -1 of $X^N - 1$. By inserting the right number of zeros, some of the garbage places are used twice while the coefficients of R are preserved. Moreover, we know $\frac{1}{2}n(n+1)$ coefficient of R' in advance, since they are zero. Thus the upper bound of $N - \frac{1}{2}n(n+1)$ essential multiplications follows from Theorem 6.

Theorem 7 (see [2, Thm. 6]). *The multiplicative complexity of the multiplication in $\mathcal{T}_n$ is at most $\frac{3}{2}n^2 + \frac{5}{2}n + 1$.*

A lower bound of $\frac{5}{4}n^2 - O(n)$ for the multiplicative complexity of the multiplication in $\mathcal{T}_n$ is proven in [2].

Since the coefficients of R are a subset of the coefficients of R', we immediately obtain the following more general theorem.

Theorem 8. *Suppose we know M coefficients of the product $AB \in \mathcal{T}_n$ of two polynomials A and B with indeterminate coefficients in advance. Then the coefficients of $AB \in \mathcal{T}_n$ can be computed with $\frac{3}{2}n^2 + \frac{5}{2}n + 1 - M$ essential multiplications.*

For any $m \leq n$, we may view $\mathcal{T}_m$ as a subspace of $\mathcal{T}_n$ just by filling the lacking coefficients with zeros. If say A is in $\mathcal{T}_m$ with $m \leq n$, then one moment's reflection shows that we will have an additional amount of $(n-m)(n+1)$ extra zeros in the result. (Another solution that is more suited for algorithmic purposes is to substitute $Y \mapsto X^{m+n+2}$ instead of $Y \mapsto X^{2n+2}$.) Thus, the below theorem follows.

Theorem 9. *Let $m \leq n$. Suppose we know M coefficients of the product $AB \in \mathcal{T}_n$ of $A \in \mathcal{T}_m$ and $B \in \mathcal{T}_n$ in advance. Then the coefficients of $AB \in \mathcal{T}_n$ can be computed with $\frac{1}{2}n^2 + mn + \frac{3}{2}n + m + 1 - M$ essential multiplications.*

2.3 Squaring in $\mathcal{T}_n$

The proof of the lower bound for the complexity of inversion will be reduced to the proof of a lower bound for squaring. Therefore, we provide a lower bound for the complexity of computing the square of an element in $\mathcal{T}_n$ in this section. Because the computation of squares is interesting on its own, we also provide upper bounds for this problem.

The straight forward method for squaring an element $A \in \mathcal{T}_n$ is of course to use the multiplication algorithm presented in the preceding section. But we can do better: let $h = \lceil (n-1)/2 \rceil$ and write $A = L + U$ where L contains only monomials of total degree less than or equal to h while U consists of all other monomials (of total degree greater than h). We have

$$A^2 = L^2 + 2LU = L(L + 2U) \qquad \text{in } \mathcal{T}_n.$$

Now we may plug in Theorem 9. This yields the following upper bound.

Theorem 10. *Suppose we know M coefficients of the square $A^2 \in \mathcal{T}_n$ of some $A \in \mathcal{T}_n$ with indeterminate coefficients in advance. Then the coefficients of $A^2 \in \mathcal{T}_n$ can be computed with $\frac{1}{2}n^2 + hn + \frac{3}{2}n + h + 1 - M \leq n^2 + 2n + 1 - M$ essential multiplications, where $h = \lceil (n-1)/2 \rceil$.*

Compared with the straight forward method, we save about $\frac{1}{2}n^2$ essential multiplications through the above construction.

Next we give a lower bound for the multiplicative complexity of squaring over fields of characteristic distinct from two.

The coefficients $b_{\mu,\nu}$ of A^2 are quadratic forms in the coefficients $a_{\mu,\nu}$ of A. In this case, according to Strassen [12], the fact that the multiplicative complexity of computing the $b_{\mu,\nu}$ from the $a_{\mu,\nu}$ is bounded from above by ℓ is equivalent to the existence of ℓ products $P_\lambda = U_\lambda \cdot V_\lambda$ of linear forms U_λ and V_λ in the $a_{\mu,\nu}$ such that

$$b_{i,j} \in \operatorname{lin}\{P_1, \ldots, P_\ell\} \qquad \text{for all } i, j \text{ with } i + j \leq n.$$

In other words, we may restrict ourselves to computations that contain only normalized multiplications and no divisions.

Theorem 11. *Over fields of characteristic distinct from two, computing the square of an element from $\mathcal{T}_n$ requires at least $n^2 - \frac{1}{2}h^2 + 3n - \frac{3}{2}h + 1$ essential operations, where $h = \lceil n/2 \rceil$.*

Proof. Consider a computation with ℓ essential operations that computes the coefficients $b_{\mu,\nu}$ of A^2 from the indeterminate coefficients $a_{\mu,\nu}$ of A. By the preceding considerations, we may restrict our attention to computations that have ℓ products of linear forms as its costing operations.

Now we exploit the so-called substitution method (see [7] or [4, Chap. 6]): we substitute one of the coefficients, say $a_{i,j}$, by a linear form in the remaining ones in such a way that one of the products P_λ is trivialized, that is, one of the linear forms U_λ or V_λ becomes zero after this substitution. Thereafter, we know that the multiplicative complexity of the original problem is at least one plus the multiplicative complexity of the quadratic forms obtained through this substitution. We may then repeat this process if desired. Of course, we can only trivialize one of the products by substituting $a_{i,j}$ if there is a product that actually depends on $a_{i,j}$. If one of the computed quadratic forms does, this is certainly the case.

Here, our substitution strategy looks as follows: we define an ordering on the $a_{\mu,\nu}$ by

$$(\mu,\nu) > (\mu',\nu') \;:\Longleftrightarrow\; \mu+\nu > \mu'+\nu' \text{ or } \mu+\nu = \mu'+\nu' \text{ and } \mu > \mu'$$

and

$$a_{\mu,\nu} > a_{\mu',\nu'} \;:\Longleftrightarrow\; (\mu,\nu) > (\mu',\nu').$$

We then substitute all monomials w.r.t. that ordering starting with $a_{n,0}$ and then stepping down until we reach $a_{0,h+1}$. In other words, we just follow the "diagonals" starting with the outermost and ending with the $(h+2)$th one.

To simplify the notations, we denote by $(s_1,t_1),(s_2,t_2),\dots,(s_r,t_r)$ the pairs $(n,0),(n-1,1),\dots,(0,h+1)$ (ordered w.r.t. "$>$"). For technical reasons, we also add an additional pair (s_0,t_0) with the property of being larger than all other of the above pairs. Let $b_{i,j}^{(\rho)}$, $0 \le \rho \le r$, denote the homomorphic image of $b_{i,j}$ after substituting $a_{s_1,t_1},\dots,a_{s_\rho,t_\rho}$.

Claim 1. The polynomial $b_{i,j}^{(\rho)}$ is a homogeneous polynomial of degree two in the $a_{\mu,\nu}$ with $(s_\rho,t_\rho) > (\mu,\nu)$.

Claim 2. If $(s_\rho,t_\rho) > (i,j)$ then $b_{i,j}^{(\rho)} = b_{i,j}$.

Claim 3. All monomials $\alpha \cdot a_{u,v}a_{u',v'}$ of $b_{i,j}^{(\rho)}$ fulfil $(i,j) \ge (u+u',v+v')$. (Here, "$\ge$" is the reflexive closure of "$>$".)

Claim 4. If $\alpha \cdot a_{u,v}a_{u',v'}$ is a monomial of $b_{i,j}^{(\rho)}$ but not of $b_{i,j}$, i.e., $\alpha \cdot a_{u,v}a_{u',v'}$ is the result of some substitution, then even $(i,j) > (u+u',v+v')$.

Claim 5. If $\alpha \cdot a_{p,q}a_{p',q'}$ is a monomial of $b_{i,j}$ with $(s_r,t_r) > (p,q)$ and $(s_r,t_r) > (p',q')$, then $\alpha \cdot a_{p,q}a_{p',q'}$ is a monomial $b_{i,j}^{(\rho)}$.

The first claim is obviously true, since we have substituted each a_{s_ρ,t_ρ} by a linear form in the remaining $a_{\mu,\nu}$ with $a_{s_\rho,t_\rho} > a_{\mu,\nu}$.

For the second claim, note that $b_{i,j}$ does not depend on $a_{\mu,\nu}$ if $(\mu,\nu) > (i,j)$. Therefore, $b_{i,j}$ is unaffected by the substitution of a_{s_ρ,t_ρ} if $(s_\rho,t_\rho) > (i,j)$. Thus, $b_{i,j}^{(\rho)} = b_{i,j}^{(\rho-1)} = \dots = b_{i,j}^{(0)} = b_{i,j}$ as long as $(s_\rho,t_\rho) > (i,j)$.

For the third, fourth and fifth claim, observe that if $(p,q) > (p',q')$, then also

$$(p+p'',q+q'') > (p'+p'',q'+q'') \qquad \text{for all } (p'',q''). \tag{4}$$

We prove these claims by induction in ρ: each monomial $\alpha \cdot a_{u,v}a_{u',v'}$ of $b_{i,j}^{(0)} = b_{i,j}$ fulfils $(u+u', v+v') = (i,j)$, thus the induction start is clear. Assume that the claims hold for some $\rho < r$. We obtain $b_{i,j}^{(\rho+1)}$ from $b_{i,j}^{(\rho)}$ by replacing $a_{s_{\rho+1},t_{\rho+1}}$ by a linear form in the remaining $a_{\mu,\nu}$ with $a_{s_{\rho+1},t_{\rho+1}} > a_{\mu,\nu}$. Let $\alpha \cdot a_{u,v}a_{s_{\rho+1},t_{\rho+1}}$ be monomial of $b_{i,j}^{(\rho)}$. The $(\rho+1)$th substitution transforms this monomial into a sum of monomials of the form $\alpha' \cdot a_{u,v}a_{\mu,\nu}$. By (4), $(u+s_{\rho+1}, v+t_{\rho+1}) > (u+\mu, v+\nu)$ proving the third and fourth claim. The monomials $\alpha \cdot a_{p,q}a_{p',q'}$ of of $b_{i,j}^{(\rho)}$ with $(s_r, t_r) > (p,q)$ and $(s_r,t_r) > (p',q')$ are not affected by the substitution, since the new monomials $\alpha' \cdot a_{u,v}a_{\mu,\nu}$ introduced by the substitution cannot cancel $\alpha \cdot a_{p,q}a_{p',q'}$, because $(p+p', q+q') = (i,j) > (u+\mu, v+\nu)$. Thus $\alpha \cdot a_{p,q}a_{p',q'}$ is also a monomial of $b_{i,j}^{(\rho+1)}$. This proves the fifth claim.

By the second claim, each of the r substitutions can be done in such a way that one costing product is killed, since $b_{s_{\rho+1},t_{\rho+1}}^{(\rho)} = b_{s_{\rho+1},t_{\rho+1}}$ depends on $a_{s_{\rho+1},t_{\rho+1}}$. Altogether, this kills $\frac{1}{2}n^2 - \frac{1}{2}h^2 + \frac{3}{2}n - \frac{3}{2}h$ products.

Since we only substitute indeterminates on the $(h+2)$th or higher diagonals, each $b_{i,j}^{(r)}$ still contains one of its original monomials after substituting, for instance $\alpha \cdot a_{\lceil i/2\rceil,\lceil j/2\rceil}a_{\lfloor i/2\rfloor,\lfloor j/2\rfloor}$ with $\alpha \in \{1,2\}$ by the fifth claim. By the third claim, no other $b_{i',j'}^{(r)}$ with $(i,j) > (i',j')$ can contain this monomial. Thus, the quadratic forms $b_{\mu,\nu}^{(r)}$ are still linearly independent and their multiplicative complexity is a least $\frac{1}{2}n^2 + \frac{3}{2}n + 1$. This proves the theorem. □

Over fields of characteristic two, the situation is quite different: squaring over fields of characteristic two is cheap and we can determine the multiplicative complexity exactly.

Let $A \in \mathcal{T}_n$ with indeterminate coefficients $a_{\mu,\nu}$. Over fields of characteristic two, we have

$$A^2 = \sum_{\mu=0}^{n}\sum_{\nu=0}^{n}\sum_{\substack{p+s=\mu\\ q+t=\nu}} a_{p,q}a_{s,t}X^{\mu}Y^{\nu} = \sum_{\mu=0}^{\lfloor n/2\rfloor}\sum_{\nu=0}^{\lfloor n/2\rfloor} a_{\mu,\nu}^2 X^{2\mu}Y^{2\nu}.$$

Thus we may compute the coefficients of A^2 just by squaring the appropriate coefficients of A. Since the nonzero coefficients of A^2 are linearly independent, this is optimal.

Theorem 12. *Over fields of characteristic two, the multiplicative complexity of squaring in $\mathcal{T}_n$ is $\frac{1}{2}h^2 + \frac{3}{2}h + 1$, where $h = \lfloor n/2\rfloor$. If we know M nonzero coefficients in advance, then the multiplicative complexity is $\frac{1}{2}h^2 + \frac{3}{2}h + 1 - M$.*

3 Inverting Bivariate Power Series

In this section, we present a second order iteration scheme (which is superior to third order here) for the inversion of bivariate power series. To avoid lengthy calculations, we state all following bounds as $c \cdot n^2 + O(n)$ with an explicitly given constant c. One should note that in each of these bounds, the constant hidden in the O-notation is rather small. Let A be a bivariate power series with indeterminate coefficients. Given a positive integer n, our task is to compute a polynomial B of total degree at most n such that

$$AB \equiv 1 \mod I_{n+1}.$$

Let $h = \lceil (n-1)/2 \rceil$ and suppose we have already computed a polynomial C of total degree at most h such that

$$AC \equiv 1 \mod I_{h+1}. \tag{5}$$

Squaring yields

$$A \cdot (-AC^2 + 2C) \equiv 1 \mod I_{n+1}.$$

Thus

$$B \equiv -AC^2 + 2C \mod I_{n+1}.$$

So we have to compute the coefficients of $-AC^2+2C$ (viewed as an element of $\mathcal{T}_n$). We proceed as follows: we first compute AC. By (5), we know the coefficients of all monomials of AC with total degree h or less. Thus by Theorem 9, the coefficients of AC can be computed with $\frac{7}{8}n^2 + O(n)$ essential multiplications. Next, we compute $C \cdot AC$. Again by (5), we know the coefficients of all monomials of $C \cdot AC$ with total degree h or less, they are simply the coefficients of C. Thus another $\frac{7}{8}n^2 + O(n)$ essential multiplications suffice to compute the coefficients of AC^2. From this, we may obtain the coefficients of B without any further costing operations. Let us analyze the number $E(n)$ of essential operations performed by this algorithm (regarded as a straight-line program after unrolling the recursive calls): we have $E(n) = E(h) + \frac{7}{4}n^2 + O(n)$ with $E(0) = 1$. This yields $E(n) = \frac{7}{3}n^2 + O(n)$ and therefore proves Theorem 1.

What can we say about lower bounds for the multiplicative complexity of inversion in $\mathcal{T}_n$? A standard linear independence argument shows that the number of the computed coefficients, $\frac{1}{2}n^2 + \frac{3}{2}n + 1$, gives a lower bound. Over fields of characteristic two, this is the best we know. Over other fields, we can further improve this. Let $a_{\mu,\nu}$ denote the indeterminate coefficients of the element A we want to invert. The coefficients of the inverse are polynomials in the $a_{\mu,\nu}$ with $(\mu, \nu) \neq (0,0)$ over $k(a_{0,0})$ which we take as our new ground field (i.e., $a_{0,0}$ becomes a scalar). Using the technique by Strassen [12], we can transform a computation for the coefficients of A^{-1} into a computation that computes only terms of degree two and has no more essential operations than the original computation. At quick look at (3) shows that this computation actually computes $k(a_{0,0})$-multiples of the coefficients of A_1^2! Now Theorem 11 implies the bound of Theorem 2. (The missing constant term of A_1^2 reduces the lower bound just by an additive amount of $O(n)$.)

4 Computing Quotients of Bivariate Power Series

Since $A/B = A \cdot B^{-1}$, Theorem 7 and Theorem 1 immediately yield the upper bound stated in Theorem 3.

For the lower bound, we once more exploit Strassen's technique [12]: let $a_{i,j}$ and $b_{i,j}$ denote the coefficients of A and B, respectively. We regard our computation as a computation over the new ground field $k(a_{0,0})$. Now we transform a given computation for the coefficients of B/A into a computation that computes only terms of degree two and has no more essential operations than the original computation. Moreover, this can be done in such a way that all costing operations are products of linear forms in the coefficients of A and B. The resulting computation now computes the coefficients of $B \cdot A_1/a_{0,0}^2$ in $\mathcal{T}_n$. Altogether, we have obtained a computation that merely multiplies two elements of $\mathcal{T}_n$ except for the missing constant term of A_1. Now the result of [2, Thm. 3] for the multiplication in $\mathcal{T}_n$ can be applied. (The missing constant term again reduces the lower bound by an additive amount of $O(n)$). This completes the proof of Theorem 4.

5 The Rectangular Case

In contrast to $\mathcal{T}_n$, in the rectangular case $\mathcal{R}_n := k[X,Y]/(X^{n+1}, Y^{n+1})$ a direct application of the iteration process in Section 3 does not work very satisfactory. The main reason is that only $(X^{n+1}, Y^{n+1})^2 = (X^{2n+2}, X^{n+1}Y^{n+1}, Y^{2n+2})$ holds, while we had $I_n^2 = I_{2n+1}$ before. It is more favourable to embed elements from $\mathcal{R}_n$ into $\mathcal{T}_{2n}$ and then use the techniques of the preceding sections. This yields a upper bound of $9\frac{1}{3}n^2 + O(n)$ for the inversion in $\mathcal{R}_n$. But we can do better: note that when embedding $\mathcal{R}_n$ into $\mathcal{T}_{2n}$ it does not matter how we choose the coefficients of the monomials with X–degree or Y–degree greater than n. So using the substitution lemma from [9, Lemma 2], we may substitute these coefficients (in the same order as proposed in Theorem 11) and we can kill $n^2 + n$ many products yielding the upper bound $8\frac{1}{3}n^2 + O(n)$.

Theorem 13. *For fields k of characteristic zero that contain all roots of unity, the multiplicative complexity of inverting in $\mathcal{R}_n$ has the upper bound $8\frac{1}{3}n^2+O(n)$.*

For a lower bound, we exploit essentially the same substitution strategy as in the triangular case. However, we substitute the coefficients of all monomials of total degree at least $d = \lceil \frac{2}{3}n \rceil$. This kills $(n+1)^2 - \frac{1}{2}d(d+1)$ many products. In contrast to the triangular case, we can now only conclude that at least $(n+1)^2 - \frac{1}{2}(2n-2d+2)(2n-2d+3)$ of the coefficients of the square are still linearly independent after substituting. This proves the following lower bound.

Theorem 14. *Over fields of characteristic distinct from two, $\frac{14}{9}n^2 - O(n)$ is a lower bound for the multiplicative complexity of inversion in $\mathcal{R}_n$.*

Building up on these results, we obtain the following bounds for the division by utilizing [2, Thms. 4 and 8].

Theorem 15. *Let k be a field of characteristic zero that contains all roots of unity. The multiplicative complexity of division in $\mathcal{R}_n$ is at most $11\frac{1}{3}n^2 + O(n)$.*

Theorem 16. *The multiplicative complexity of division in $\mathcal{R}_n$ has the lower bound $\frac{7}{3}n^2 - O(n)$.*

Acknowledgements

I would like to thank the referees for pointing out some inaccuracies in the preliminary version.

References

1. A. Alder and V. Strassen. On the algorithmic complexity of associative algebras. *Theoret. Comput. Sci.*, 15:201–211, 1981.
2. Markus Bläser. Bivariate polynomial multiplication. In *Proc. 39th Ann. IEEE Symp. on Foundations of Comput. Sci. (FOCS)*, pages 186–191, 1998.
3. Markus Bläser. On the number of multiplications needed to invert a monic power series over fields of characteristic two. Technical report, Institut für Informatik II, Universität Bonn, January 2000.
4. Peter Bürgisser, Michael Clausen, and M. Amin Shokrollahi. *Algebraic Complexity Theory.* Springer, 1997.
5. K. Kalorkoti. Inverting polynomials and formal power series. *SIAM J. Comput.*, 22:552–559, 1993.
6. H. T. Kung. On computing reciprocals of power series. *Numer. Math.*, 22:341–348, 1974.
7. Victor Ya. Pan. Methods for computing values of polynomials. *Russ. Math. Surv.*, 21:105–136, 1966.
8. Arnold Schönhage. Bivariate polynomial multiplication patterns. In *Proc. 11th Applied Algebra and Error Correcting Codes Conf. (AAECC)*, Lecture Notes in Comput. Sci. 948, pages 70–81. Springer, 1995.
9. Arnold Schönhage. Multiplicative complexity of Taylor shifts and a new twist of the substitution method. In *Proc. 39th Ann. IEEE Symp. on Foundations of Comput. Sci. (FOCS)*, pages 212–215, 1998.
10. Arnold Schönhage. Variations on computing reciprocals of power series. *Inf. Proc. Letters*, 74:41–46, 2000.
11. Malte Sieveking. An algorithm for division of power series. *Computing*, 10:153–156, 1972.
12. Volker Strassen. Vermeidung von Divisionen. *J. Reine Angew. Math.*, 264:184–202, 1973.
13. Volker Strassen. Algebraic complexity theory. In J. van Leeuven, editor, *Handbook of Theoretical Computer Science Vol. A*, pages 634–672. Elsevier, 1990.