Electronic Colloquium on Computational Complexity, Report No. 108 (2026)

Factoring Products of Sparse Irreducibles of Bounded Individual Degree via Rational Interpolation

Aminadav Chuyoon* Amir Shpilka*

Abstract

We design a deterministic algorithm that, given blackbox access to the product $f = \prod_{i=1}^{\ell} h_i$ of ℓ irreducible s-sparse n-variate polynomials of bounded individual degree d, over fields of characteristic zero, and more generally over fields of sufficiently large positive characteristic, recovers the h_i's and their multiplicities in time $\mathrm{poly}(n, (s\ell d)^d)$. For any constant $d > 2$, this is the first *polynomial-time* algorithm for this problem, resolving for the bounded-individual-degree regime an open question of Dutta, Sinhababu, and Thierauf (Random 2026). The previous best deterministic algorithm, due to the authors, runs in $\mathrm{poly}(n, d^d, s^{d \log \ell}, \ell^d)$ time, which is quasi-polynomial in ℓ.

The improvement is enabled by a new sparse rational interpolation theorem in the bounded-individual-degree setting, based on reconstructing the denominator from its logarithmic derivatives. Given blackbox access to rational functions $a_1/b, \ldots, a_N/b$ where the a_j's and b are s-sparse of individual degree at most d with $\gcd(a_1, \ldots, a_N, b) = 1$, we recover the a_j's and b in time $\mathrm{poly}(n, d!, s^d, N)$. In contrast to the rational interpolation theorem of Chuyoon-Shpilka (2026), our algorithm reconstructs the denominator directly and does not require a precomputed list of its irreducible factors.

*Blavatnik School of Computer Science and AI, Tel Aviv University, chuyoon1@mail.tau.ac.il, shpilka@tauex.tau.ac.il. This research was funded by the European Union (ERC, EACTP, 101142020). Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Research Council Executive Agency. Neither the European Union nor the granting authority can be held responsible for them.

ISSN 1433-8092

Contents

1 Introduction

1.1 Background and motivation

This paper continues the line of work, going back to von zur Gathen and Kaltofen [vzGK85], on factorization of *sparse* polynomials. An n-variate polynomial is called s-sparse if it is a sum of at most s monomials, and has *bounded individual degree* d if each variable appears in it with degree at most d. We write $\mathcal{P}(n, s, d)$ for the class of n-variate s-sparse polynomials of individual degree at most d, and $\mathcal{P}_{\text{Factors}}(n, s, d)$ for the class of their factors. Sparse polynomials form one of the most basic subclasses of polynomials in algebraic complexity: they are precisely those polynomials computed by polynomial-size $\Sigma\Pi$ circuits, and have received considerable attention from the perspectives of polynomial identity testing [KS01], reconstruction [BT88, GKS90], and factorization [vzGK85, Kal85a, Vol15, Vol17, BSV20, KRS24, DST24, BV25, BKR+25, CS26]. A central problem in this area is the *efficient deterministic factorization* of multivariate polynomials whose irreducible factors are sparse. In fact, this problem is interesting even in the bounded-individual-degree case. Concretely, Dutta, Sinhababu, and Thierauf [DST24] ask the following.

Question 1.1 ([DST24], Section 6, item 4)**.** *Given a blackbox computing the product of ℓ sparse irreducible polynomials $f_1, \ldots, f_\ell$ of bounded individual degree d, find the f_i's in deterministic polynomial time.*

This question is open even when $d = 3$. Several special cases have been resolved. Volkovich [Vol15] gave a polynomial-time algorithm for the case where all f_i's are multilinear, and in [Vol17] he gave an efficient deterministic algorithm for factoring multiquadratic sparse polynomials (i.e., when the product itself is multiquadratic and sparse). In [CS26] the authors solve the problem for $d = 2$. In [BSV20] the authors proved a sparsity bound on factors of $\mathcal{P}(n, s, d)$ polynomials and used it to obtain a deterministic quasi-polynomial-time algorithm for factoring sparse polynomials of bounded individual degree. Kumar, Ramanathan, and Saptharishi [KRS24] and Dutta, Sinhababu, and Thierauf [DST24] gave efficient deterministic reconstruction of constant degree factors of sparse polynomials. Recently, in [CS26] the authors obtained the following.

Theorem 1.2 ([CS26], Theorem 1.10)**.** *Let $\mathbb{F}$ be a field of characteristic zero or larger than $2d$. There is a deterministic $\mathrm{poly}(n, s^{d \log \ell}, (\ell d)^d)$-time algorithm which, given blackbox access to a product $f = \prod_{r=1}^{\ell} h_r$, where $h_1, \ldots, h_\ell \in \mathcal{P}(n, s, d)$ are irreducible, returns the h_r's. Over arbitrary fields the algorithm runs in $\mathrm{poly}(n, (d^2)!, s^{d \log \ell + d^3}, \ell^d)$ time.*

Observe that the running time of Theorem 1.2 is *quasi-polynomial* as it contains the factor $s^{d \log \ell}$. Theorem 1.2 therefore answers Question 1.1 only when $\ell = O(1)$. One of the main contributions of this paper is a deterministic polynomial-time algorithm for Question 1.1 when the individual degree d is constant.

An important ingredient in the algorithm of [CS26] is a solution to the sparse rational interpolation problem in the bounded-individual-degree case. This problem concerns the reconstruction of a rational function from blackbox access, given that its numerator and denominator are sparse polynomials. Surprisingly, the best known deterministic algorithm for the unrestricted degree version of this question runs in exponential time [GKS94]. In [CS26], the authors solved the following variants, where a list of candidate irreducible factors of the denominator is supplied as input:

Theorem 1.3 ([CS26, Theorem 5.2])**.** *Let $\mathbb{F}$ be a field of characteristic zero or larger than $2d$. Suppose we have blackbox access to rational functions $\{a_j f^j / b\}_{1 \le j \le N}$, where:*

(1) $a_j \in \mathcal{P}(n, s, d)$ for every j.

(2) f *is a product of* ℓ *polynomials in* $\mathcal{P}_{Factors}(n, s, d)$*, accessible as a blackbox.*

(3) $\mathcal{F}$ *is a set of* s*-sparse irreducible factors of* f*, supplied as input.*

(4) b *is an* s*-sparse factor of* f *whose irreducible factors are in* $\mathcal{F}$*.*

(5) $\gcd(a_1, \ldots, a_N, b) = 1$*.*

Then there is a deterministic algorithm that recovers $a_1, \ldots, a_N$ *and* b *in time* $\mathrm{poly}(n, d!, s^d, N, |\mathcal{F}|, \ell)$*.*

The factorization algorithm of [CS26] applies a somewhat weaker version of Theorem 1.3 recursively: at each level, a precomputed list $\mathcal{F}$ of *all* the s-sparse divisors of an auxiliary free-term polynomial f_0 is produced using recursion, and a divisibility test is then used to produce the denominator b. This enumeration is the source of the $s^{d \log \ell}$ factor in Theorem 1.2.

The second main result of this work is an efficient algorithm for the bounded-individual-degree sparse rational interpolation problem that requires no precomputed information about the factors of b. Theorem 1.5 eliminates the list $\mathcal{F}$ by reconstructing $b\left(G_{(m)}\right)$ directly from its *logarithmic derivatives*.

1.2 Our results

Our first main result resolves Question 1.1 in the bounded-individual-degree regime for any constant d.

Theorem 1.4 (**Factoring products of sparse irreducibles**)**.** *Let* $\mathbb{F}$ *be a field of characteristic zero. There is a deterministic* $\mathrm{poly}\left(n, (s\ell d)^d\right)$*-time algorithm which, given blackbox access to a product* $f = \prod_{i=1}^{\ell} h_i$*, where* $h_1, \ldots, h_\ell \in \mathcal{P}(n, s, d)$ *are (not necessarily distinct) irreducible polynomials, returns (scalar multiples of) the* h_r*'s together with their multiplicities.*

Comparing with Theorem 1.2, the $s^{d \log \ell}$ factor is replaced by s^d (by [CS26, Theorem 1.8], an (n, s, d)-sparse polynomial may have as many as s^d sparse divisors, which may explain this factor). In particular, this resolves Question 1.1 for constant d.

Theorem 1.4 is obtained from a sparse rational interpolation theorem for rational functions in which both numerator and denominator are from $\mathcal{P}(n, s, d)$. The rational interpolation theorem of [CS26, Theorem 5.2] (see Theorem 1.3) assumes as input a list $\mathcal{F}$ of s-sparse irreducible polynomials containing every irreducible factor of the denominator b. Our new theorem removes the assumption entirely.

Theorem 1.5 (**Bounded-individual-degree sparse rational interpolation**)**.** *Let* $\mathbb{F}$ *be a field of characteristic zero. Suppose we have blackbox access to rational functions of the form*

$$\frac{a_j(\mathbf{x})}{b(\mathbf{x})}, \qquad 1 \le j \le N,$$

where $a_1, \ldots, a_N, b \in \mathcal{P}(n, s, d)$ *and* $\gcd(a_1, \ldots, a_N, b) = 1$*. Then there is a deterministic algorithm that, in time* $\mathrm{poly}(n, d!, s^d, N)$*, recovers* $\lambda a_1, \ldots, \lambda a_N, \lambda b$ *for some* $\lambda \in \mathbb{F} \setminus \{0\}$*.*

In particular, the case $N = 1$ *solves the sparse rational interpolation problem in the case of coprime sparse polynomials of bounded individual degree.*

The common scalar λ in Theorem 1.5 can be avoided if we assume some normalization of the input polynomials.

Remark 1.6. We note that Theorem 1.4 and Theorem 1.5 hold over fields of characteristic larger than $\mathrm{poly}(n, d!, s^d)$ as well (this is required in the proof of Lemma 3.3). In order not to overload the claims we state them over characteristic zero alone.

1.3 Proof overview

Let us sketch the new idea. The construction in [CS26, §3] provides a generator

$$G_{(m)} = \big((G_{(m)})_1, \ldots, (G_{(m)})_n\big) : \mathbb{F}^6 \to \mathbb{F}^n,$$

depending on six *generator variables* (t, y, z, w, u, v), such that for any $f \in \mathcal{P}(n, s, d)$ the polynomial $f\big(G_{(m)}(t, y, z, w, u, v)\big)$ determines f. In fact,

$$G_{(m)}(t, y, z, w, u, v) \;=\; G^{\mathrm{KS}}_{(m)}(t, y, z, w) \;+\; G^{\mathrm{SV}}(u, v).$$

Here G^{SV} is the generator of Shpilka-Volkovich [SV14], used to revive a selected original variable, while G^{KS} is the Klivans-Spielman generator [KS01], used for sparse interpolation after the revival step.

To recover the denominator $b \in \mathcal{P}(n, s, d)$ from blackbox access to the a_j/b, the strategy of [CS26] is to compute, for each irreducible factor ϕ in a precomputed list $\mathcal{F}$, the multiplicity of ϕ as a factor of b. This requires knowing $\mathcal{F}$ in advance.

Our approach instead reconstructs $b\Big(G^{\mathrm{KS}}_{(m)}\Big)$ *directly*, via its logarithmic derivatives:

$$\mathcal{L}_\xi \;:=\; \frac{\partial_\xi\Big(b\Big(G^{\mathrm{KS}}_{(m)}\Big)\Big)}{b\Big(G^{\mathrm{KS}}_{(m)}\Big)}, \qquad \xi \in \{t, y, z, w\}.$$

By the chain rule,

$$\mathcal{L}_\xi = \sum_{i=1}^{n} \frac{(\partial_{x_i} b)(G^{\mathrm{KS}}_{(m)})}{b(G^{\mathrm{KS}}_{(m)})} \cdot \partial_\xi (G^{\mathrm{KS}}_{(m)})_i,$$

so it suffices to recover the "original-variable" logarithmic derivatives $(\partial_{x_i} b)(G^{\mathrm{KS}}_{(m)})/b(G^{\mathrm{KS}}_{(m)})$ for each i. To do so, we specialize v, obtaining a new generator $G_{(m,i)}$, in which u plays the role of the revived variable x_i. By computing the lcm (least common multiple) of the denominators of the rational functions $a_j(G_{(m,i)})/b(G_{(m,i)})$, viewed as rational functions in u over $\mathbb{F}(t, y, z, w)$, we get access to the product of all irreducible factors of b that depend on x_i, composed with $G_{(m,i)}$, up to a multiplication by some element of the base field $\mathbb{F}(t, y, z, w)$. Taking the logarithmic derivative with respect to u eliminates these $\mathbb{F}(t, y, z, w)$-scalar factors and yields $\partial_u b(G_{(m,i)})/b(G_{(m,i)})$, which, after substituting $u = (G^{\mathrm{KS}}(m))_i$, gives by the chain rule the desired $(\partial x_i b)(G^{\mathrm{KS}}(m))/b(G^{\mathrm{KS}}_{(m)})$. The coprimality results of [CS26] ensure this extraction works: the $\phi_\nu(G_{(m,i)})$'s for distinct irreducible factors ϕ_ν of b remain pairwise coprime as polynomials in the revived variable, see Lemma 2.17.

Once the $\mathcal{L}_\xi$'s are known, we reconstruct $b(G^{\mathrm{KS}}_{(m)})$ by solving the linear system

$$\partial_\xi P - \mathcal{L}_\xi \cdot P = 0 \qquad \text{for all } \xi \in \{t, y, z, w\}$$

in the space of polynomials in (t, y, z, w) of degree at most $\mathrm{poly}(m)$. In characteristic zero (or larger than the total degrees involved in the equation, see Remark 3.5) this system has a one-dimensional solution space spanned by $b(G^{\mathrm{KS}}_{(m)})$, since the common kernel of the four partial derivations on $\mathbb{F}(t, y, z, w)$ is exactly $\mathbb{F}$.

With the bounded-individual-degree sparse rational interpolation result at hand, the factorization follows the template of the meta-algorithm of [CS26], which we sketch next.

Given the input product $f = \prod_{j=1}^{\ell} h_j$ of irreducible polynomials in $\mathcal{P}(n+1, s, d)$, we pick a distinguished variable x_0 and view f as a polynomial in x_0 with coefficients in $\mathbb{F}[\mathbf{x}]$, $f = \sum_j f_j \cdot x_0^j$.

We next define $\tilde{f}(X, \mathbf{x}) := f(X \cdot f_0, \mathbf{x})/f_0$, which is reverse-monic in X and has coefficients in $\mathbb{F}(\mathbf{x})$. By Claim 2.11, each irreducible factor $h = \sum_{j=0}^{d_h} c_j \cdot x_0^j$ of f with $x_0 \in \mathrm{var}(h)$ corresponds to an irreducible factor $\tilde{h}(X, \mathbf{x}) = \sum_{j=0}^{d_h} (c_j \cdot f_0^j/c_0) \cdot X^j$ of $\tilde{f}$, with rational coefficients in $\mathbb{F}(\mathbf{x})$.

We now compose $\tilde{f}$ with the generator $G_{(m)}$ in the $\mathbf{x}$ coordinates, obtaining $\hat{f}(X, t, y, z, w, u, v) := \tilde{f}(X, G_{(m)}(t, y, z, w, u, v))$, a polynomial in 7 variables, which we recover explicitly by dense interpolation and factor. Each irreducible factor $\tilde{h}$ of $\tilde{f}$ as above corresponds to a divisor of $\hat{f}$, whose coefficients as a polynomial in X provide us with blackbox access to the rational functions $(c_j \cdot f_0^j/c_0)(G_{(m)})$. Dividing the jth coefficient by $f_0(G_{(m)})^j$ yields blackbox access to $c_j(G_{(m)})/c_0(G_{(m)})$. Since h is irreducible and depends on x_0, Gauss's lemma gives $\gcd(c_0, c_1, \ldots, c_{d_h}) = 1$, and since $h \in \mathcal{P}(n+1, s, d)$, all c_j's are in $\mathcal{P}(n, s, d)$. We can therefore invoke Theorem 1.5 (with $b = c_0$ and $a_j = c_j$) to recover the c_j's, and hence h, up to a common scalar. Enumerating divisors of $\hat{f}$ of X-degree at most d yields a polynomially bounded list of candidate factors of f. We then use algorithms of [CS26] to prune out any reducible or non-dividing polynomial from the list and to compute the multiplicity of each irreducible factor.

1.4 Related work

We elaborate on the work most directly relevant to the present paper.

Sparsity bounds for factors. Volkovich [Vol15, Vol17] showed that factors of multilinear, respectively multiquadratic, s-sparse polynomials are again s-sparse. Bhargava, Saraf, and Volkovich [BSV20] extended this to give the first nontrivial sparsity bound for factors of (n, s, d)-sparse polynomials: $s^{O(d^2 \log n)}$. Whether the true bound is polynomial in s and n is a major open problem. Bisht and Saxena [BS25] improved this to $s^{\mathrm{poly}(d)}$ under the assumption that the polynomial is also symmetric. Bisht and Volkovich [BV25] obtained an $s^{O(d \log s)}$ bound on the sparsity of a quotient of an s-sparse polynomial of bounded individual degree d by a multilinear factor. From the work of Forbes [For15] and [BV25] it follows that the sparsity of the quotient of $f \in \mathcal{P}(n, s, d)$ by a degree r polynomial is bounded by $(ns)^{O(dr)}$.

Deterministic factorization. For a single (n, s, d)-sparse polynomial, [BSV20] gave a $\mathrm{poly}(n, s^{d^7 \log n})$-time deterministic factorization algorithm, later improved by [CS26] to $\mathrm{poly}(n, s^{d^3 \log n})$.

Very recently, Bhattacharjee, Kothary, Rai and Saraf [BKRS26] obtained a complementary list-factorization result for sparse polynomials of bounded individual degree. They show that, given an explicit n-variate s-sparse polynomial of individual degree at most d, one can deterministically output in time $\mathrm{poly}(n, d!, s^d)$ a list of s^d constant-depth circuits containing every factor not divisible by a monomial. The list may contain spurious circuits. They also consider the more general setting in which the input sparse polynomial need not have bounded individual degree. For an s-sparse polynomial of individual degree at most D, they give a deterministic quasipolynomial-time algorithm that outputs a list of size $\binom{D}{\leq d}^{\log s}$ containing all factors of individual degree at most d that are not divisible by a monomial. The list may contain spurious elements.

For products of sparse polynomials, the situation has been more limited: Volkovich's algorithms [Vol15, Vol17] handle products with multilinear or multiquadratic factors, and [CS26] gave the quasi-polynomial-time algorithm of Theorem 1.2. Theorem 1.4 of the present paper is the first polynomial-time algorithm, for constant d, for factoring products of irreducible polynomials from $\mathcal{P}(n, s, d)$ over characteristic zero (or over polynomially large characteristic), thereby resolving Question 1.1 for every constant d.

Rational interpolation. Deterministic interpolation of a rational function a/b from blackbox access, where a and b are sparse, is a long-standing open problem; the best deterministic algorithm of Grigoriev, Karpinski, and Singer [GKS94] runs in time exponential in the number of variables. Randomized polynomial-time algorithms are known [CL11, vdHL21].

[CS26, Theorems 5.1,5.2] solve the bounded-individual-degree case, where the algorithm also receives a precomputed list containing all irreducible factors of b. Theorem 1.5 settles a bounded-individual-degree special case: the numerators $a_1, \ldots, a_N$ and common denominator b all lie in $\mathcal{P}(n, s, d)$ and satisfy $\gcd(a_1, \ldots, a_N, b) = 1$. Compared with [CS26, Theorems 5.1,5.2], the present theorem removes the need for a precomputed list containing all irreducible factors of b.

1.5 Organization

Section 2 sets up notation and collects the prerequisite results from [CS26], which we will use as black boxes. Section 3.1 proves the local denominator extraction lemma. In Section 3 we give our rational reconstruction algorithm, and prove Theorem 1.5. In Section 4 we deduce Theorem 1.4 from Theorem 1.5. Section 5 concludes with a brief discussion.

2 Preliminaries

Throughout the paper we let $\mathbb{F}$ be a field of characteristic zero. When we state results from other works we may allow $\mathbb{F}$ to have positive characteristic, but our claims and proofs assume characteristic zero (though as mentioned in Remark 1.6, they hold over characteristic larger than $\mathrm{poly}(n, d!, s^d)$ as well).

We use bold letters to denote tuples. Specifically, we let $\mathbf{x} = (x_1, \ldots, x_n)$ denote an n-tuple of indeterminates. Two nonzero polynomials are *associate* if one is a scalar multiple of the other.

Whenever we speak of blackbox access to a rational function $R \in \mathbb{F}(\mathbf{x})$, we assume that on input $\boldsymbol{\alpha} \in \mathbb{F}^n$ the blackbox either returns $R(\boldsymbol{\alpha})$ or reports that R is undefined at $\boldsymbol{\alpha}$.

Definition 2.1. A polynomial $f \in \mathbb{F}[\mathbf{x}]$ is *s-sparse* if it is a sum of at most s monomials. It is of *bounded individual degree d* if $\deg_{x_i}(f) \le d$ for every i. We denote the class of n-variate, s-sparse polynomials of bounded individual degree d by $\mathcal{P}(n, s, d)$, and the class of factors of polynomials in $\mathcal{P}(n, s, d)$ by $\mathcal{P}_{\mathrm{Factors}}(n, s, d)$.

For $\phi \in \mathcal{P}_{\mathrm{Factors}}(n, s, d)$ and a polynomial $f \in \mathbb{F}[\mathbf{x}]$, we denote by $v_\phi(f)$ the multiplicity of ϕ as a factor of f (i.e., the largest $k \ge 0$ such that $\phi^k \mid f$, with $v_\phi(f) = 0$ if $\phi \nmid f$). If $f = 0$ then we set $v_\phi(f) = \infty$. We write $\mathrm{var}(f) \subseteq \{x_1, \ldots, x_n\}$ for the set of variables on which f depends nontrivially.

2.1 Basic polynomial algorithms

The following well-known facts on univariate factorization and dense polynomial interpolation will be used throughout. For univariate factorization over $\mathbb{Q}$ we use the algorithm of Lenstra, Lenstra, and Lovász [LLL82]; over finite fields we use the algorithm of von zur Gathen and Shoup [vzGS92]. For multivariate factorization we use the reduction of [Kal85b, Kal85a] to the univariate case.

Theorem 2.2 (Univariate factorization over $\mathbb{Q}$)**.** *Let $f \in \mathbb{Q}[x]$ have degree d and bit complexity B. Then there is a deterministic algorithm that outputs all irreducible factors of f whose running time is $\mathrm{poly}(d, B)$.*

We next state a theorem concerning deterministic multivariate factorization. See e.g., [Len85, Kal85b].

Theorem 2.3 (Multivariate polynomial factorization)**.** *There exists a deterministic algorithm that given an n-variate polynomial $f \in \mathbb{F}[\mathbf{x}]$ of total degree d, and bit complexity B, outputs its factorization into irreducible polynomials. Its running time is* $\mathrm{poly}(d^n) \cdot T(d, B, \mathbb{F})$*, where $T(d, B, \mathbb{F})$ is the running time of univariate factorization of degree d polynomials, with bit complexity B, over $\mathbb{F}$.*

Lemma 2.4 (Dense multivariate interpolation, folklore)**.** *There is a deterministic algorithm that, given blackbox access to an n-variate polynomial $f \in \mathbb{F}[\mathbf{x}]$ of total degree at most d, returns its monomial representation in time* $\mathrm{poly}((d+1)^n)$*.*

We shall also rely on rational interpolation.

Lemma 2.5 (Rational interpolation)**.** *There is a deterministic algorithm that, given blackbox access to an n-variate rational function $R(\mathbf{x}) \in \mathbb{F}(\mathbf{x})$, with the promise that $R(\mathbf{x}) = f(\mathbf{x})/g(\mathbf{x})$ for some $f, g \in \mathbb{F}[\mathbf{x}]$ of total degree at most d, returns the unique (up to a common nonzero scalar) coprime pair (f^*, g^*) with $R = f^*/g^*$, in time* $\mathrm{poly}((d+1)^n)$*.*

For completeness, we sketch the proof.

Proof. Let $S \subset \mathbb{F}$ be a set of size $3d+1$. Let $\phi, \psi \in \mathbb{F}[\mathbf{x}]$ be unknown polynomials of total degree at most d. We construct and solve the linear system $R(\boldsymbol{\alpha})\phi(\boldsymbol{\alpha}) = \psi(\boldsymbol{\alpha})$, where $\boldsymbol{\alpha}$ ranges over all points in S^n for which $R(\boldsymbol{\alpha})$ is defined. For every such $\boldsymbol{\alpha}$, we have $g(\boldsymbol{\alpha}) \neq 0$ and

$$f(\boldsymbol{\alpha})\phi(\boldsymbol{\alpha}) - g(\boldsymbol{\alpha})\psi(\boldsymbol{\alpha}) = 0.$$

Thus, the polynomial $g(\mathbf{x})(f(\mathbf{x})\phi(\mathbf{x}) - g(\mathbf{x})\psi(\mathbf{x}))$ evaluates to zero on the entire grid S^n. The total degree of this polynomial is at most $3d$. By the polynomial identity lemma, since $|S| > 3d$, it follows that $g(\mathbf{x})(f(\mathbf{x})\phi(\mathbf{x}) - g(\mathbf{x})\psi(\mathbf{x})) = 0$. Because $g(\mathbf{x})$ is not the zero polynomial, $f(\mathbf{x})\phi(\mathbf{x}) - g(\mathbf{x})\psi(\mathbf{x}) = 0$, which implies $f/g = \psi/\phi$. Dividing ψ and ϕ by their greatest common divisor yields the unique coprime pair (f^*, g^*) with $f^*/g^* = R$, up to a common nonzero scalar.

To bound the running time, observe that the grid S^n contains $(3d+1)^n$ points. The polynomials ϕ and ψ each have $\binom{n+d}{n} \leq (d+1)^n$ unknown coefficients. Consequently, the linear system consists of at most $(3d+1)^n$ equations in $2\binom{n+d}{n}$ variables. Since $(3d+1)^n$ is bounded by $\mathrm{poly}((d+1)^n)$, the dimensions of the system are polynomial in $(d+1)^n$. Evaluating the black box, solving the linear system, and computing the greatest common divisor of the resulting dense multivariate polynomials all take time polynomial in the size of the system, which is bounded by $\mathrm{poly}((d+1)^n)$. □

In our application, all uses of Theorem 2.3 and Lemma 2.5 will be on polynomials and rational functions in a *constant* number of variables (the generator variables), so the running times will be polynomial in the relevant parameters.

We next recall the definition of Lagrange interpolation.

Definition 2.6. Let $\alpha_1, \ldots, \alpha_r \in \mathbb{F}$ be different field elements. Then the polynomials

$$A_i(x) = \prod_{1 \leq j \leq r, j \neq i} \frac{x - \alpha_j}{\alpha_i - \alpha_j} \tag{1}$$

satisfy $\deg(A_i) = r - 1$ and $A_i(\alpha_j) = \delta_{i,j}$, where $\delta_{i,j}$ is Kronecker's delta. The A_i-s are called the Lagrange interpolating polynomials of $\alpha_1, \ldots, \alpha_r$.

We will also need the notion of a reduced form of a rational function.

Definition 2.7 (Reduced rational function)**.** A rational function $R(u) \in \mathbb{F}(u)$ is said to be in *reduced form* if it is expressed as the fraction

$$R(u) = \frac{f(u)}{g(u)},$$

such that the following conditions hold:

(1) $f(u), g(u) \in \mathbb{F}[u]$ with $g(u) \neq 0$.

(2) $f(u)$ and $g(u)$ are coprime in $\mathbb{F}[u]$ (that is, $\gcd(f, g) = 1$).

(3) $g(u)$ is a monic polynomial.

Under these conditions, the polynomials $f(u)$ and $g(u)$ are uniquely determined by $R(u)$. The polynomial $g(u)$ is referred to as the *reduced denominator* of $R(u)$.

2.2 Reverse-monic Polynomials and Normalization

We shall need the following notions and claims from [CS26].

Definition 2.8 ([CS26, Definition 2.2])**.** A polynomial $f \in \mathbb{F}[x_0, \mathbf{x}]$ is said to be reverse-monic with respect to x_0 if its free term as a polynomial in $\mathbb{F}(\mathbf{x})[x_0]$ is 1. Equivalently, f is reverse-monic with respect to x_0 if it is of the form $f = 1 + \sum_{i \geq 1} f_i x_0^i$ where $f_i \in \mathbb{F}[\mathbf{x}]$.

Claim 2.9 ([CS26, Claim 2.3])**.** *Let $f \in \mathbb{F}[x_0, \mathbf{x}]$ be a reverse-monic polynomial with respect to x_0. Then, the number of irreducible factors of f is at most* $\deg_{x_0}(f)$.

We next define a *normalization* operation that turns a polynomial into a reverse-monic one.

Definition 2.10 ([CS26, Definition 2.4])**.** Let $f(x_0, \mathbf{x}) = \sum_{i=0}^{d} f_i(\mathbf{x}) x_0^i \in \mathbb{F}[x_0, \mathbf{x}]$, such that $x_0 \nmid f$. The normalization of f with respect to x_0 is the reverse-monic polynomial

$$\tilde{f}(y, \mathbf{x}) = f(y f_0, x_1, \ldots, x_n)/f_0 = 1 + \sum_{i=1}^{d} f_i(\mathbf{x}) f_0^{i-1}(\mathbf{x}) y^i. \tag{2}$$

Claim 2.11 ([CS26, Claim 2.5])**.** *Let f as in Definition 2.10. Suppose $f = \prod_{i=0}^{N} h_i \in \mathbb{F}[x_0, \mathbf{x}]$ is the factorization of f into irreducible factors h_i, and denote each factor by $h_i = \sum_{j=0}^{d_i} c_{i,j} x_0^j$ (where $c_{i,j} \in \mathbb{F}[\mathbf{x}]$). Then, the factorization of $\tilde{f}$ is given by*

$$\tilde{f} = \prod_{i=0}^{N} \tilde{h}_i \text{ , where } \quad \tilde{h}_i = \sum_{j=0}^{d_i} \frac{c_{i,j}}{c_{i,0}} f_0^j y^j \ .$$

Moreover, every factor (not necessarily irreducible) of f of the form $\sum_{j=0}^{d} \zeta_j x_0^j$ (where $\zeta_j \in \mathbb{F}[\mathbf{x}]$) corresponds to a factor of $\tilde{f}$ of the form $\sum_{j=0}^{d} \frac{\zeta_j}{\zeta_0} f_0^j y^j$. In particular, every multiquadratic factor of f of the form $a_2 x_0^2 + a_1 x_0 + b$ corresponds to a factor of $\tilde{f}$ of the form $1 + \frac{a_1 f_0}{b} y + \frac{a_2 f_0^2}{b} y^2$. In addition, we have blackbox access to $\tilde{f}$ from blackbox access to f.

2.3 The generator of [CS26]

We follow the formulation in [CS26, §3]. For a positive integer m (which will be taken to be a polynomial in the input parameters), let q be a prime in $[m, 2m]$, and fix sets $\mathcal{A} = \{\alpha_1, \ldots, \alpha_m\}$, $\mathcal{B} = \{\beta_1, \ldots, \beta_n\}$, $\mathcal{C} = \{\gamma_1, \ldots, \gamma_n\} \subseteq \mathbb{F}$ of distinct field elements. Let $\{A_i(y)\}_{i=1}^m$ be the Lagrange interpolation polynomials for $\mathcal{A}$, and similarly $\{B_i(z)\}_{i=1}^n$ for $\mathcal{B}$ and $\{C_i(v)\}_{i=1}^n$ for $\mathcal{C}$.

Definition 2.12 (KS generator)**.** The Klivans–Spielman generator (KS-generator) is the map $G^{\mathrm{KS}}_{(m)} : \mathbb{F}^4 \to \mathbb{F}^n$ defined by

$$G^{\mathrm{KS}}_{(m)}(t, y, z, w) \;=\; \sum_{i=1}^{m} A_i(y) \cdot \big((1 + B_1(z)(w-1))\, t^{i \bmod q},\; \ldots,\; (1 + B_n(z)(w-1))\, t^{i^n \bmod q}\big).$$

Observation 2.13. *The degree of (each coordinate of) $G^{KS}_{(m)}$ can be upper bounded by*

$$\deg(G^{KS}_{(m)} \le (m-1) + (n-1) + 1 + (q-1) < 3m + n.$$

Theorem 2.14 ([KS01], see also [CS26, §3])**.** *Let $f \in \mathcal{P}(n, s, d)$ and $m = (nds)^2$. Then f can be recovered, in deterministic time* $\mathrm{poly}(m)$*, from the values of $f\big(G^{KS}_{(m)}(t, y, z, w)\big)$ as a polynomial in the four variables (t, y, z, w).*

The KS-generator does not allow one to "revive" a single original variable x_i in a controlled way. To remedy this, [CS26], following [SV14], adds a second component that uses two further variables (u, v):

$$G^{\mathrm{SV}}(u, v) \;=\; (C_1(v)u,\; \ldots,\; C_n(v)u).$$

Definition 2.15 ([CS26, Definition 3.3])**.** The generator $G_{(m)} : \mathbb{F}^6 \to \mathbb{F}^n$ is

$$G_{(m)}(t, y, z, w, u, v) \;=\; G^{\mathrm{KS}}_{(m)}(t, y, z, w) \;+\; G^{\mathrm{SV}}(u, v).$$

We shall refer to (t, y, z, w, u, v) as the *generator variables*, and write $\boldsymbol{\xi} = (t, y, z, w, u, v) \in \mathbb{F}^6$ for short. Thus $G_{(m)} = G_{(m)}(\boldsymbol{\xi})$.

For each $1 \le i \le n$, the *ith revived generator* $G_{(m,i)}$ is obtained from $G_{(m)}$ by the substitution $u \leftarrow u - G^{\mathrm{KS}}_{(m)}(t, y, z, w)_i$ and $v \leftarrow \gamma_i$, where $G^{\mathrm{KS}}_{(m)}(\cdot)_i$ denotes the ith coordinate of $G^{\mathrm{KS}}_{(m)}$. Since $C_j(\gamma_i) = \delta_{j,i}$, we get

$$\begin{aligned} G_{(m,i)}(\boldsymbol{\xi}) \;=\; \big(&G^{\mathrm{KS}}_{(m)}(t, y, z, w)_1,\; \ldots,\; G^{\mathrm{KS}}_{(m)}(t, y, z, w)_{i-1},\; u,\\ &G^{\mathrm{KS}}_{(m)}(t, y, z, w)_{i+1},\; \ldots,\; G^{\mathrm{KS}}_{(m)}(t, y, z, w)_n\big). \end{aligned} \tag{3}$$

In particular, the i-th revived generator $G_{(m,i)}$ depends on the five generator variables (t, y, z, w, u), and the variable u plays the role of the "revived" original variable x_i.

We will repeatedly use the following basic properties of $G_{(m)}$, all established in [CS26].

Lemma 2.16 ([CS26, Lemma 3.4, Corollaries 3.5, 3.6])**.** *Let $f \in \mathcal{P}_{Factors}(n, s, d)$ and let $m = (nds)^2$. Then:*

(1) $f(G_{(m)}) \neq 0$. In fact, if $f \in \mathcal{P}(n, s, d)$, then f can be recovered from $f(G_{(m)})$ in deterministic time $\mathrm{poly}(m)$*.*

(2) For every $1 \le i \le n$, $\deg_u\big(f(G_{(m,i)})\big) = \deg_{x_i}(f)$.

(3) In particular, if $x_i \notin var(f)$ then $f(G_{(m,i)})$ does not depend on u, while if $x_i \in var(f)$ then $\deg_u(f(G_{(m,i)})) \ge 1$.

2.4 Algorithmic tools from [CS26]

An important ingredient of the analysis in [CS26] is the following preservation of coprimality under composition with the revived generator. We will use it as a black box.

Lemma 2.17 ([CS26] Lemma 4.15)**.** *Let $\mathbb{F}$ be a field of characteristic zero or larger than $2d$. Let ϕ, ψ be non-associate irreducible polynomials in $\mathcal{P}_{Factors}(n,s,d)$, both dividing (n,s,d)-sparse polynomials. Fix $1 \le i \le n$ with $x_i \in var(\phi)$. Then, for some explicit $m = \mathrm{poly}(n, d!, s^d)$, the polynomials $\phi(G_{(m,i)})$ and $\psi(G_{(m,i)})$ are coprime as polynomials in u over $\mathbb{F}(t,y,z,w)$. Moreover, $\phi(G_{(m,i)})$ is square-free as a polynomial in u.*

The following is a consequence of the recovery property of $G_{(m)}$.

Lemma 2.18 ([CS26, Corollary 3.8])**.** *There is a deterministic $\mathrm{poly}(n,d,s,\ell)$-time algorithm that, given blackbox access to a product f of ℓ polynomials in $\mathcal{P}_{Factors}(n,s,d)$, computes the multiplicity of each x_i as a factor of f. If M is the maximal monomial divisor of f, the algorithm also provides blackbox access to f/M.*

We shall also rely on the following result that enables computing the multiplicity of $\phi \in \mathcal{P}_{\mathrm{Factors}}(n,s,d)$ as a factor of a product f of ℓ polynomials in $\mathcal{P}_{\mathrm{Factors}}(n,s,d)$

Lemma 2.19 ([CS26, Lemma 4.19])**.** *Let $\mathbb{F}$ be a field of characteristic zero, or of characteristic larger than $2d$. There is a deterministic $\mathrm{poly}(n, d!, s^d, \ell)$-time algorithm that, given blackbox access to a product, f, of ℓ polynomials in $\mathcal{P}_{Factors}(n,s,d)$ and blackbox access to an irreducible polynomial ϕ in $\mathcal{P}_{Factors}(n,s,d)$, both defined over $\mathbb{F}$, returns the multiplicity of ϕ as a factor of f. In fact, the algorithm only needs to know the values of f and ϕ on the image of the generator $G_{(m)}$ for some explicit $m = \mathrm{poly}(n, d!, s^d)$.*

The following divisibility testing result will also be useful to us.

Theorem 2.20 ([CS26, Theorem 1.16])**.** *Let $\mathbb{F}$ be a field of characteristic zero or larger than $2d$. There exists a deterministic $\mathrm{poly}(n, d!, s^d, \ell)$-time algorithm that, given blackbox access to two polynomials (over $\mathbb{F}$) f and g that are products of ℓ polynomials in $\mathcal{P}_{Factors}(n,s,d)$, decides if $f|g$.*

3 Bounded-individual-degree sparse rational interpolation

In this section we prove Theorem 1.5 for fields of characteristic zero (or polynomially large). We first show how to compute the logarithmic derivative of the denominator b (Section 3.1). We then set up the linear differential equation and prove that its solution is what we need (Section 3.2). Finally, we combine both results to obtain our reconstruction algorithm (Section 3.3).

3.1 Computing the logarithmic derivative

In this section we prove the following lemma, which is the technical starting point of our reconstruction. Recall that the generator variables are $\boldsymbol{\xi} = (t,y,z,w,u,v)$, and that the i-th revived generator $G_{(m,i)}$ depends only on the five generator variables (t,y,z,w,u). Throughout we assume that $\mathbb{F}$ is a field of characteristic zero. As mentioned, the result also holds when the characteristic is at least $\mathrm{poly}(n, d!, s^d)$, and we shall explain where this assumption is needed.

Lemma 3.1. *Let $a_1, \ldots, a_N, b \in \mathcal{P}(n, s, d)$, over $\mathbb{F}$, with $\gcd(a_1, \ldots, a_N, b) = 1$. Let $m = \mathrm{poly}(n, d!, s^d)$ be as in Lemma 2.17. Fix $1 \le i \le n$, and write the factorization of b into irreducible polynomials as*

$$b \;=\; \beta \cdot \prod_{\nu=1}^{r} \phi_\nu^{e_\nu},$$

where $\beta \in \mathbb{F} \setminus \{0\}$, the ϕ_ν's are pairwise non-associate irreducibles, and $e_\nu \ge 1$. For each j, let

$$R_{j,i}(u) \;:=\; \frac{a_j(G_{(m,i)})}{b(G_{(m,i)})} \;\in\; \mathbb{F}(t, y, z, w)(u),$$

and let $D_{j,i}(u) \in \mathbb{F}(t, y, z, w)[u]$ be the monic reduced denominator of $R_{j,i}(u)$ when viewed as a rational function in u over $\mathbb{F}(t, y, z, w)$ (recall Definition 2.7). Then:

(1) For every j,

$$D_{j,i}(u) \;=\; c_{j,i} \cdot \prod_{\nu \,:\, x_i \in var(\phi_\nu)} \phi_\nu(G_{(m,i)})^{\max(e_\nu - v_{\phi_\nu}(a_j),\, 0)} \tag{4}$$

for some nonzero $c_{j,i} \in \mathbb{F}(t, y, z, w)$.

(2) Setting $D_i(u) := \mathrm{lcm}_{1 \le j \le N}\big(D_{j,i}(u)\big) \in \mathbb{F}(t, y, z, w)[u]$,[1] we have

$$D_i(u) \;=\; c_i \cdot \prod_{\nu \,:\, x_i \in var(\phi_\nu)} \phi_\nu(G_{(m,i)})^{e_\nu} \tag{5}$$

for some nonzero $c_i \in \mathbb{F}(t, y, z, w)$.

(3) As a rational function in u over $\mathbb{F}(t, y, z, w)$, the rational function $\frac{\partial_u D_i(u)}{D_i(u)}$ equals $\frac{\partial_u b(G_{(m,i)})}{b(G_{(m,i)})}$.

We refer to the rational functions $D_{j,i}(u)$ as the *local denominators* of $R_{j,i}$, and to $D_i(u)$ as the *i-th local denominator*.

Proof. Fix j and write the factorization of a_j into irreducible polynomials as

$$a_j \;=\; \alpha_j \cdot \prod_{\nu=1}^{r} \phi_\nu^{v_{\phi_\nu}(a_j)} \cdot \prod_{\mu} \psi_{j,\mu}^{f_{j,\mu}},$$

where $\alpha_j \in \mathbb{F} \setminus \{0\}$, $v_{\phi_\nu}(a_j) \ge 0$, and each $\psi_{j,\mu}$ is irreducible and non-associate to every ϕ_ν. Since $a_j \in \mathcal{P}(n, s, d)$, every irreducible factor of a_j lies in $\mathcal{P}_{\mathrm{Factors}}(n, s, d)$; in particular, so does every $\psi_{j,\mu}$.

Composing with $G_{(m,i)}$ gives

$$R_{j,i}(u) \;=\; \frac{\alpha_j}{\beta} \cdot \prod_{\nu=1}^{r} \phi_\nu(G_{(m,i)})^{v_{\phi_\nu}(a_j) - e_\nu} \cdot \prod_{\mu} \psi_{j,\mu}(G_{(m,i)})^{f_{j,\mu}}. \tag{6}$$

By Lemma 2.16, for every (nonzero) irreducible $\zeta \in \mathcal{P}_{\mathrm{Factors}}(n, s, d)$ we have $\deg_u(\zeta(G_{(m,i)})) \ge 1$ if $x_i \in \mathrm{var}(\zeta)$, and $\zeta(G_{(m,i)}) \in \mathbb{F}(t, y, z, w) \setminus \{0\}$ otherwise (in the latter case $\zeta(G_{(m,i)})$ is a nonzero element of $\mathbb{F}(t, y, z, w)$, hence a unit in $\mathbb{F}(t, y, z, w)[u]$). Apply this to each ϕ_ν and each $\psi_{j,\mu}$. The factors with $x_i \notin \mathrm{var}(\zeta)$ are units, hence they factor out from the denominator in the reduced form of $R_{j,i}$.

[1] Since we defined $v_\phi(0) = \infty$, no zero polynomial appears in the computation of the lcm.

Restrict attention to ν's with $x_i \in \mathrm{var}(\phi_\nu)$. For two distinct such ν, ν', the irreducible polynomials $\phi_\nu, \phi_{\nu'}$ are non-associate (and clearly both in $\mathcal{P}_{\mathrm{Factors}}(n,s,d)$). By Lemma 2.17, $\phi_\nu(G_{(m,i)})$ and $\phi_{\nu'}(G_{(m,i)})$ are coprime as polynomials in u over $\mathbb{F}(t,y,z,w)$. Similarly, for every ν with $x_i \in \mathrm{var}(\phi_\nu)$ and every μ, the irreducible polynomial $\psi_{j,\mu}$ is non-associate to ϕ_ν, and by Lemma 2.17, $\phi_\nu(G_{(m,i)})$ and $\psi_{j,\mu}(G_{(m,i)})$ are coprime as polynomials in u. Moreover, the $\phi_\nu(G_{(m,i)})$'s for $x_i \in \mathrm{var}(\phi_\nu)$ are each square-free as polynomials in u.

Therefore, it follows that the multiplicity of (the reduced form of) $\phi_\nu(G_{(m,i)})$ in the reduced denominator of $R_{j,i}(u)$ is exactly $\max(e_\nu - v_{\phi_\nu}(a_j), 0)$ for ν with $x_i \in \mathrm{var}(\phi_\nu)$, and the factors with $x_i \notin \mathrm{var}(\phi_\nu)$ contribute only to a unit factor in $\mathbb{F}(t,y,z,w)$. The latter unit becomes the constant $c_{j,i}$ after we make the denominator monic in u. This proves (4).

Since $\gcd(a_1,\ldots,a_N,b) = 1$, no irreducible polynomial divides all of $a_1,\ldots,a_N$ and b simultaneously. In particular, for every irreducible factor ϕ_ν of b, there exists at least one $j \in \{1,\ldots,N\}$ such that $v_{\phi_\nu}(a_j) = 0$. For that j,

$$\max(e_\nu - v_{\phi_\nu}(a_j),\, 0) \;=\; e_\nu.$$

Taking the LCM of the $D_{j,i}$'s over j therefore picks up $\phi_\nu(G_{(m,i)})$ with multiplicity exactly e_ν. The LCM also picks up a $\mathbb{F}(t,y,z,w)$-scalar from the $c_{j,i}$'s, which becomes the constant c_i in (5).

Clearly, $b(G_{(m,i)}) = \beta \cdot \prod_\nu \phi_\nu(G_{(m,i)})^{e_\nu}$. Split this product into the part where $x_i \in \mathrm{var}(\phi_\nu)$ and the part where $x_i \notin \mathrm{var}(\phi_\nu)$. By Lemma 2.16, the latter part lies in $\mathbb{F}(t,y,z,w) \setminus \{0\}$. Therefore, by (5),

$$b(G_{(m,i)}) \;=\; c_i' \cdot D_i(u)$$

for some nonzero $c_i' \in \mathbb{F}(t,y,z,w)$ (specifically, $c_i' = \beta \cdot (\prod_{x_i \notin \mathrm{var}(\phi_\nu)} \phi_\nu(G_{(m,i)})^{e_\nu})/c_i$). Since c_i' does not depend on u, $\partial_u c_i' = 0$, and therefore

$$\frac{\partial_u b(G_{(m,i)})}{b(G_{(m,i)})} \;=\; \frac{c_i' \cdot \partial_u D_i(u)}{c_i' \cdot D_i(u)} \;=\; \frac{\partial_u D_i(u)}{D_i(u)},$$

as required. □

The next lemma converts Lemma 3.1 into an algorithmic statement, expressed in terms of the partial derivative of b with respect to x_i composed with the KS-generator.

Lemma 3.2. *Under the assumptions of Lemma 3.1, for m as in Lemma 3.1 and any $1 \le i \le n$, the rational function*

$$\mathcal{L}_i(t,y,z,w) \;:=\; \frac{(\partial_{x_i} b)\big(G_{(m)}^{KS}(t,y,z,w)\big)}{b\big(G_{(m)}^{KS}(t,y,z,w)\big)} \;\in\; \mathbb{F}(t,y,z,w)$$

can be computed in deterministic $\mathrm{poly}(m,N)$*-time from blackbox access to the rational functions $a_1/b,\ldots,a_N/b$.*

Proof. Fix $1 \le i \le n$. The first step is to interpolate the $R_{j,i}$'s. Each $R_{j,i}(u) = a_j(G_{(m,i)})/b(G_{(m,i)})$, viewed as a rational function in the five generator variables (t,y,z,w,u), can be written as f/g with $f,g \in \mathbb{F}[t,y,z,w,u]$ of total degree $\mathrm{poly}(m)$. Note that f and g need not be coprime, since a_j and b may share common factors ϕ_ν. By choosing $\mathrm{poly}(m)$ many evaluation points for (t,y,z,w,u), we can evaluate $R_{j,i}$ using blackbox access to a_j/b. Applying Lemma 2.5 to $R_{j,i}$ (with the bound on total degree of numerator and denominator), we recover the unique coprime pair representing $R_{j,i}$ in lowest terms. From this we read off the monic reduced denominator $D_{j,i}(u) \in \mathbb{F}(t,y,z,w)[u]$

(viewing the reduced rational function in u over $\mathbb{F}(t,y,z,w)$). The total work is $\mathrm{poly}(m)$ for each j, hence $\mathrm{poly}(m,N)$ overall.

Next, compute $D_i(u) := \mathrm{lcm}_{1\le j\le N} D_{j,i}(u)$ via standard univariate lcm computations in $\mathbb{F}(t,y,z,w)[u]$. This runs in $\mathrm{poly}(m,N)$ time.

By Lemma 3.1,

$$\frac{\partial_u D_i(u)}{D_i(u)} \;=\; \frac{\partial_u b(G_{(m,i)})}{b(G_{(m,i)})}.$$

We compute $\partial_u D_i(u)$ explicitly, and form the rational function $\partial_u D_i / D_i$ in $\mathbb{F}(t,y,z,w,u)$.

By the chain rule for the polynomial $b(G_{(m,i)}(t,y,z,w,u))$ viewed as a polynomial in the five variables (t,y,z,w,u), we have

$$\partial_u\big(b(G_{(m,i)})\big) \;=\; \sum_{k=1}^{n} (\partial_{x_k} b)(G_{(m,i)}) \cdot \partial_u (G_{(m,i)})_k.$$

By (3), only the i-th coordinate of $G_{(m,i)}$ depends on u, and that coordinate equals u itself. Hence, $\partial_u (G_{(m,i)})_k = \delta_{k,i}$, and the chain rule gives

$$\partial_u\big(b(G_{(m,i)})\big) \;=\; (\partial_{x_i} b)(G_{(m,i)}). \tag{7}$$

Now substitute $u \leftarrow G^{\mathrm{KS}}_{(m)}(t,y,z,w)_i$ into (3). By construction (Definition 2.15), the resulting vector is exactly $G^{\mathrm{KS}}_{(m)}(t,y,z,w)$. Therefore

$$\big(\partial_{x_i} b\big)(G_{(m,i)})\big|_{u=G^{\mathrm{KS}}_{(m)}(t,y,z,w)_i} \;=\; (\partial_{x_i} b)\big(G^{\mathrm{KS}}_{(m)}(t,y,z,w)\big),$$

and similarly for $b(G_{(m,i)})$. Combining with (7) and Lemma 3.1(3),

$$\frac{\partial_u D_i(u)}{D_i(u)}\bigg|_{u=G^{\mathrm{KS}}_{(m)}(t,y,z,w)_i} \;=\; \frac{(\partial_{x_i} b)\big(G^{\mathrm{KS}}_{(m)}(t,y,z,w)\big)}{b\big(G^{\mathrm{KS}}_{(m)}(t,y,z,w)\big)} \;=\; \mathcal{L}_i(t,y,z,w),$$

and the specialization can be performed in $\mathrm{poly}(m)$ arithmetic. □

3.2 Reconstruction from the logarithmic derivative

We work with the four-variable polynomial

$$B(t,y,z,w) \;:=\; b\big(G^{\mathrm{KS}}_{(m)}(t,y,z,w)\big),$$

and show how to reconstruct B from the rational functions $\mathcal{L}_1,\dots,\mathcal{L}_n$ provided by Lemma 3.2, by solving a homogeneous linear system. The recovery of b from B then follows from Theorem 2.14.

Lemma 3.3. *Let $b \in \mathcal{P}(n,s,d)$, and set $B(t,y,z,w) := b\big(G^{KS}_{(m)}(t,y,z,w)\big)$. Let Δ be any upper bound on the total degree of B as a polynomial in the four variables (t,y,z,w). Assume the rational functions*

$$\mathcal{L}_i(t,y,z,w) \;=\; \frac{(\partial_{x_i} b)\big(G^{KS}_{(m)}\big)}{b\big(G^{KS}_{(m)}\big)}, \qquad 1 \le i \le n,$$

are explicitly known as elements of $\mathbb{F}(t,y,z,w)$. Then $B(t,y,z,w)$ can be recovered, up to a nonzero scalar in $\mathbb{F}$, by solving a homogeneous linear system over $\mathbb{F}$ of size $\mathrm{poly}(n,m,\Delta)$.

Proof. Let ξ denote any one of the four generator variables t, y, z, w. By the chain rule applied to $B(t,y,z,w) = b\big(G^{\mathrm{KS}}_{(m)}(t,y,z,w)\big)$,

$$\partial_\xi B \;=\; \sum_{k=1}^{n} (\partial_{x_k} b)\big(G^{\mathrm{KS}}_{(m)}\big) \cdot \partial_\xi \big(G^{\mathrm{KS}}_{(m)}\big)_k. \tag{8}$$

By Lemma 2.16 (applied to $b \neq 0$ in $\mathcal{P}(n,s,d)$) we have $B \neq 0$. Dividing (8) by B,

$$\frac{\partial_\xi B}{B} \;=\; \sum_{k=1}^{n} \mathcal{L}_k(t,y,z,w) \cdot \partial_\xi \big(G^{\mathrm{KS}}_{(m)}\big)_k. \tag{9}$$

Clearly, $\frac{\partial_\xi B}{B}$ can be easily computed from the $\mathcal{L}_k$'s and the explicit polynomials $\partial_\xi(G^{\mathrm{KS}}_{(m)})_k$, in time $\mathrm{poly}(n,m)$. Similarly to the proof of Lemma 2.5, after clearing denominators of the explicitly known rational functions $\mathcal{L}_k$, each equation gives polynomially many linear constraints on the coefficients of P. We can therefore set up the following homogeneous linear system in an unknown polynomial $P \in \mathbb{F}[t,y,z,w]$ of total degree at most Δ:

$$P \cdot \frac{\partial_\xi B}{B} \;-\; \partial_\xi P \;=\; 0, \qquad \xi \in \{t,y,z,w\}. \tag{10}$$

Equation (10) gives a finite set of homogeneous linear constraints on the coefficients of P, and is equivalent to $P \cdot \partial_\xi B = \partial_\xi P \cdot B$ as an identity in $\mathbb{F}(t,y,z,w)$. The total number of unknowns is $\binom{\Delta+4}{4} = \mathrm{poly}(\Delta)$, and the total number of equations is at most $\mathrm{poly}(n,m,\Delta)$.

It is clear that $P = B$ satisfies (10). Suppose $P \in \mathbb{F}[t,y,z,w]$ of degree at most Δ is a nonzero solution of (10). Then for every ξ,

$$\frac{\partial_\xi P}{P} \;=\; \frac{\partial_\xi B}{B},$$

so $\partial_\xi(P/B) = (B \cdot \partial_\xi P - P \cdot \partial_\xi B)/B^2 = 0$ for every $\xi \in \{t,y,z,w\}$. Since $\mathbb{F}$ has characteristic zero, the common kernel of $\partial_t, \partial_y, \partial_z, \partial_w$ on $\mathbb{F}(t,y,z,w)$ equals $\mathbb{F}$. Therefore $P/B \in \mathbb{F} \setminus \{0\}$, and the solution space is one-dimensional.

The linear system (10) has $\mathrm{poly}(n,m,\Delta)$ unknowns and equations, both computable explicitly from the $\mathcal{L}_i$'s and $G^{\mathrm{KS}}_{(m)}$ in $\mathrm{poly}(n,m,\Delta)$ time. Solving in time $\mathrm{poly}(n,m,\Delta)$ yields a basis vector of the one-dimensional kernel, which is a polynomial $\widetilde{B} = \lambda B$ for some $\lambda \in \mathbb{F} \setminus \{0\}$. □

3.3 Putting it together

We now combine Lemma 3.2 with Lemma 3.3 to prove Theorem 1.5.

Proof of Theorem 1.5. The proof proceeds in four steps: extract logarithmic derivatives, reconstruct B up to scalar, recover b, then recover the a_j's up to the same scalar.

Set $m = \mathrm{poly}(n, d!, s^d)$ as in Lemma 2.17 and Lemma 3.1. Since each coordinate of $G^{\mathrm{KS}}_{(m)}$ is a polynomial in (t,y,z,w) of total degree $\mathrm{poly}(m)$, and b has total degree at most nd, the polynomial $B(t,y,z,w) = b\big(G^{\mathrm{KS}}_{(m)}(t,y,z,w)\big)$ has total degree

$$\Delta \;\leq\; nd \cdot \mathrm{poly}(m) \;=\; \mathrm{poly}(n, d!, s^d).$$

(1) For each $1 \leq i \leq n$, apply Lemma 3.2 to obtain $\mathcal{L}_i \in \mathbb{F}(t,y,z,w)$ explicitly. The cost is $\mathrm{poly}(m,N)$ per i, hence $\mathrm{poly}(n,m,N)$ overall.

(2) Apply Lemma 3.3 with the degree bound Δ. The output is a nonzero polynomial $\widetilde{B} \in \mathbb{F}[t,y,z,w]$ of total degree at most Δ satisfying $\widetilde{B} = \lambda B$ for some $\lambda \in \mathbb{F} \setminus \{0\}$. The running time is $\mathrm{poly}(n,m,\Delta)$.

(3) Apply Theorem 2.14 to $\widetilde{B} = \lambda B = (\lambda b)\big(G^{\mathrm{KS}}_{(m)}\big)$. This recovers the polynomial $\lambda b \in \mathbb{F}[\mathbf{x}]$ in time $\mathrm{poly}(m)$.

(4) Finally, for each $1 \le j \le N$, since we have blackbox access to a_j/b and λb is now known explicitly, we can compose both with $G^{\mathrm{KS}}_{(m)}$ and multiply, to obtain blackbox access to $\lambda a_j \left(G^{\mathrm{KS}}_{(m)}\right)$. There is a subtlety here that we need to avoid zeros of $b\left(G^{\mathrm{KS}}_{(m)}\right)$. This can be easily circumvented as in Lemma 2.5: we interpolate $(\lambda a_j)(G^{\mathrm{KS}}_{(m)})$ by solving a system of linear equations, which is defined by evaluations at nonzeros of $b\left(G^{\mathrm{KS}}_{(m)}\right)$. We then recover λa_j using Theorem 2.14.

It is important to note that the scalar λ obtained in Step 2 is the *same* for every j: it is determined entirely by the reconstruction of B. Therefore the output is $(\lambda a_1, \ldots, \lambda a_N, \lambda b)$ for a single common scalar $\lambda \in \mathbb{F} \setminus \{0\}$.

The total running time, summing over the four steps, is $\mathrm{poly}(n,m,\Delta,N) = \mathrm{poly}(n,d!,s^d,N)$, as claimed. □

Remark 3.4. Theorem 1.5 assumes blackbox access to $\frac{a_j(\mathbf{x})}{b(\mathbf{x})}$, but the proof only requires blackbox access to their images under $G_{(m)}$, namely to $\frac{a_j\left(G_{(m)}\right)}{b\left(G_{(m)}\right)}$.

Remark 3.5 (The requirement in positive characteristic). The proof of Lemma 3.3 uses $\mathrm{char}(\mathbb{F}) = 0$ to argue that the common kernel of $\partial_t, \partial_y, \partial_z, \partial_w$ on $\mathbb{F}(t,y,z,w)$ equals $\mathbb{F}$. This is true only in characteristic zero. In characteristic $p > 0$, this kernel equals $\mathbb{F}(t^p, y^p, z^p, w^p)$. The linear system (10) can then have a solution space of dimension greater than one, and the proof of Lemma 3.3 fails. Note that this is a real obstruction even when $p \gg d$: although our $b \in \mathcal{P}(n,s,d)$ has individual degree at most $d < p$, the polynomial $B = b(G^{\mathrm{KS}}_{(m)}) \in \mathbb{F}[t,y,z,w]$ has total degree $\Delta = \mathrm{poly}(n,d!,s^d) \gg p$ in general, so $B \cdot H^p \in \mathbb{F}[t,y,z,w]$ for various H can satisfy the system (10) together with B.

By the same argument, our algorithm applies without change when the characteristic is sufficiently large: concretely, whenever $\mathrm{char}(\mathbb{F}) > \Delta = \mathrm{poly}(n,d!,s^d)$, the only $H \in \mathbb{F}[t,y,z,w]$ with $\deg(B \cdot H^p) \le \Delta$ is $H \in \mathbb{F}$, so the kernel of (10) remains one-dimensional.

4 Factoring products of sparse irreducibles

In this section we deduce Theorem 1.4 from Theorem 1.5. The proof follows the meta-algorithm of [CS26, §6], with the key ingredient (the rational interpolation theorem) replaced by our version. We recall the statement of Theorem 1.4.

Theorem 1.4 (Factoring products of sparse irreducibles). *Let $\mathbb{F}$ be a field of characteristic zero. There is a deterministic $\mathrm{poly}\big(n,(s\ell d)^d\big)$-time algorithm which, given blackbox access to a product $f = \prod_{i=1}^{\ell} h_i$, where $h_1, \ldots, h_\ell \in \mathcal{P}(n,s,d)$ are (not necessarily distinct) irreducible polynomials, returns (scalar multiples of) the h_r's together with their multiplicities.*

Proof. For convenience, we assume $f \in \mathbb{F}[x_0, \mathbf{x}]$ is a product of ℓ irreducible polynomials in $\mathcal{P}(n+1,s,d)$, with $\mathbf{x} = (x_1, \ldots, x_n)$.

We first compute the multiplicity of each x_i ($0 \le i \le n$) as a factor of f, and replace f by $f/$(maximal monomial divisor), using Lemma 2.18. From now on f is not divisible by any x_i. We assume, without loss of generality, that f depends on x_0 after this step. Write $f = \sum_{j=0}^{\ell d} f_j \cdot x_0^j$, where $f_j \in \mathbb{F}[\mathbf{x}]$ and $f_0 \neq 0$.

Consider $\tilde{f}(X, \mathbf{x}) := f(X \cdot f_0(\mathbf{x}), \mathbf{x})/f_0(\mathbf{x})$, which is reverse-monic in X (meaning its constant term with respect to X is exactly 1). By Claim 2.11, $\tilde{f} \in \mathbb{F}[X, \mathbf{x}]$ and we have blackbox access to it. Let

$$\hat{f}(X, t, y, z, w, u, v) \;:=\; \tilde{f}\big(X,\, G_{(m)}(t, y, z, w, u, v)\big),$$

where $G_{(m)}$ is the generator of Definition 2.15 with $m = \mathrm{poly}(n, d!, s^d)$ as in Theorem 1.5.

Using Lemma 2.4 and Theorem 2.3, obtain the factorization of $\hat{f}$ into irreducibles, as a polynomial in 7 variables of total degree $\mathrm{poly}(m, \ell d)$. The total running time is $\mathrm{poly}(m, \ell d)$, since $\hat{f}$ has $O(1)$ variables.

Since $\tilde{f}$ is reverse-monic in X, so is $\hat{f}$. By Claim 2.9, $\hat{f}$ has at most ℓd nonconstant irreducible factors, each of positive X-degree. Hence the number of (not necessarily irreducible) divisors of $\hat{f}$ of X-degree at most d is at most $\binom{\ell d + d}{d} \le (e(\ell+1)d)^d$.

For each reverse-monic divisor $\hat{h}$ of $\hat{f}$ of X-degree at most d, we attempt to recover an irreducible factor $h = \sum_{j=0}^{d_h} c_j(\mathbf{x}) \cdot x_0^j \in \mathcal{P}(n+1, s, d)$ of f corresponding to $\hat{h}$, as follows.

By Claim 2.11, every irreducible factor $h = \sum_{j=0}^{d_h} c_j(\mathbf{x}) x_0^j$ of f with $x_0 \in \mathrm{var}(h)$ gives rise to an irreducible factor $\tilde{h}$ of $\tilde{f}$:

$$\tilde{h}(X, \mathbf{x}) \;=\; \sum_{j=0}^{d_h} \frac{c_j(\mathbf{x}) \cdot f_0(\mathbf{x})^j}{c_0(\mathbf{x})} \cdot X^j \;\in\; \mathbb{F}(\mathbf{x})[X].$$

Composing with $G_{(m)}$, let $\hat{h}(X, t, y, z, w, u, v) := \tilde{h}(X, G_{(m)}(t, y, z, w, u, v))$. If $\hat{h}$ is correctly identified as the candidate factor, then for $1 \le j \le d_h$ the coefficient of X^j in $\hat{h}$ equals $e_j = (c_j(G_{(m)}) \cdot f_0(G_{(m)})^j)/c_0(G_{(m)})$.

Dividing by $f_0(G_{(m)})^j$ (which is nonzero by Lemma 2.16),

$$Q_j \;:=\; \frac{e_j}{f_0(G_{(m)})^j} \;=\; \frac{c_j(G_{(m)})}{c_0(G_{(m)})}, \qquad 1 \le j \le d_h.$$

We have blackbox access to Q_j on the image of $G_{(m)}$. Moreover, since h is irreducible and depends on x_0, Gauss's lemma gives $\gcd(c_0, c_1, \ldots, c_{d_h}) = 1$. Since $h \in \mathcal{P}(n+1, s, d)$, all c_j's are in $\mathcal{P}(n, s, d)$.

The conditions of Theorem 1.5 are thus satisfied with $N = d_h \le d$, numerators $a_j = c_j$ for $1 \le j \le d_h$, denominator $b = c_0$, and rational functions $Q_j = c_j(G_{(m)})/c_0(G_{(m)})$. The algorithm of Theorem 1.5 returns $\lambda c_0, \lambda c_1, \ldots, \lambda c_{d_h}$ for some $\lambda \in \mathbb{F} \setminus \{0\}$, in time $\mathrm{poly}(n, d!, s^d)$. Form the candidate $h^* := \sum_{j=0}^{d_h} (\lambda c_j) x_0^j = \lambda \cdot h \in \mathcal{P}(n+1, s, d)$, and add it to a candidate list $\mathcal{F}(f)$, provided h^* is $(n+1, s, d)$-sparse, and no polynomial that is an associate of h^* is already in the list.

Repeat the procedure with each of the $n+1$ variables $x_0, x_1, \ldots, x_n$ playing the role of distinguished variable. Since every non-monomial irreducible factor of f depends on at least one variable, each such factor appears in $\mathcal{F}(f)$ for the corresponding choice of distinguished variable.

Pruning the list. The list $\mathcal{F}(f)$ contains at most $(n+1) \cdot (e(\ell+1)d)^d$ candidates, each $(n+1, s, d)$-sparse. We discard from $\mathcal{F}(f)$ every candidate that does not divide f, using the divisibility tester of Theorem 2.20 (running in $\mathrm{poly}(n, d!, s^d, \ell)$ per test). Among the survivors, a candidate ϕ is irreducible iff no other candidate $\psi \in \mathcal{F}(f)$ properly divides ϕ; we identify the irreducibles via $|\mathcal{F}(f)|^2$ pairwise divisibility tests.

Computing multiplicities. For each irreducible factor ϕ identified above, compute its multiplicity as a factor of f using Lemma 2.19, in $\mathrm{poly}(n, d!, s^d, \ell)$ time per factor.

Running-time analysis. As shown above, the candidate list satisfies

$$|\mathcal{F}(f)| \le (n+1) \cdot (e(\ell+1)d)^d.$$

The construction of the candidate list, including the dense interpolation and factorization of the constant-variate polynomial $\widehat{f}$, takes time polynomial in $n, d!, s^d, \ell$ and in $|\mathcal{F}(f)|$. For each candidate, the invocation of Theorem 1.5 takes time $\mathrm{poly}(n, d!, s^d)$. The pruning step uses at most $|\mathcal{F}(f)|^2$ divisibility tests, and the multiplicity computation uses one call to Lemma 2.19 for each surviving irreducible candidate. Thus the total running time is

$$|\mathcal{F}(f)|^2 \cdot \mathrm{poly}(n, d!, s^d, \ell) = \mathrm{poly}\Big(n, d!, s^d, (\ell d)^d\Big) = \mathrm{poly}\Big(n, (s\ell d)^d\Big),$$

as claimed. □

Remark 4.1. As noted in Remark 3.5, our result can be extended to polynomially large characteristic, namely $\mathrm{char}(\mathbb{F}) = 0$ or $\mathrm{char}(\mathbb{F}) \ge \mathrm{poly}(n, d!, s^d)$. We do not know how to extend it to smaller positive characteristic.

5 Conclusion and open problems

We have shown that, for every constant d, products of ℓ irreducible (n, s, d)-sparse polynomials over any field of characteristic zero (or polynomially large characteristic) can be factored deterministically in polynomial time, resolving Question 1.1 in the constant-individual-degree regime over such fields.

Our proof applies only in characteristic zero or sufficiently large positive characteristic, but it is natural to ask whether the argument extends to fields of arbitrary characteristic. There are two distinct obstacles. The first concerns the step of recovering $b(G_{(m,i)})$ up to a (t, y, z, w)-content from the local denominators. Here the coprimality lemma of [CS26] (Lemma 2.17) requires $\mathrm{char}(\mathbb{F}) = 0$ or $\mathrm{char}(\mathbb{F}) > 2d$. This obstacle can be circumvented by working with primitive divisors as in [CS26].

The second, more serious obstacle is the differential reconstruction step. If $B = b\Big(G^{\mathrm{KS}}_{(m)}\Big)$ has total degree $\Delta = \mathrm{poly}(n, d!, s^d)$, then in small characteristic one may have $(p \le \Delta)$. In characteristic $(p > 0)$, the common kernel of the derivations $\{\partial t, \partial y, \partial z, \partial w\}$ on $\mathbb{F}(t, y, z, w)$ is $\mathbb{F}(t^p, y^p, z^p, w^p)$, rather than $\mathbb{F}$. Consequently, the linear system for P may have a kernel of dimension greater than one, so the characteristic-zero argument no longer identifies B up to a scalar. Extending our results to small positive characteristic therefore seems to require a replacement for the logarithmic-derivative reconstruction step.

This leaves two natural directions for future work. Can the logarithmic-derivative reconstruction step be replaced by a method that applies in small characteristic? More generally, can one obtain a comparable rational interpolation theorem without assuming bounded individual degree?

Bibliography

[BKR+25] Somnath Bhattacharjee, Mrinal Kumar, Shanthanu S Rai, Varun Ramanathan, Ramprasad Saptharishi, and Shubhangi Saraf. Closure under factorization from a result of Furstenberg. *arXiv preprint arXiv:2506.23214*, 2025. 1

[BKRS26] Somnath Bhattacharjee, Rishabh Kothary, Shanthanu S. Rai, and Shubhangi Saraf. Deterministic algorithms for low individual degree factors of sparse polynomials, 2026. 4

[BS25] Pranav Bisht and Nitin Saxena. Derandomization via symmetric polytopes: Poly-time factorization of certain sparse polynomials. *ACM Transactions on Computation Theory*, 17(2):1–20, 2025. 4

[BSV20] Vishwas Bhargava, Shubhangi Saraf, and Ilya Volkovich. Deterministic factorization of sparse polynomials with bounded individual degree. *Journal of the ACM (JACM)*, 67(2):1–28, 2020. 1, 4

[BT88] Michael Ben-Or and Prasoon Tiwari. A deterministic algorithm for sparse multivariate polynomial interpolation (extended abstract). In Janos Simon, editor, *Proceedings of the 20th Annual ACM Symposium on Theory of Computing, May 2-4, 1988, Chicago, Illinois, USA*, pages 301–309. ACM, 1988. 1

[BV25] Pranav Bisht and Ilya Volkovich. On solving sparse polynomial factorization related problems. *Computational Complexity*, 34(1):7, 2025. 1, 4

[CL11] Annie Cuyt and Wen-shin Lee. Sparse interpolation of multivariate rational functions. *Theoretical Computer Science*, 412(16):1445–1456, 2011. 5

[CS26] Aminadav Chuyoon and Amir Shpilka. On factorization of sparse polynomials of bounded individual degree. *CoRR*, abs/2603.07589, 2026. 2, 1, 3, 4, 5, 7, 8, 9, 14, 16

[DST24] Pranjal Dutta, Amit Sinhababu, and Thomas Thierauf. Derandomizing multivariate polynomial factoring for low degree factors. In Amit Kumar and Noga Ron-Zewi, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2024, London School of Economics, London, UK, August 28-30, 2024*, volume 317 of *LIPIcs*, pages 75:1–75:20. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024. 1

[For15] Michael A Forbes. Deterministic divisibility testing via shifted partial derivatives. In *2015 IEEE 56th Annual Symposium on Foundations of Computer Science*, pages 451–465. IEEE, 2015. 4

[GKS90] Dima Grigoriev, Marek Karpinski, and Michael F. Singer. Fast parallel algorithms for sparse multivariate polynomial interpolation over finite fields. *SIAM J. Comput.*, 19(6):1059–1063, 1990. 1

[GKS94] Dima Grigoriev, Marek Karpinski, and Michael F Singer. Computational complexity of sparse rational interpolation. *SIAM Journal on Computing*, 23(1):1–11, 1994. 1, 5

[Kal85a] Erich Kaltofen. Computing with polynomials given by straight-line programs I: greatest common divisors. In *Proceedings of the seventeenth annual ACM symposium on Theory of computing*, pages 131–142, 1985. 1, 5

[Kal85b] Erich Kaltofen. Polynomial-time reductions from multivariate to bi- and univariate integral polynomial factorization. *SIAM J. Comput.*, 14(2):469–489, 1985. 5

[KRS24] Mrinal Kumar, Varun Ramanathan, and Ramprasad Saptharishi. Deterministic algorithms for low degree factors of constant depth circuits. In *Proceedings of the 2024 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 3901–3918. SIAM, 2024. 1

[KS01] Adam R. Klivans and Daniel Spielman. Randomness efficient identity testing of multivariate polynomials. STOC '01, page 216–223, New York, NY, USA, 2001. Association for Computing Machinery. 1, 3, 8

[Len85] Arjen Klaas Lenstra. Factoring multivariate polynomials over finite fields. *J. Comput. System Sci.*, 30(2):235–248, 1985. 5

[LLL82] Arjen Klaas Lenstra, Hendrik Willem Jr. Lenstra, and László Lovász. Factoring polynomials with rational coefficients. *Mathematische Annalen*, 261:515–534, 1982. 5

[SV14] Amir Shpilka and Ilya Volkovich. On reconstruction and testing of read-once formulas. *Theory Comput.*, 10:465–514, 2014. 3, 8

[vdHL21] Joris van der Hoeven and Grégoire Lecerf. On sparse interpolation of rational functions and gcds. *ACM Commun. Comput. Algebra*, 55(1):1–12, 2021. 5

[Vol15] Ilya Volkovich. Deterministically factoring sparse polynomials into multilinear factors and sums of univariate polynomials. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2015)*, pages 943–958. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2015. 1, 4

[Vol17] Ilya Volkovich. On some computations on sparse polynomials. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2017)*, pages 48:1–48:21. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2017. 1, 4

[vzGK85] Joachim von zur Gathen and Erich Kaltofen. Factoring sparse multivariate polynomials. *Journal of Computer and System Sciences*, 31(2):265–287, 1985. 1

[vzGS92] Joachim von zur Gathen and Victor Shoup. Computing Frobenius maps and factoring polynomials. *Comput. Complexity*, 2(3):187–224, 1992. 5

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il