On probabilistic analysis of randomization in hybrid symbolic-numeric algorithms*

Erich Kaltofen, Zhengfeng Yang
Department of Mathematics
North Carolina State University
Raleigh, North Carolina 27695-8205, USA
kaltofen,zyang4@math.ncsu.edu
http://www.kaltofen.us

Lihong Zhi
Key Laboratory of Mathematics Mechanization
Academy of Mathematics and Systems Science
Beijing 100080, China
lzhi@mmrc.iss.ac.cn
http://www.mmrc.iss.ac.cn/~lzhi/

ABSTRACT

Algebraic randomization techniques can be applied to hybrid symbolic-numeric algorithms. Here we consider the problem of interpolating a sparse rational function from noisy values. We develop a new hybrid algorithm based on Zippel's original sparse polynomial interpolation technique. We show experimentally that our algorithm can handle sparse polynomials with large degrees. We also give a (partial) mathematical justification why the Zippel's algebraic randomization technique can be used with our approximate data: the randomly generated non-zero values are expected to be bounded away from zero. We show that the random Fourier-like matrices arising in our algorithm, have the desired rank property in the exact case, and appear usable numerically.

Furthermore, we show that Sylvester matrices of polynomials with nonidentically distributed random coefficients have large condition numbers. That phenomenon has precluded several algebraic randomization techniques from use in the approximate hybrid setting.

Categories and Subject Descriptors: I.2.1 [Computing Methodologies]: Symbolic and Algebraic Manipulation —Algorithms; G.1.2 [Mathematics of Computing]: Numerical Analysis—Approximation

General Terms: algorithms, theory, experimentation

Keywords: multivariate rational function, interpolation, sparse polynomial, random matrix, structured matrix, condition number, probabilistic analysis, symbolic/numeric hybrid method

*This research was supported in part by the National Science Foundation of the USA under Grants CCF-0514585 (Kaltofen and Yang) and OISE-0456285 (Kaltofen, Yang and Zhi), and by NKBRPC (2004CB318000) and the Chinese National Natural Science Foundation under Grant 10401035 (Zhi).

SNC'07, July 25–27, 2007, London, Ontario, Canada.

1. RANDOMIZATION IN ALGEBRAIC AND HYBRID COMPUTATION

Since the discovery of the Zippel-Schwartz lemma [6, 38, 43] in 1979, randomization techniques based on introducing elements that have been randomly sampled from a finite subset of the domain of scalars have become ubiquitous. Zippel's original application was to sparse multivariate polynomial interpolation [44, 45], Schwartz's to establishing polynomial identities, and DeMillo's and Lipton's to proving programs correct by executing at random inputs.

A classical application is to compute the GCD of many polynomials by computing the GCD of random linear combinations [21, Theorem 6.2 and Note added in proof]. Important applications are the effective Hilbert irreducibility theorems [13, 20, 25] for polynomial factors and preconditioning of dense, sparse and black box matrices [2, 27, 28, 42]. Randomization is needed in Gao's polynomial factorization algorithm [11] and in solvers for linear systems whose coefficient matrices have small displacement rank [24, Appendix]. There are many more examples for exact symbolic algorithms.

Hybrid symbolic-numeric algorithms permit errors in the input scalars due to floating point round-off or through physical measurement. In order to obtain a non-trivial solution, the algorithms minimally deform those input variables to obtain the desired result, say

1. a solution to an inconsistent system of linear and polynomial equations (total least squares: TLS [16], structured TLS: STLS [32, 34], symbolic-numeric elimination [15, 17, 33, 35]),
2. the GCD of relatively prime polynomials (approximate GCD [4, 8, 30, 36]),
3. a non-trivial factorization of irreducible multivariate polynomials with real or complex coefficients (approximate factorization [12, 26, 37]),
4. a univariate rational function interpolant [19] and a sparse multivariate interpolant (sparse numeric interpolation of polynomials: SNIP [14], sparse numeric interpolation of rational functions: SNIPR [29]).

When employing algorithms from the exact symbolic setting for numerical inputs, randomization again can be advantageous. In the exact setting, the Zippel-Schwartz lemma

allows the elimination of singular cases. In the hybrid setting, the same randomization can avoid ill-conditioned subproblems [12, 14, 26], unstable divisions by elements that are near zero, or areas of divergence or local minima in iterative refinement [30].

In Section 2 we consider the approximate GCD problem when using random projections of the coefficients. In particular, we investigate the condition number of the Sylvester matrix of random polynomials. Large random dense matrices with random entries from a standard Gaussian distribution have a small condition number [3], and random perturbations of the entries of well-conditioned dense matrices remain well-conditioned [39]. Because of the root distribution of random polynomials, the Sylvester matrices of two random polynomials with coefficients from a Gaussian distribution have a similar expected values of their condition numbers. We exploit such well-conditionedness in our new rational function interpolation algorithm. When the coefficients are binomially distributed, however, the roots concentrate on the real axis and the condition numbers of the corresponding Sylvester matrices become very large. The latter phenomenon caused our earlier hybrid SNIPR algorithm in [29, Section 6] to fail on large degree inputs.

In Section 4 we present our hybrid ZNIPR algorithm for numerically interpolating rational functions from noisy values. We adapt Zippel's original variable-by-variable sparse polynomial interpolation algorithm in two ways: first, we show how to deploy the algorithm for sparse rational functions and give a complete probabilistic analysis (Section 4.1) for exact arithmetic. We then use the algorithm in the setting where the rational function values are noisy. Our experiments show that the method can handle large degree polynomials, unlike our earlier SNIPR method. In Section 3 we show that Zippel's assumptions have a justification in the numerical hybrid setting. As said above, the needed relative primeness assumptions can also be understood. Finally, it may be possible to also give estimates on the condition number of the arising random Fourier-like matrices from recent work in signal processing.

2. APPROXIMATE GCD OF RANDOM UNIVARIATE POLYNOMIALS

Let $\{a_j(\omega)\}_{j=0}^{\infty}$ denote a sequence of random variables with respect to the distribution ω. Let $p_d(x) = \sum_{j=0}^{d} a_j(\omega)x^j$ denote the random univariate polynomial of degree d defined by the sequence $\{a_j(\omega)\}$. Let α, β and δ be three arbitrary numbers such that $0 \le \alpha < \beta \le 2\pi$, and $0 < \delta \le 1$. Consider the following subsets of the complex numbers:

$$B = \{z \in \mathbb{C} \colon \alpha < \arg z < \beta\}$$
$$C = \{z \in \mathbb{C} \colon 1 - \delta \le |z| \le 1 + \delta\}.$$

The following result is stated and proved in [40], see also [1, Theorem 8.1]:

Theorem 2.1 *Let the coefficients $a_j(\omega)$ of a random algebraic polynomial $p_d(x)$ be independently and identically distributed complex-valued random variables, let the expected value*

$$E(\max\{0, \log|a_j|\}) < \infty \text{ for all } j = 0, 1, \ldots, d,$$

and let $N_d(B, \omega)$ and $N_d(C, \omega)$ denote the expected number of zeros of the random polynomials that are contained in the set B and C, respectively. Then

$$\lim_{d\to\infty} \frac{N_d(B,\omega)}{d} = \frac{\beta - \alpha}{2\pi} \text{ and } \lim_{d\to\infty} \frac{N_d(C,\omega)}{d} = 1.$$

Theorem 2.1 tells us, as the degree of the polynomial increases, the zeros tend to concentrate on the circumference of the unit circle and appear to be uniformly distributed. Moreover, it is also shown in [1, Theorem 8.5], as the number of sample polynomials increase, the sample zeros cluster about the averaged zeros. This implies that the probability for random polynomials having common roots increases along with the growth of degrees of polynomials. So high degree random polynomials always have an approximate GCD within fixed precision deformation [5]. However, we also notice that for random polynomials with independent and normally distributed coefficients, the condition number of the Sylvester matrix generated by the polynomials increases almost linearly in the degrees of polynomials, as is the case for arbitrary dense matrices [3, Theorem 6.1]:

Theorem 2.2 *For an $m \times n$ complex random matrix $G_{m\times n}$ whose elements are independent and identically distributed standard normal random variables, then the expected logarithm of the 2-norm condition number satisfies:*

$$E(\log \kappa_2(G_{m\times n})) < \log \frac{n}{n-m+1} + 2.258,$$

for any $n \ge m \ge 2$.

We do not know if the structured condition number for Sylvester matrices of random polynomials with normally distributed coefficients, which is smaller than the condition number, has linear growth.

For the univariate polynomials coming from the ZNIPR algorithm of Section 4, the coefficients of these polynomials distributed almost independently and normally, the degrees of polynomials used for Table 1 and 2 are less than 100, so by the arguments above, the roots of these univariate polynomials are well separated and the condition numbers of the Sylvester matrices are relatively small (in the 100's).

If the random polynomials have nonidentical coefficients, then the zeros of these random polynomials are distributed differently from the zeros of random polynomials with Gaussian coefficients. For example, as reported in [9, 10], the random polynomial of the form

$$p_d(x) = \sum_{j=0}^{d} a_j(\omega)\sqrt{\tbinom{d}{j}}\, x^j \tag{1}$$

with Gaussian coefficients $\{a_j(\omega)\}_{j=0}^{d}$ has $\sqrt{d}$ real zeros on average, while polynomials with identically distributed coefficients have fewer than $(2\log d + 14)/\pi$ real roots for $d \to \infty$ [18]. See Figures 1 and 2 for the distribution of the zeros of the random polynomials of degrees 20 with coefficients distributed identically and binomially respectively.

Moreover, we observe that the condition numbers of Sylvester matrices generated by the random polynomials with binomial coefficients increase exponentially in the degrees of the polynomials. That explains why the condition numbers of the Sylvester matrices come from the SNIPR algorithm are huge even for polynomials with moderate degrees. The random univariate polynomials obtained by evaluating $y \leftarrow w_3x - w_3w_1 + w_2$ where w_i are complex points on the unit

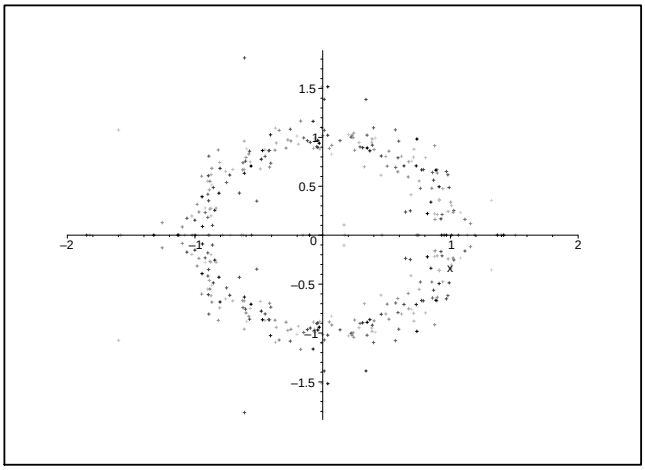

Figure 1: Roots for identically distr. coeff.'s

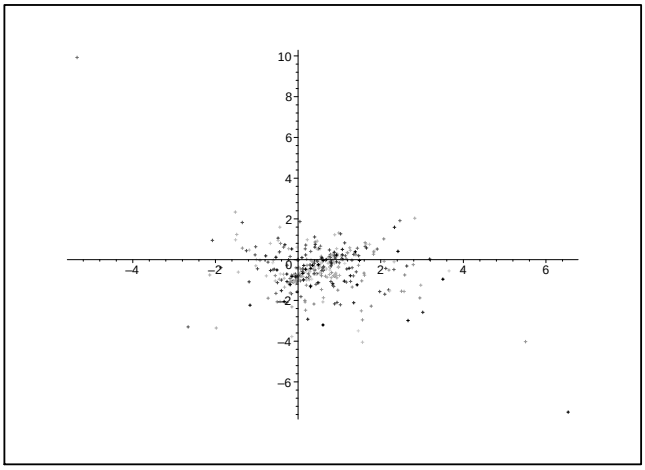

Figure 2: Roots for binomially distr. coeff.'s

circle, have coefficients distributed binomially. A mathematical analysis of the condition number and structured condition number of Sylvester matrices of polynomials with non-identically distributed coefficients, say as in (1), appears open.

3. A NUMERIC ZIPPEL-SCHWARTZ LEMMA

In Section 4 we will employ the randomization as in Zippel's original application, namely to determine if coefficient polynomial is identical zero by computing its value at a random point. We need to show that for certain random points, polynomial values are actually bounded away from zero.

Lemma 3.1 *Let* $0 \neq h(y_1,\ldots,y_r) \in \mathbb{Z}[\boldsymbol{i}][y_1,\ldots,y_r]$ *and for all* $1 \leq i \leq r$ *let* $p_i = \exp(\frac{2\pi \boldsymbol{i}}{b_i}) \in \mathbb{C}$, *with the* $b_i \in \mathbb{Z}_{\geq 3}$ *distinct prime numbers. Suppose* $h(p_1,\ldots,p_r) \neq 0$. *Then for random integers* s_i *with* $1 \leq s_i < b_i$ *the expected value*

$$E\{|h(p_1^{s_1},\ldots,p_r^{s_r})|\} \geq 1.$$

Proof. Let $B = b_1 \cdots b_r$ and let $\zeta_B = \exp(\frac{2\pi \boldsymbol{i}}{B})$. The cyclotomic polynomial $\Phi_B(z)$ of order B is irreducible over $\mathbb{Q}(\boldsymbol{i})$. Therefore $h(p_1^{s_1},\ldots,p_r^{s_r})$ is an automorphic image of $h(p_1,\ldots,p_r)$ in $\mathbb{Q}(\zeta_B,\boldsymbol{i})$ over $\mathbb{Q}(\boldsymbol{i})$ and thus non-zero. We have for the norm

$$0 \neq \prod_{1\leq s_1<b_1} \cdots \prod_{1\leq s_r<b_r} h(p_1^{s_1},\ldots,p_r^{s_r}) \in \mathbb{Z}[\boldsymbol{i}].$$

Therefore for $N = (b_1-1)\cdots(b_r-1)$ we have by the arithmetic-geometric mean inequality

$$\frac{1}{N}\sum |h(p_1^{s_1},\ldots,p_r^{s_r})| \geq \sqrt[N]{\prod |h(p_1^{s_1},\ldots,p_r^{s_r})|} \geq 1. \ \square$$

By the Schwartz-Zippel lemma we can achieve the premise $h(p_1,\ldots,p_r) \neq 0$ with high probability. Since all $|h(p_1^{s_1},\ldots,p_r^{s_r})| \leq \|h\|_1$ a reasonable number of randomly selected $h(p_1^{s_1},\ldots,p_r^{s_r})$ can be heuristically expected to be bounded from 0.

4. ZIPPEL NUMERICAL INTERPOLATION OF RATIONAL FUNCTIONS (ZNIPR)

4.1 Probabilistic analysis of exact algorithm

Consider the rational function $f/g \in F(x_1,\ldots,x_n)$, where the numerator and denominator are represented as

$$f = \sum_{j=1}^{t_f} \psi_j \mathbf{x}^{\mathbf{d}_j},\ g = \sum_{k=1}^{t_g} \chi_k \mathbf{x}^{\mathbf{e}_k}, \quad \psi_j,\chi_k \in F\setminus\{0\}, \qquad (2)$$

where F is an arbitrary field and the terms are denoted by $\mathbf{x}^{\mathbf{d}_j} = x_1^{d_{j,1}}\cdots x_n^{d_{j,n}}$ and $\mathbf{x}^{\mathbf{e}_k} = x_1^{e_{k,1}}\cdots x_n^{e_{k,n}}$. We analyze our variant of Zippel's sparse interpolation technique to recover the numerator and denominator. Zippel's technique [22, Section 4] determines the support of $f_i = f(x_1,\ldots,x_i, a_{i+1},\ldots,a_n)$ and $g_i = g(x_1,\ldots,x_i, a_{i+1},\ldots,a_n)$ incrementally from the support of f_{i-1} and g_{i-1}, where $a_2,\ldots,a_n \in F$ is a random anchor point. We will use Zippel's probabilistic assumption that each term $x_1^{d_{j,1}}\cdots x_{i-1}^{d_{j,i-1}}$, $1 \leq j \leq t_f$ and each term $x_1^{e_{k,1}}\cdots x_{i-1}^{e_{k,i-1}}$, $1 \leq k \leq t_g$ has a non-zero coefficient in f_{i-1} and g_{i-1}. The sets of possible terms in f_i can be restricted to

$$D_i = \{x_1^{d_{j,1}}\cdots x_{i-1}^{d_{j,i-1}} \cdot x_i^{\delta} \mid 1 \leq j \leq t_f,\\ 0 \leq \delta \leq \min(\deg(f) - d_{j,1} - \cdots - d_{j,i-1}, \deg_{x_i}(f))\}$$

and in g_i the term set can be restricted to

$$E_i = \{x_1^{e_{k,1}}\cdots x_{i-1}^{e_{k,i-1}} \cdot x_i^{\eta} \mid 1 \leq k \leq t_g,\\ 0 \leq \eta \leq \min(\deg(g) - e_{k,1} - \cdots - e_{k,i-1}, \deg_{x_i}(g))\}.$$

Here we make the assumption that f_{i-1} and g_{i-1} are correctly determined and, as said earlier, contain the full set of possible terms, that with high probability as we will inductively argue. We also assume that we know $\deg(f)$, $\deg_{x_i}(f)$, $\deg(g)$ and $\deg_{x_i}(g)$. Let $\mathbf{y}$ and $\mathbf{z}$ be the coefficient vectors of f_i and g_i for the terms in D_i and E_i. For any $l = 0,1,2\ldots$ and any point $p_1,\ldots,p_i \in F$ the value of the rational function

$$\gamma_{i,l} = f_i(p_1^l,\ldots,p_i^l)/g_i(p_1^l,\ldots,p_i^l) \in F\setminus\{0,\infty\}$$

constitutes a linear equation for the coefficient vector,

$$\left.\begin{aligned}\sum_{j,\delta} y_{j,\delta}(p_1^{d_{j,1}}\cdots p_{i-1}^{d_{j,i-1}}p_i^{\delta})^l\\ -\gamma_{i,l}\sum_{k,\eta} z_{k,\eta}(p_1^{e_{k,1}}\cdots p_{i-1}^{e_{k,i-1}}p_i^{\eta})^l = 0.\end{aligned}\right\} \qquad (3)$$

With $l = 0, \ldots, L-1$ the equations (3) form a linear system

$$[V_{i,L}(p_1, \ldots, p_i), -\Gamma_{i,L} W_{i,L}(p_1, \ldots, p_i)] \begin{bmatrix} \mathbf{y}^T \\ \mathbf{z}^T \end{bmatrix} = \mathbf{0}, \qquad (4)$$

where $\Gamma_{i,L}$ is a diagonal non-singular matrix of rational function values and $V_{i,L}$ and $W_{i,L}$ are Vandermonde matrices.

Provided f_{i-1} and g_{i-1} were correctly computed in the previous iterations, the coefficient vector $[\mathbf{y}\ \mathbf{z}]$ of f_i and g_i solves (4). Suppose now the f_i and g_i are relatively prime in $F[x_1, \ldots, x_i]$. For random anchor points $a_2, \ldots, a_n$, this will be true with high probability. In fact, f_i and g_i are then random projections of the primitive parts of f and g after removing their contents in $F[x_{i+1}, \ldots, x_n]$. We argue now that for random $p_1, \ldots, p_i$ and for $L \geq |D_i| \cdot |E_i|$ the linear system (4) has with high probability no second linearly independent solution.

We shall first assume that the random choices for $p_1, \ldots, p_i \in S \subset F$ are such that no two terms in D_i and no two terms in E_i evaluate at $x_\mu \leftarrow p_\mu$, $1 \leq \mu \leq i$, to the same element in F. Now let $\bar{f}$ and $\bar{g}$ be the polynomials for a second solution. Because $V_{i,L}$ and $W_{i,L}$ are Vandermonde matrices, we must have for $L \geq \max(|D_i|, |E_i|)$ that $\bar{f} \neq 0$ and $\bar{g} \neq 0$. Furthermore,

$$\forall l, 0 \leq l \leq L-1 \colon \frac{\bar{f}}{\bar{g}}(p_1^l, \ldots, p_i^l) = \frac{f_i}{g_i}(p_1^l, \ldots, p_i^l).$$

So

$$\forall l, 0 \leq l \leq L-1 \colon (\bar{f} g_i - f_i \bar{g})(p_1^l, \ldots, p_i^l) = 0. \qquad (5)$$

The terms of the polynomial $\bar{f} g_i - f_i \bar{g}$ are in

$$D_i E_i = \{\sigma \cdot \tau \mid \sigma \in D_i, \tau \in E_i\} \text{ with } |D_i E_i| \leq |D_i| \cdot |E_i|.$$

Note that for $i = 1$ we have $|D_1 E_1| \leq \deg_{x_1}(f) + \deg_{x_1}(g) + 1$. Finally, we assume that the random choices for $p_1, \ldots, p_i \in S$ are such that no two terms in $D_i E_i$ evaluate to the same value (which subsumes our earlier assumption). For $L \geq |D_i E_i|$ we then must have

$$\bar{f} g_i - f_i \bar{g} = 0,$$

because the coefficent vector of $\bar{f} g_i - f_i \bar{g}$ is by (5) a kernel vector in a square non-singular Vandermonde matrix. Thus $\bar{f}/\bar{g} = f_i/g_i$ and because of the degree conditions imposed on D_i and E_i, $(\bar{f}, \bar{g})$ cannot be a polynomial multiple of (f_i, g_i).

The linear system (4) may yield f_i and g_i for smaller L, and one may incrementally add equations until null space dimension 1 occurs. The above proof gives an upper bound on L, which can be used to diagnose bad random choices.

4.2 STLN-based numeric variant

Consider the rational function $f/g \in \mathbb{Q}(\boldsymbol{i})(x_1, \ldots, x_n)$ with $\gcd(f, g) = 1$ and f, g are represented as (2), where $F = \mathbb{Q}(\boldsymbol{i}) \subset \mathbb{C}$. In this subsection, Zippel's method [44] is implemented to numerically interpolate f and g from the approximate black box of f/g. If the actual supports of f_{i-1} and g_{i-1} are D_{i-1} and E_{i-1}, respectively, the candidate support of f_i and g_i can be obtained from D_{i-1}, E_{i-1} and $\deg(f), \deg(g)$. For the degree bounds of f and g are $\bar{d}$ and $\bar{e}$ respectively, we explain how to interpolate f and g variable by variable.

In order to interpolate f_i and g_i from $\bar{d}, \bar{e}$ and D_{i-1}, E_{i-1}, we solve the following two problems:

P1 Construct the candidate support of f_i and g_i from the degree bounds $\bar{d}, \bar{e}$ and D_{i-1}, E_{i-1},

P2 Compute the coefficients corresponding to the candidate support from **P1** and get the actual support of f_i and g_i.

Let $k = \min(\bar{d} - \deg_{x_i}(f), \bar{e} - \deg_{x_i}(g))$, and let $\bar{d}_i = \bar{d} - k$, $\bar{e}_i = \bar{e} - k$, then at least one of the equations below is true:

$$\bar{d}_i = \deg_{x_i}(f) \quad \text{or} \quad \bar{e}_i = \deg_{x_i}(g).$$

The possible terms in f_i and g_i can be constructed from k. Without loss of generality, we assume that $\bar{d}_i = \deg_{x_i}(f)$. In this case, the possible terms in f_i are

$$\bar{D}_i = \{x_1^{d_{j,1}} \cdots x_{i-1}^{d_{j,i-1}} \cdot x_i^{\delta} \mid 1 \leq j \leq t_f, \\ 0 \leq \delta \leq \min(\bar{d} - d_{j,1} - \cdots - d_{j,i-1}, \bar{d}_i)\} \qquad (6)$$

and the possible terms in g_i are

$$\bar{E}_i = \{x_1^{e_{r,1}} \cdots x_{i-1}^{e_{r,i-1}} \cdot x_i^{\eta} \mid 1 \leq r \leq t_g, \\ 0 \leq \eta \leq \min(\bar{e} - e_{r,1} - \cdots - e_{r,i-1}, \bar{e}_i)\}. \qquad (7)$$

We show that the interpolants $\bar{f}_i$ and $\bar{g}_i$ computed with $\bar{D}_i$ and $\bar{E}_i$ must be the form:

$$\bar{f}_i = q\, f_i, \ \bar{g}_i = q\, g_i, \quad \text{where } q \in \mathbb{C} \setminus \{0\}. \qquad (8)$$

Assume to the contrary that $\deg(q) > 0$. Because D_{i-1} and E_{i-1} are actual supports, we must have $\deg_{x_i}(q) > 0$. Hence, $\bar{d}_i \geq \deg_{x_i}(\bar{f}_i) > \deg_{x_i}(f_i)$, which is in contradiction with $\bar{d}_i = \deg_{x_i}(f)$. Property (8) justifies the use of $\bar{D}_i$ and $\bar{E}_i$ as the the set of possible terms of f_i and g_i.

Now let us show how to compute k. Denote the univariate polynomials $f^{[i]}$ and $g^{[i]}$ as:

$$f^{[i]} = f(a_1, \ldots, a_{i-1}, x_i, a_{i+1}, \ldots, a_n) = \sum_{j=0}^{\bar{d}} \psi_j x_i^j,$$
$$g^{[i]} = g(a_1, \ldots, a_{i-1}, x_i, a_{i+1}, \ldots, a_n) = \sum_{k=0}^{\bar{e}} \chi_k x_i^k,$$

where $\psi_s, \chi_t \in \mathbb{C}$. Given a random root of unity $p \in \mathbb{C}$, we compute the evaluations

$$\sigma_l = \frac{f^{[i]}}{g^{[i]}}(p^l) \in \mathbb{C} \setminus \{0, \infty\}, \quad l = 0, 1, \ldots, \bar{d} + \bar{e} + 1,$$

and construct the following linear equations

$$\sum_{j=0}^{\bar{d}} y_j p^{l\,j} - \sigma_l \sum_{k=0}^{\bar{e}} z_k p^{l\,k} = 0, \quad l = 0, 1, \ldots, \bar{d} + \bar{e} + 1.$$

The above equations form a linear system

$$G \begin{bmatrix} \mathbf{y}^T \\ \mathbf{z}^T \end{bmatrix} = [V_i, -\Gamma_i W_i] \begin{bmatrix} \mathbf{y}^T \\ \mathbf{z}^T \end{bmatrix} = \mathbf{0}, \qquad (9)$$

where V_i, W_i are Vandermonde matrices generated by the vectors $[1, p, \ldots, p^{\bar{d}}]^T$ and $[1, p, \ldots, p^{\bar{e}}]^T$, and where

$$\Gamma_i = \mathrm{diag}(\sigma_0, \sigma_1, \ldots, \sigma_{\bar{d}+\bar{e}+1}).$$

According to [19, Corollary 2.2] (see also Section 4.1), k is the rank deficiency of G. We can estimate k by checking the number of small singular values of G or finding the largest gap among the singular values [26].

Now we apply Structured Total Least Norm (STLN) [34] method to solve **P2**. Suppose the possible terms in f_i and g_i are

$$\bar{D}_i = \{x_1^{\bar{d}_{j,1}} \cdots x_i^{\bar{d}_{j,i}}, j = 1, 2, \ldots \bar{t}_f\}$$

and

$$\bar{E}_i = \{x_1^{\bar{e}_{j,1}} \cdots x_i^{\bar{e}_{j,i}}, j = 1, 2, \ldots \bar{t}_g\}$$

We assume that f_i and g_i are represented as

$$f_i = \sum_{j=1}^{\bar{t}_f} y_j x_1^{\bar{d}_{j,1}} \cdots x_i^{\bar{d}_{j,i}},\ g_i = \sum_{k=1}^{\bar{t}_g} z_k x_1^{\bar{e}_{k,1}} \cdots x_i^{\bar{e}_{k,i}}, \quad (10)$$

where y_j and z_k are unknown. Since some terms in $\bar{D}_i$ and $\bar{E}_i$ do not exist, the values of some y_j and z_k will be very small or zero. In other words, the terms corresponding to those y_j and z_k have zero coefficients in the true f_i and g_i.

The unknown coefficients y_j and z_k are computed via an STLN algorithm. Let $b_1, \ldots, b_i \in \mathbb{Z}_{>0}$ be sufficient large distinct prime numbers and s_j be random integers with $1 \le s_j < b_j$. We choose $p_j = \exp(2\pi \boldsymbol{i}/b_j)^{s_j} \in \mathbb{C}$ for $1 \le j \le i$ (see [14]). In the exact case, discussed in Section 4.1 above, we know that rank deficiency of the matrix in (4) is 1 for $L \ge \bar{t}_f \bar{t}_g$ evaluations. In fact, $\bar{t}_f \bar{t}_g$ is an upper bound which guarantees that the rank deficiency of the matrix (4) is no more than 1. For the random examples shown in Table 1 and Table 2, our algorithm only needs $L = \bar{t}_f + \bar{t}_g + 10$ probes to achieve a unique rational function solution.

The structured matrix input for the STLN algorithm is

$$G(\mathbf{c}) = [V_{i,L}(p_1, \ldots, p_i), -\mathrm{diag}(\mathbf{c}) W_{i,L}(p_1, \ldots, p_i)] \quad (11)$$

(cf. (4)), where $L = \bar{t}_f + \bar{t}_g + \xi$ ($\xi \ge 1$), $V_{i,L}, W_{i,L}$ are Vandermonde matrices, $\mathbf{c} = [\widetilde{\gamma}_{i,0}, \ldots, \widetilde{\gamma}_{i,L-1}]^T$, and

$$\widetilde{\gamma}_{i,l} \approx \frac{f_i(p_1^l, \ldots, p_i^l)}{g_i(p_1^l, \ldots, p_i^l)} \text{ for } l = 0, \ldots, L-1,$$

which are the noisy evaluations for the rational function f/g.

We briefly present the STLN [30, 31] method to compute a singular matrix

$$G(\widetilde{\mathbf{c}}) = [V_{i,L}(p_1, \ldots, p_i), -\mathrm{diag}(\widetilde{\mathbf{c}}) W_{i,L}(p_1, \ldots, p_i)]$$

such that $\|\widetilde{\mathbf{c}} - \mathbf{c}\|$ is minimized. We choose the column $\mathbf{b}$ as the $(m + \bar{t}_f)$-th column corresponding to the absolutely largest component in the last $\bar{t}_g$ elements of $\mathbf{v}$ with $\mathbf{v}$ is the last singular vector of $G(\mathbf{c})$ [30]. The matrix $A(\mathbf{c})$ consists of the remaining columns of $G(\mathbf{c})$. Our problem can be transformed as the following polynomial optimization problem (POP):

$$\left.\begin{array}{l} \min_{\mathbf{z},\mathbf{u}} \|\mathbf{z}\| \\ \text{s. t.} \quad A(\mathbf{c}+\mathbf{z})\mathbf{u} = b(\mathbf{c}+\mathbf{z}). \end{array}\right\} \quad (12)$$

Solving (12) by STLN requires two matrices P and Y with the following properties:

$$\mathbf{b}(\mathbf{z}) = P\,\mathbf{z}, \quad A(\mathbf{z})\,\mathbf{u} = Y(\mathbf{u})\,\mathbf{z}. \quad (13)$$

Let $\hat{\mathbf{w}}_j, 1 \le j \le L$, be formed by the j-th row of the matrix $W_{i,L}$, after deleting the element corresponding to the column $\mathbf{b}$. Let the vector $\hat{\mathbf{u}}$ be the subvector consisting of the last $\bar{t}_g - 1$ elements of $\mathbf{u}$. Then P and Y are diagonal matrices, where

$$P = \mathrm{diag}(1, v_i, {v_i}^2, \ldots, {v_i}^{L-1}) \quad \text{with} \quad v_i = p_1^{\bar{e}_{m,1}} \cdots p_i^{\bar{e}_{m,i}}$$

and

$$Y = \mathrm{diag}(\hat{\mathbf{w}}_1\,\hat{\mathbf{u}},\ \hat{\mathbf{w}}_2\,\hat{\mathbf{u}},\ \ldots,\ \hat{\mathbf{w}}_L\,\hat{\mathbf{u}}).$$

With the matrices P and Y, we now can carry out the STLN method to solve the problem (12). The details are described in [30, 31].

The solution $\mathbf{u}$ to (12) constitutes the coefficient vector of f_i and g_i. One obtains the exact support of f_i and g_i by removing terms whose coefficients y_j or z_k are smaller than the given tolerance.

Algorithm *Zippel Numerical Interpolation of Rational Functions*

Input:
- $\frac{f(x_1,\ldots,x_n)}{g(x_1,\ldots,x_n)} \in \mathbb{C}(x_1, \ldots, x_n)$ input as a black box.
- $(x_1, \ldots, x_n)$: an ordered list of variables in f/g.
- $\bar{d}, \bar{e}$: degree bounds $\bar{d} \ge \deg(f)$ and $\bar{e} \ge \deg(g)$.
- $\epsilon \in \mathbb{R}_{>0}$: the given tolerance.

Output:
- $f(x_1, \ldots, x_n)/c$ and $g(x_1, \ldots, x_n)/c$, where $c \in \mathbb{C}$.

1. Initialize the anchor points and the support of f and g: choose $a_1, a_2, \ldots, a_n$ as random roots of unity, let $D_0 = \{1\}$ and $E_0 = \{1\}$.

2. For $i = 1, 2, \ldots, n$ do:
   Interpolate the polynomials f_i and g_i as follows:

   (a) We compute $\bar{d}_i$ and $\bar{e}_i$ described as above, which are the possible degrees of f and g for the variable x_i:

   Choose a random root of unity p and get the approximate evaluation:

   $$\widetilde{\gamma}_{i,l} \approx \frac{f(a_1, \ldots, a_{i-1}, p^l, a_{i+1}, \ldots, a_n)}{g(a_1, \ldots, a_{i-1}, p^l, a_{i+1}, \ldots, a_n)}, \quad l = 0, 1, 2, \ldots, \bar{d} + \bar{e} + 1.$$

   Construct the matrix G in (9) from $\widetilde{\gamma}_{i,l}$ and p. Compute the SVD of G and find k. Let $\bar{d}_i = \bar{d} - k$, and $\bar{e}_i = \bar{e} - k$.

   (b) From $\bar{d}$, $\bar{d}_i$ and D_{i-1}, get the possible terms $\bar{D}_i$ of f_i, similarly get the possible terms $\bar{E}_i$ of g_i from $\bar{e}$, $\bar{e}_i$ and E_{i-1}.

   (c) Using STLN method, interpolate f_i and g_i and get their actual terms D_i, E_i:

   Choose random roots of unity $p_1, \ldots, p_i$. For $l = 0, 1, 2, \ldots$, compute approximate values:

   $$\widetilde{\gamma}_{i,l} \approx f_i(p_1^l, \ldots, p_i^l)/g_i(p_1^l, \ldots, p_i^l),$$

   and construct the matrix G in (11) from $\widetilde{\gamma}_{i,l}$ and $\bar{D}_i, \bar{E}_i$.

   Compute the almost nearest singular matrix $\tilde{G}$ by STLN method and get the solution $\mathbf{u}$ in (12).

   Obtain f_i and g_i from $\mathbf{u}$ and $\bar{D}_i$, $\bar{E}_i$. Check whether f_i and g_i are approximate relative prime (e.g., by our algorithm [30]).

   If this is the case, get D_i and E_i of f_i and g_i by cutting off the small terms according to ϵ.

   Otherwise, go back step 2a to choose new points $p, p_1, \ldots, p_i$ to interpolate f_i and g_i.

3. With the support of f_n and g_n, interpolate $f(x_1,\ldots,x_n)/c$ and $g(x_1,\ldots,x_n)/c$ again to improve the accuracy of the coefficients:

   (a) Construct the matrix G from the evaluations $\gamma_{n,l}$ and the exact terms D_n and E_n. Compute $\tilde{G}$ and the solution $\mathbf{u}$ in (12) using STLN method.

   (b) Obtain $f(x_1,\ldots,x_n)/c$ and $g(x_1,\ldots,x_n)/c$ from $\mathbf{u}$ and D_n, E_n. □

4.3 Experiments

Algorithm ZNIPR has been implemented in Maple and the performance is reported in the following two tables. All examples in Table 1 and Table 2 are run in Maple 10 under Windows for *Digits*:=10. In Table 1 we exhibit the performance of Algorithm ZNIPR for recovering univariate rational functions from a black box for noisy values. A univariate rational function can also be interpolated from approximate oversampled values by solving the Toeplitz-like linear system [23]. In [29], the STLN method is also applied to solve that overdetermined system. Comparing the backward errors, ZNIPR can get better results than the method in [29], since ZNIPR takes advantage of the sparsity of rational functions in Step 3. For each example, we construct two relatively prime polynomials with random integer coefficients in the range $-5 \le c \le 5$. Here *Random Noise* denotes the noise in this range randomly added to the black box of f/g; d_f and d_g denote the degree of the numerator and denominator respectively; t_f and t_g denote the number of terms of the numerator and denominator respectively; whereas *error (ZNIRP)* and *error (KY'07)* are relative errors, namely $(\|\tilde{f}-f\|_2^2+\|\tilde{g}-g\|_2^2)/(\|f\|_2^2+\|g\|_2^2)$, computed by our algorithm and the algorithm in [29].

Ex.	*Random Noise*	d_f, d_g	t_f, t_g	*error (ZNIPR)*	*error (KY'07)*
1	$10^{-4} \sim 10^{-2}$	3, 3	1, 3	1.46102e–7	6.52633e–7
2	$10^{-5} \sim 10^{-3}$	4, 5	2, 4	6.90952e–7	2.38658e–5
3	$10^{-6} \sim 10^{-4}$	8, 3	4, 3	6.82760e–9	5.02298e–8
4	$10^{-5} \sim 10^{-3}$	10, 10	4, 4	1.05930e–6	1.16975e–4
5	$10^{-6} \sim 10^{-4}$	3, 15	2, 6	1.32383e–8	8.99870e–6
6	$10^{-6} \sim 10^{-4}$	20, 20	5, 5	2.31127e–9	4.92399e–8
7	$10^{-6} \sim 10^{-4}$	30, 7	6, 3	1.07707e–8	2.2445 e–7
8	$10^{-7} \sim 10^{-5}$	5, 40	4, 7	2.68987e–11	2.00818e–8
9	$10^{-7} \sim 10^{-5}$	50, 50	5, 5	1.02862e–11	8.34669e–10
10	$10^{-9} \sim 10^{-7}$	80, 80	6, 6	9.80489e–15	2.31186e–12
11	$10^{-9} \sim 10^{-7}$	100, 100	7, 7	1.59983e–15	5.84762e–7

Table 1: Algorithm performance on benchmarks (univariate case)

In Table 2 we exhibit the performance of Algorithm ZNIPR on multivariate inputs. For each example, we construct two relatively prime multivariate polynomials with random integer coefficients in the range $-5 \le c \le 5$. Here *Random Noise* denotes the noise in this range randomly added to the black box of f/g; d_f and d_g denote the degree of the numerator and denominator respectively; t_f and t_g denote the number of terms of the numerator and denominator respectively; n denotes the number of the variables of the rational functions; N denotes the number of the black box probes needed to interpolate the approximate multivariate rational function; finally, *error (ZNIRP)* denotes the relative backward error computed by our algorithm. Example 13 is one polynomial test (c.f. [14]), which demonstrates that Algorithm ZNIPR can also interpolate sparse multivariate polynomials from noisy values.

Ex.	*Random Noise*	d_f, d_g	t_f, t_g	n	N	*error (ZNIPR)*
1	$10^{-5} \sim 10^{-3}$	1, 1	2, 2	2	138	7.05479e–8
2	$10^{-5} \sim 10^{-3}$	2, 2	3, 3	2	140	4.29232e–7
3	$10^{-5} \sim 10^{-3}$	1, 4	2, 4	3	247	5.91114e–7
4	$10^{-6} \sim 10^{-4}$	5, 2	10, 6	3	308	4.92402e–8
5	$10^{-7} \sim 10^{-5}$	7, 7	25, 25	5	1456	4.01293e–7
6	$10^{-7} \sim 10^{-5}$	10, 3	15, 5	8	4781	3.04625e–8
7	$10^{-7} \sim 10^{-5}$	5, 13	4, 6	10	1498	5.37480e–8
8	$10^{-7} \sim 10^{-5}$	20, 20	7, 7	15	3658	3.36386e–10
9	$10^{-8} \sim 10^{-6}$	30, 30	6, 6	20	6391	1.20737e–12
10	$10^{-8} \sim 10^{-6}$	40, 40	6, 6	5	2810	1.02589e–10
11	$10^{-8} \sim 10^{-6}$	60, 60	7, 7	4	2862	3.51967e–13
12	$10^{-8} \sim 10^{-6}$	80, 80	6, 6	10	6864	7.59227e–13
13	$10^{-8} \sim 10^{-6}$	60, 0	6, 1	20	2862	2.00141e–12

Table 2: Algorithm performance on benchmarks (multivariate case)

Note that the numbers N of black box evaluations needed in Table 2 are somewhat high due to the loose degree estimation in Step 2(a) in Algorithm ZNIPR. We are investigating how to compute sharper estimates for the degrees of the terms in the sets $\bar{D}_i$ and $\bar{E}_i$.

Acknowledgement: We thank Terence Tao for his comments on the condition number of random Fourier matrices.

5. REFERENCES

[1] Bharucha-Reid, A. T., and Sambandham, M. *Random Polynomials*. Academic Press, INC, London, England, 1986.

[2] Chen, L., Eberly, W., Kaltofen, E., Saunders, B. D., Turner, W. J., and Villard, G. Efficient matrix preconditioners for black box linear algebra. *Linear Algebra and Applications 343–344* (2002), 119–146. Special issue on *Structured and Infinite Systems of Linear Equations*, edited by P. Dewilde, V. Olshevsky and A. H. Sayed.

[3] Chen, Z., and Dongarra, J. J. Condition numbers of Gaussian random matrices. *SIAM Journal on Matrix Analysis and Applications 27*, 3 (2005), 603–620.

[4] Corless, R. M., Gianni, P. M., Trager, B. M., and Watt, S. M. The singular value decomposition for polynomial systems. In *Proc. 1995 Internat. Symp. Symbolic Algebraic Comput. ISSAC'95* (New York, N. Y., 1995), A. H. M. Levelt, Ed., ACM Press, pp. 96–103.

[5] Corless, R. M., Watt, S. M., and Zhi, L. QR factoring to compute the GCD of univariate approximate polynomials. *IEEE Transactions on Signal Processing 52* (Dec. 2004), 3394–3402.

[6] DeMillo, R. A., and Lipton, R. J. A probabilistic remark on algebraic program testing. *Information Process. Letters 7*, 4 (1978), 193–195.

[7] Dumas, J.-G., Ed. *ISSAC MMVI Proc. 2006 Internat. Symp. Symbolic Algebraic Comput.* (New York, N. Y., 2006), ACM Press.

[8] Dunaway, D. K. Calculation of zeros of a real polynomial through factorization using Euclid's algorithm. *SIAM J. Numer. Anal. 11*, 6 (1974), 1087–1104.

[9] Edelman, A., and Kostlan, E. How many zeros of a random polynomial are real. *Bulletin of the American Mathematical Society 32*, 1 (1995), 1–37.

[10] Farahmand, K. Algebraic polynomials with random coefficients. *Journal of Applied Mathematics and Stochastic Analysis 15*, 1 (2002), 83–88.

[11] Gao, S. Factoring multivariate polynomials via partial differential equations. *Math. Comput. 72*, 242 (2003), 801–822.

[12] Gao, S., Kaltofen, E., May, J. P., Yang, Z., and Zhi, L. Approximate factorization of multivariate polynomials via differential equations. In *ISSAC 2004 Proc. 2004 Internat. Symp. Symbolic Algebraic Comput.* (New York, N. Y., 2004), J. Gutierrez, Ed., ACM Press, pp. 167–174. ACM SIGSAM's ISSAC 2004 Distinguished Student Author Award (May and Yang).

[13] von zur Gathen, J. Irreducibility of multivariate polynomials. *J. Comput. System Sci. 31* (1985), 225–264.

[14] Giesbrecht, M., Labahn, G., and Lee, W. Symbolic-numeric sparse interpolation of multivariate polynomials. In Dumas [7], pp. 116–123.

[15] Giusti, M., and Éric Schost. Solving some overdetermined polynomial systems. In *Proc. 1999 Internat. Symp. Symbolic Algebraic Comput. (ISSAC'99)* (New York, N. Y., 1999), S. Dooley, Ed., ACM Press, pp. 1–8.

[16] Golub, G. H., and Van Loan, C. F. *Matrix Computations*, third ed. Johns Hopkins University Press, Baltimore, Maryland, 1996.

[17] Heintz, J., Krick, T., Puddu, S., Sabia, J., and Waissbein, A. Deformation techniques for efficient polynomial equation solving. *J. Complex. 16*, 1 (2000), 70–109.

[18] Kac, M. On the average number of real roots of a random algebraic equation. *Bulletin of the American Mathematical Society 49* (1943), 314–320.

[19] Kai, H. Rational interpolation and its ill-conditioned property. In Wang and Zhi [41], pp. 47–53.

[20] Kaltofen, E. Effective Hilbert irreducibility. *Information and Control 66* (1985), 123–137.

[21] Kaltofen, E. Greatest common divisors of polynomials given by straight-line programs. *J. ACM 35*, 1 (1988), 231–264.

[22] Kaltofen, E. Factorization of polynomials given by straight-line programs. In *Randomness and Computation*, S. Micali, Ed., vol. 5 of *Advances in Computing Research*. JAI Press Inc., Greenwhich, Connecticut, 1989, pp. 375–412.

[23] Kaltofen, E. Asymptotically fast solution of Toeplitz-like singular linear systems. In *Proc. 1994 Internat. Symp. Symbolic Algebraic Comput. (ISSAC'94)* (New York, N. Y., 1994), ACM Press, pp. 297–304. Journal version in [24].

[24] Kaltofen, E. Analysis of Coppersmith's block Wiedemann algorithm for the parallel solution of sparse linear systems. *Math. Comput. 64*, 210 (1995), 777–806.

[25] Kaltofen, E. Effective Noether irreducibility forms and applications. *J. Comput. System Sci. 50*, 2 (1995), 274–295.

[26] Kaltofen, E., May, J., Yang, Z., and Zhi, L. Approximate factorization of multivariate polynomials using singular value decomposition. Manuscript, 22 pages. Submitted, Jan. 2006.

[27] Kaltofen, E., and Saunders, B. D. On Wiedemann's method of solving sparse linear systems. In *Proc. AAECC-9* (Heidelberg, Germany, 1991), H. F. Mattson, T. Mora, and T. R. N. Rao, Eds., vol. 539 of *Lect. Notes Comput. Sci.*, Springer Verlag, pp. 29–38.

[28] Kaltofen, E., and Villard, G. On the complexity of computing determinants. *Computational Complexity 13*, 3-4 (2004), 91–130.

[29] Kaltofen, E., and Yang, Z. On exact and approximate interpolation of sparse rational functions. In *ISSAC 2007 Proc. 2007 Internat. Symp. Symbolic Algebraic Comput.* (New York, N. Y., 2007), C. W. Brown, Ed., ACM Press. To appear.

[30] Kaltofen, E., Yang, Z., and Zhi, L. Approximate greatest common divisors of several polynomials with linearly constrained coefficients and singular polynomials. In Dumas [7], pp. 169–176.

[31] Kaltofen, E., Yang, Z., and Zhi, L. Structured low rank approximation of a Sylvester matrix. In Wang and Zhi [41], pp. 69–83.

[32] Lemmerling, P., Mastronardi, N., and Van Huffel, S. Fast algorithm for solving the Hankel/Toeplitz Structured Total Least Squares problem. *Numerical Algorithms 23* (2000), 371–392.

[33] Mourrain, B., and Trebuchet, P. Generalized normal forms and polynomial system solving. In *ISSAC'05 Proc. 2005 Internat. Symp. Symbolic Algebraic Comput.* (New York, N. Y., 2005), M. Kauers, Ed., ACM Press, pp. 253–260.

[34] Park, H., Zhang, L., and Rosen, J. B. Low rank approximation of a Hankel matrix by structured total least norm. *BIT 39*, 4 (1999), 757–779.

[35] Reid, G., and Zhi, L. Solving nonlinear polynomial system via symbolic-numeric elimination method. In *Proceedings of the International Conference on Polynomial System Solving* (Paris, France, 2004).

[36] Sasaki, T., and Noda, M. Approximate square-free decomposition and root-finding of ill-conditioned algebraic equations. *J. Inf. Process. 12*, 2 (1989), 159–168. Information Processing Society of Japan, Tokyo.

[37] Sasaki, T., Suzuki, M., Kolář, M., and Sasaki, M. Approximate factorization of multivariate polynomials and absolute irreducibility testing. *Japan J. of Industrial and Applied Mathem. 8*, 3 (Oct. 1991), 357–375.

[38] Schwartz, J. T. Fast probabilistic algorithms for verification of polynomial identities. *J. ACM 27* (1980), 701–717.

[39] Tao, T., and Vu, V. On the condition number of a randomly perturbed matrix. In *Proc. 39th Annual ACM Symp. Theory Comput.* (New York, N.Y., 2007), ACM Press. to appear.

[40] Šparo, D. I., and Šur, M. G. On the distribution of roots of random polynomials. *Vestn. Mosk. Univ., Ser. 1: Mat., Mekh.* (1962), 40–53.

[41] Wang, D., and Zhi, L., Eds. *Symbolic-Numeric Computation*. Trends in Mathematics. Birkhäuser Verlag, Basel, Switzerland, 2007.

[42] Wiedemann, D. Solving sparse linear equations over finite fields. *IEEE Trans. Inf. Theory* IT-*32* (1986), 54–62.

[43] Zippel, R. Probabilistic algorithms for sparse polynomials. In *Symbolic and Algebraic Computation* (Heidelberg, Germany, 1979), vol. 72 of *Lect. Notes Comput. Sci.*, Springer Verlag, pp. 216–226. Proc. EUROSAM '79.

[44] Zippel, R. Interpolating polynomials from their values. *J. Symbolic Comput. 9*, 3 (1990), 375–403.

[45] Zippel, R. E. *Probabilistic algorithms for sparse polynomials*. PhD thesis, Massachusetts Inst. of Technology, Cambridge, USA, Sept. 1979.