Sparse Interpolation With Errors in Chebyshev Basis
Beyond Redundant-Block Decoding*

Erich L. Kaltofen and Zhi-Hong Yang

Department of Mathematics, North Carolina State University,
Raleigh, North Carolina 27695-8205, USA
Department of Computer Science, Duke University,
Durham, North Carolina 27708-0129, USA
kaltofen@ncsu.edu; https://kaltofen.math.ncsu.edu/
kaltofen@cs.duke.edu; https://users.cs.duke.edu/~elk27
zyang28@ncsu.edu; zy99@cs.duke.edu

Abstract
We present sparse interpolation algorithms for recovering a polynomial with \( \leq B \) terms from \( N \) evaluations at distinct values for the variable when \( \leq E \) of the evaluations can be erroneous. Our algorithms perform exact arithmetic in the field of scalars \( K \) and the terms can be standard powers of the variable or Chebyshev polynomials, in which case the characteristic of \( K \) is \( \neq 2 \). Our algorithms return a list of valid sparse interpolants for the \( N \) support points and run in polynomial-time. For standard power basis our algorithms sample at \( N = \left\lfloor \frac{4}{3}E + 2 \right\rfloor B \) points, which are fewer points than \( N = 2(E + 1)B - 1 \) given by Kaltofen and Pernet in 2014. For Chebyshev basis our algorithms sample at \( N = \left\lfloor \frac{2}{3}E + 2 \right\rfloor B \) points, which are also fewer than the number of points required by the algorithm given by Arnold and Kaltofen in 2015, which has \( N = 74\left\lfloor \frac{E}{13} + 1 \right\rfloor \) for \( B = 3 \) and \( E \geq 222 \). Our method shows how to correct 2 errors in a block of \( 4B \) points for standard basis and how to correct 1 error in a block of \( 3B \) points for Chebyshev Basis.

1. Introduction

Let \( f(x) \) be a polynomial with coefficients from a field \( K \) (of characteristic \( \neq 2 \)),

\[
f(x) = \sum_{j=1}^{t} c_j T_{\delta_j}(x) \in K[x], \quad 0 \leq \delta_1 < \delta_2 < \cdots < \delta_t = \deg(f), \forall j, 1 \leq j \leq t: c_j \neq 0,
\]

*This research was supported by the National Science Foundation under Grant CCF-1717100 (Kaltofen and Yang).
where \( T_d(x) \) is the Chebyshev Polynomial of the First Kind (of degree \( d \) for \( d \geq 0 \), defined by the recurrence

\[
\begin{bmatrix}
T_d(x) \\
T_{d+1}(x)
\end{bmatrix} =
\begin{bmatrix}
0 & 1 \\
-1 & 2x
\end{bmatrix}^d
\begin{bmatrix}
1 \\
x
\end{bmatrix}
\text{ for } d \in \mathbb{Z}.
\]  

(2)

We say that \( f(x) \) is Chebyshev-1 \( t \)-sparse. We wish to compute the term degrees \( \delta_j \) and the coefficients \( c_j \) from values of \( a_i = f(\zeta_i) \) for \( i = 1, 2, \ldots \), where the distinct arguments \( \zeta_i \in \mathbb{K} \) can be chosen by the algorithms; the latter is the setting of Prony-like sparse interpolation methods. Our objective is to interpolate with a number of points that is proportional to the sparsity \( t \) of \( f \). The algorithms have as input an upper bound \( B \geq t \) for the sparsity, otherwise the zero polynomial (of sparsity 0) is indistinguishable from \( f(x) = \prod_i (x - \zeta_i) \) at \( \leq \deg(f) \) evaluation points \( a_i = 0 \). The algorithms by [Lakshman Y. N. and Saunders 1995; Arnold and Kaltofen 2015; Imamoglu, Kaltofen, and Yang 2018], based on Prony-like interpolation [Prony III (1795); Ben-Or and Tiwari 1988; Kaltofen and Lee 2003], can interpolate \( f(x) \) (see (1)) from \( 2B \) values at points \( \zeta_i = T_i(\beta) = (\omega^i + 1/\omega^i)/2 \) for \( i = 0, 1, \ldots, 2B - 1 \) where \( \beta = (\omega + 1/\omega)/2 \) with \( \omega \in \mathbb{K} \) such that \( \omega^{\delta_j} \neq \omega^{\delta_k} \) for all \( 1 \leq j < k \leq t \).

Like Prony’s original algorithm, our algorithms utilize an algorithm for computing roots in \( \mathbb{K} \) of polynomials with coefficients in \( \mathbb{K} \) and logarithms to base \( \omega \). More precisely, one utilizes an algorithm that on input \( \omega \) and \( \omega^d \) for an integer \( d \in \mathbb{Z} \) computes \( d \), possibly modulo the finite multiplicative order \( \eta \) of \( \omega \) (\( \omega^\eta = 1 \) minimally) [Imamoglu and Kaltofen 2020]. We note that in [Arnold and Kaltofen 2015] we show that one may instead use the odd-indexed argument \( T_{2i+1}(\beta) \) for \( i = 0, 1, \ldots, 2B - 1 \), provided \( \omega^{2\delta_j + 1} \neq \omega^{2\delta_k + 1} \) for all \( 1 \leq j < k \leq t \).

Here we consider the case when the evaluations \( a_i \), which we think of being computed by probing a black box that evaluates \( f \), can have sporadic errors. We write \( \hat{a}_i \) for the black box values, which at some unknown indices \( \ell \) can have \( \hat{a}_\ell \neq a_\ell \). In the plot in Fig. 1 below, which is for the range \(-1 \leq x \leq 1 \), the purple function is \( T_{15}(x) - 2T_{11}(x) + T_{2}(x) \) that fits 37 of the 40 values, while the red model is a polynomial least squares fit of degree \( \leq 19 \). The red function captures 3 possible outliers, resulting in a model which has a lower accuracy on the remaining 37 data points.

Figure 1: Sparse Chebyshev-1 polynomial fit after removing 3 errors vs. polynomial least squares fit

We shall assume that we have an upper bound \( E \) for the number of errors on a batch of \( N \) evaluations. Therefore our sequence of black box calls has a non-stochastic error rate \( \leq E/N \). We shall also assume that the black box for \( f \) does not return stochastic errors, meaning that if \( \hat{a} \neq f(\zeta) \) then a second evaluation of the black box at \( \zeta \) produces the same erroneous \( \hat{a} \). Furthermore, we perform list-interpolation which produces a valid list of sparse interpolants for the black box values with errors, analogously to list-decoding error correcting.
codes. We restrict to algorithms that run in polynomial time in $B$ and $E$ ($N$ is computed by the algorithms), which limits the list length to polynomial in $B$ and $E$.

A simple sparse list-interpolation algorithm with errors evaluates $E + 1$ blocks of $2B$ arguments, which produce $N = (E + 1)2B$ black box values $\hat{a}_{i,\sigma}$ at the arguments

$$
\begin{align*}
T_1(\beta_1), & \quad T_3(\beta_1), \quad \ldots, \quad T_{4B-1}(\beta_1), \\
T_1(\beta_2), & \quad T_3(\beta_2), \quad \ldots, \quad T_{4B-1}(\beta_2), \\
\vdots & \quad \vdots \\
T_1(\beta_{E+1}), & \quad T_3(\beta_{E+1}), \quad \ldots, \quad T_{4B-1}(\beta_{E+1}),
\end{align*}
$$

where $\beta_\sigma = (\omega_\sigma + 1/\omega_\sigma)/2$ and where the arguments in (3) are selected distinct: $T_{2i+1}(\beta_\sigma) \neq T_{2m+1}(\beta_\tau)$ for $i \neq m$ and $\sigma \neq \tau$ ($\iff \omega_{2i+1} \neq \omega_{2m+1}$). If we have for all $\omega_\sigma$ distinct term values $\omega_{\delta_j} \neq \omega_{\delta_k}$ ($j \neq k$) then the algorithm in [Arnold and Kaltofen 2015] can recover $f$ from those lines in (3) at which the black box does not evaluate to an error, because we assume $\leq E$ errors there is such a block of good arguments/values. Other blocks with errors may lead to a different $t$-sparse Chebyshev-1 interpolant with $t \leq B$. The goal is to recover $f$ (and possible other sparse interpolants with $\leq E$ errors) from $N < (E + 1)2B$ evaluations.

In [Arnold and Kaltofen 2015] we give algorithms for the following bounds $B, E$:

$$
\begin{align*}
B = 1: \forall E \geq 57: \quad N &= 23\left\lfloor \frac{E}{14} + 1 \right\rfloor < 2(E + 1) = 2B(E + 1); \quad \frac{23}{14} \leq 1.65, \\
B = 2: \forall E \geq 86: \quad N &= 43\left\lfloor \frac{E}{12} + 1 \right\rfloor < 4(E + 1) = 2B(E + 1); \quad \frac{43}{12} \leq 3.59, \\
B = 3: \forall E \geq 222: \quad N &= 74\left\lfloor \frac{E}{13} + 1 \right\rfloor < 6(E + 1) = 2T(E + 1); \quad \frac{74}{13} \leq 5.70.
\end{align*}
$$

The evaluation counts (4) are derived by using the method of [Kaltofen and Pernet 2014]: subsampling at all subsequences $x \leftarrow T_{r+ia}(\beta)$ of arguments whose indices are arithmetic progressions to locate a subsequence without an error. The counts (4) are established by explicitly computed lengths for the Erdős-Turán Problem for arithmetic progressions of length $\leq 9$. Here we give an algorithm that recovers $f$ (and possible other sparse interpolants) for all $B \geq 1, E \geq 1$ bounds from

$$
N = \left\lfloor \frac{3}{2}E + 2 \right\rfloor B
$$

evaluations with $\leq E$ errors. Our new algorithm uses fewer evaluations than (4). We show that one can list-interpolate from $3B$ points correcting a single error, which with blocking yields (5). We correct one error from $3B$ points by deriving a non-trivial univariate polynomial for the value as a variable in each possible position.

Our technique applies to Prony’s original problem of interpolating a $t$-sparse polynomial with $t \leq B$ in power basis $1, x, x^2, \ldots$ in the presence of erroneous points. In [Kaltofen and Pernet 2014, Lemma 2] it was shown that from $(E + 1)2B - 1$ points one can correct $\leq E$ errors. Here we show that

$$
N = \left\lfloor \frac{4}{3}E + 2 \right\rfloor B
$$

points suffice to correct $\leq E$ errors. The counts (6) are achieved by correcting $\leq 2$ errors from $4B$ points and blocking. We correct 2 errors at $4B$ points by deriving a bivariate
Pham system for variables in place of the values in all possible error locations, which yields a bounded number of possible value pairs among which are the actual values. We note that for $E = 2$ the count $4B$ is smaller than the values $n_{2B,2}$ in [Kaltofen and Pernet 2014, Table 1], which are the counts for having a clean arithmetic progression of length $2B$ in the presence of 2 errors.

Finally we note that our sparse list-interpolation algorithms are interpolation algorithms over the reals $K = \mathbb{R}$ if $\omega \sigma > 1$ (or $\omega > 0$ when $f$ is in power basis) and $N \geq 2B + 2E$, that is, there will only be a single sparse interpolant computed by our algorithms. Uniqueness is a consequence of Descartes’s Rule of Signs and its generalization to polynomials in orthogonal bases by Obrechkoff’s Theorem of 1918 [Dimitrov and Rafaeli 2009] (see also Corollary 2 in [Kaltofen and Pernet 2014] and Corollary 2.4 in [Arnold and Kaltofen 2015]). Over fields with roots of unity, the sparse list-interpolation problem for the power bases with $< (2E + 1)2B$ points can have more than a single $B$-sparse solution [Kaltofen and Pernet 2014, Theorem 3], which is also true for the Chebyshev base as shown by Example 3.3.

2. Sparse Interpolation in Standard Power Basis with Error Correction

2.1. Correcting One Error

Let $K$ be a field of scalars. Let $f(x) \in K[x, x^{-1}]$ be a sparse univariate Laurent polynomial represented by a black box and it is equal to:

$$f(x) = \sum_{j=1}^{t} c_j x^{\delta_j}, \quad \delta_1 < \delta_2 < \cdots < \delta_t = \deg(f), \forall j, 1 \leq j \leq t: c_j \neq 0. \quad (7)$$

We assume that the black box for $f$ returns the same value when probed multiple times at the same input. Let $B$ be an upper bound on the sparsity of $f(x)$ and $D \geq |\delta_j|$ for all $1 \leq j \leq t$. Choose a point $\omega \in K \setminus \{0\}$ such that:

1. $\omega$ has order $\geq 2D + 1$, meaning that $\forall \eta, 1 \leq \eta \leq 2D: \omega^{\eta} \neq 1$.

2. $\omega^{i_1} \neq \omega^{i_2}$ for all $1 \leq i_1 < i_2 \leq 3B$.

The first condition is an input specification of the Integer Logarithm Algorithm (see Algorithm 2.1) that computes $\delta_j$ from $\omega^{\delta_j}$. The second condition guarantees that the inputs probed at the black box are distinct so that we don’t get the same error at different locations.

For $i = 1, 2, \ldots, 3B$, let $\hat{a}_i$ be the output of the black box for $f$ probed at input $\omega^i$. Assume there is at most one error in the evaluations, that is, there exists $1$ index $i \in \{1, 2, \ldots, 3B\}$ such that $\hat{a}_i \neq f(\omega^i)$. We present an algorithm to compute a list of sparse polynomials which contains $f$. 

---

4
For $r = 1, \ldots, B$, let $H_r$ denote the following $(B + 1) \times (B + 1)$ Hankel matrix:

$$H_r = \begin{bmatrix}
\hat{a}_r & \hat{a}_{r+1} & \cdots & \hat{a}_{r+B-1} & \hat{a}_{r+B} \\
\hat{a}_{r+1} & \hat{a}_{r+2} & \cdots & \hat{a}_{r+B} & \hat{a}_{r+B+1} \\
\vdots & \vdots & \ddots & \vdots & \vdots \\
\hat{a}_{r+B-1} & \hat{a}_{r+B} & \cdots & \hat{a}_{r+2B-2} & \hat{a}_{r+2B-1} \\
\hat{a}_{r+B} & \hat{a}_{r+B+1} & \cdots & \hat{a}_{r+2B-1} & \hat{a}_{r+2B}
\end{bmatrix} \in \mathbb{K}^{(B+1)\times(B+1)}. \quad (8)$$

Let $\ell$ be the error location, i.e., $\hat{a}_\ell \neq f(\omega^\ell)$. There are three cases to be considered:

Case 1: $1 \leq \ell \leq B$;

Case 2: $B + 1 \leq \ell \leq 2B$;

Case 3: $2B + 1 \leq \ell \leq 3B$.

For Case 1 and Case 3, we can use Prony’s algorithm (see Algorithm 2.2) to recover $f(x)$ from a consecutive sequence of length $2B$: either $(\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_{2B})$ or $(\hat{a}_{B+1}, \hat{a}_{B+2}, \ldots, \hat{a}_{3B})$. To deal with Case 2, we replace the erroneous value $\hat{a}_\ell$ by a symbol $\alpha$. Then the determinant the Hankel matrix $H_{\ell-B}$ (see (8)) is univariate polynomial of degree $B + 1$ in $\alpha$. By Prony/Blahut/Ben-Or/Tiwari Theorem [Prony III (1795); Blahut 1983; Ben-Or and Tiwari 1988], $(f(\omega^k))_{k \geq 0}$ is a linearly generated sequence and its minimal generator has degree $\leq B$. Therefore $f(\omega^\ell)$ is a solution of the equation:

$$\text{det}(H_{\ell-B}) = 0. \quad (9)$$

By solving the equation (9), we obtain a list of candidates $\{\xi_1, \ldots, \xi_b\}$ for the correct value $f(\omega^\ell)$. For each candidate $\xi_k (1 \leq k \leq b)$, we substitute $\hat{a}_\ell$ by $\xi_k$ in the sequence $(\hat{a}_{B+1}, \hat{a}_{B+2}, \ldots, \hat{a}_{2B})$ and try Prony’s algorithm on the updated sequence $(\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_{2B})$, which gives us a list of sparse polynomials with $f(x)$ being contained. The process of correcting one error from $3B$ evaluations is illustrated by the following example.

**Example 2.1.** Assume that we are given $B = 3$. With $3B = 9$ evaluations $\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_9$ obtained from the black box for $f$ at inputs $\omega, \omega^2, \ldots, \omega^9$, we have the following $6 \times 4$ matrix:

$$H = \begin{bmatrix}
\hat{a}_1 & \hat{a}_2 & \hat{a}_3 & \hat{a}_4 \\
\hat{a}_2 & \hat{a}_3 & \hat{a}_4 & \hat{a}_5 \\
\hat{a}_3 & \hat{a}_4 & \hat{a}_5 & \hat{a}_6 \\
\hat{a}_4 & \hat{a}_5 & \hat{a}_6 & \hat{a}_7 \\
\hat{a}_5 & \hat{a}_6 & \hat{a}_7 & \hat{a}_8 \\
\hat{a}_6 & \hat{a}_7 & \hat{a}_8 & \hat{a}_9
\end{bmatrix} \in \mathbb{K}^{6\times4}$$

For $r = 1, 2, 3$, the matrices $H_r$ (see (8)) are $4 \times 4$ submatrices of $H$:

$$H_1 = \begin{bmatrix}
\hat{a}_1 & \hat{a}_2 & \hat{a}_3 & \hat{a}_4 \\
\hat{a}_2 & \hat{a}_3 & \hat{a}_4 & \hat{a}_5 \\
\hat{a}_3 & \hat{a}_4 & \hat{a}_5 & \hat{a}_6 \\
\hat{a}_4 & \hat{a}_5 & \hat{a}_6 & \hat{a}_7
\end{bmatrix}, \quad H_2 = \begin{bmatrix}
\hat{a}_2 & \hat{a}_3 & \hat{a}_4 & \hat{a}_5 \\
\hat{a}_3 & \hat{a}_4 & \hat{a}_5 & \hat{a}_6 \\
\hat{a}_4 & \hat{a}_5 & \hat{a}_6 & \hat{a}_7 \\
\hat{a}_5 & \hat{a}_6 & \hat{a}_7 & \hat{a}_8
\end{bmatrix}, \quad H_3 = \begin{bmatrix}
\hat{a}_3 & \hat{a}_4 & \hat{a}_5 & \hat{a}_6 \\
\hat{a}_4 & \hat{a}_5 & \hat{a}_6 & \hat{a}_7 \\
\hat{a}_5 & \hat{a}_6 & \hat{a}_7 & \hat{a}_8 \\
\hat{a}_6 & \hat{a}_7 & \hat{a}_8 & \hat{a}_9
\end{bmatrix}.$$ 

Suppose there is one error $\hat{a}_\ell \neq f(\omega^\ell)$ in these $3B$ evaluations. We recover $f(x)$ by the following steps.
1. Try to recover \( f(x) \) from \((\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_6)\) and \((\hat{a}_4, \hat{a}_5, \ldots, \hat{a}_9)\) by Prony’s algorithm; \( f(x) \) will be returned if \( \ell \in \{7, 8, 9\} \) or \( \ell \in \{1, 2, 3\} \).

2. For \( \ell \in \{4, 5, 6\} \), substitute \( \hat{a}_\ell \) by \( \alpha \), then \( \det(H_{\ell-3}) \) is a univariate polynomial of degree 4 in \( \alpha \) and \( f(\omega^\ell) \) is a root of \( \det(H_{\ell-3}) \). Compute the roots \( \{\xi_k\}_{k \geq 1} \) of \( \det(H_{\ell-3}) \). For each root \( \xi_k \), replace \( \hat{a}_\ell \) by \( \xi_k \) and check if the matrix \( H \) has rank \( \leq 3 \). If yes, then use Prony’s algorithm (see Algorithm 2.2) on the updated sequence \((\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_6)\). As \( f(\omega^\ell) \) is equal to some \( \xi_k \), this step will recover \( f(x) \) in case that \( \ell \in \{4, 5, 6\} \).

For computing the term degrees \( \delta_j \) of \( f \), we need an integer logarithm algorithm having the following input and output specifications.

**Algorithm 2.1. Integer Logarithm Algorithm**

**Input:**
- An upper bound \( D \in \mathbb{Z}_{\geq 0} \).
- \( \omega \in K \setminus \{0\} \) and has order \( \geq 2D + 1 \), meaning that \( \forall \eta \geq 1, \omega^\eta = 1 \Rightarrow \eta \geq 2D + 1 \).
- \( \rho \in K \setminus \{0\} \).

**Output:**
- Either \( \delta \in \mathbb{Z} \) with \( |\delta| \leq D \) and \( \omega^\delta = \rho \),
- or FAIL.

We describe the subroutine which we call Try Prony’s algorithm. This subroutine will be frequently used in our main algorithms.

**Algorithm 2.2. Try Prony’s algorithm**

**Input:**
- A position \( r \) and sequence \((\hat{a}_r, \ldots, \hat{a}_{r+2B-1})\) in \( K \) where \( K \) is a field of scalars.
- An upper bound \( D \in \mathbb{Z}_{\geq 0} \).
- \( \omega \in K \setminus \{0\} \) and has order \( \geq 2D + 1 \).
- An algorithm that computes all roots \( \in K \) of a polynomial \( \in K[x] \).
- Algorithm 2.1: Integer Logarithm Algorithm that takes \( D, \omega, \rho \) as input and outputs:
  - either \( \delta \in \mathbb{Z} \) with \( |\delta| \leq D \) and \( \omega^\delta = \rho \),
  - or FAIL.

**Output:**
- A sparse Laurent polynomial of sparsity \( t \leq B \) and has term degrees \( \delta_j \) with \( |\delta_j| \leq D \), or FAIL.

**Step 1:** Use Berlekamp/Massey algorithm to compute the minimal linear generator of the sequence \((\hat{a}_r, \ldots, \hat{a}_{r+2B-1})\) and denote it by \( \Lambda(z) \). If \( \Lambda(0) = 0 \) return FAIL.

**Step 2:** Compute all distinct roots \( \in K \) of \( \Lambda(z) \), denoted by \( \rho_1, \ldots, \rho_t \). If \( t < \deg(\Lambda) \) then return FAIL.

**Step 3:** For \( j = 1, \ldots, t \), use the Algorithm 2.1: Integer Logarithm Algorithm to compute \( \delta_j = \log_\omega \rho_j \). If the Integer Logarithm Algorithm returns FAIL, then return FAIL.

**Step 4:** Compute the coefficients \( c_1, \ldots, c_t \) by solving the following transposed generalized Vandermonde system

\[
\begin{bmatrix}
\rho_1^r & \rho_2^r & \cdots & \rho_t^r \\
\rho_1^{r+1} & \rho_2^{r+1} & \cdots & \rho_t^{r+1} \\
\vdots & \vdots & \ddots & \vdots \\
\rho_1^{r+t-1} & \rho_2^{r+t-1} & \cdots & \rho_t^{r+t-1}
\end{bmatrix}
\begin{bmatrix}
c_1 \\
c_2 \\
\vdots \\
c_t
\end{bmatrix}
= 
\begin{bmatrix}
\hat{a}_r \\
\hat{a}_{r+1} \\
\vdots \\
\hat{a}_{r+t-1}
\end{bmatrix}
\]
Step 5: Return the polynomial $\sum_{j=1}^{t} c_j x^{\delta_j}$.

Now we give an algorithm for interpolating a black-box polynomial with sparsity bounded by $B$. This algorithm can correct one error in $3B$ evaluations.

**Algorithm 2.3.** A list-interpolation algorithm for power-basis sparse polynomials with evaluations containing at most one error.

**Input:**
- A black box representation of a polynomial $f \in K[x, x^{-1}]$ where $K$ is a field of scalars.

The black box for $f$ returns the same (erroneous) output when probed multiple times at the same input.
- An upper bound $B$ on the sparsity of $f$.
- An upper bound $D \geq \max_j |\delta_j|$, where $\delta_j$ are term degrees of $f$.
- $\omega \in K \setminus \{0\}$ satisfying:
  - $\omega$ has order $\geq 2D + 1$;
  - $\omega^{i_1} \neq \omega^{i_2}$ for all $1 \leq i_1 < i_2 \leq 3B$.
- An algorithm that computes all roots $\in K$ of a polynomial $\in K[x]$.

**Output:**
- An empty list or a list of sparse polynomials $\{f^{[1]}, \ldots, f^{[M]}\}$ with each $f^{[k]}$ $(1 \leq k \leq M)$ satisfying:
  - $f^{[k]}$ has sparsity $\leq B$ and has term degrees $\delta_j$ with $|\delta_j| \leq D$;
  - $f^{[k]}$ is represented by its term degrees and coefficients;
  - there is at most one index $i \in \{1, 2, \ldots, 3B\}$ such that $f^{[k]}(\omega^i) \neq \hat{a}_i$, where $\hat{a}_i$ is the output of the black box probed at input $\omega^i$;
  - $f$ is contained in the list.

**Step 1:** For $i = 1, 2, \ldots, 3B$, get the output $\hat{a}_i$ of the black box for $f$ at input $\omega^i$. Let $L$ be an empty list.

**Step 2:** Use Algorithm 2.2 on the sequence $(\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_{2B})$. If the algorithm returns a sparse polynomial $\tilde{f}$ of sparsity $\leq B$ and has term degrees $\delta_j$ with $|\delta_j| \leq D$, and there is at most one index $i \in \{1, 2, \ldots, 3B\}$ such that $\tilde{f}(\omega^i) \neq \hat{a}_i$, then add $\tilde{f}$ to the list $L$.

If the error is in $(\hat{a}_{2B+1}, \hat{a}_{2B+2}, \ldots, \hat{a}_{3B})$, then the sequence $(\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_{2B})$ is free of errors, so Algorithm 2.2 in Step 2 will return $f$, and $f$ will be added to the list $L$.

**Step 3:** Use Algorithm 2.2 on the sequence $(\hat{a}_{B+1}, \hat{a}_{B+2}, \ldots, \hat{a}_{3B})$. If the algorithm returns a sparse polynomial $\tilde{f}$ of sparsity $\leq B$ and has term degrees $\delta_j$ with $|\delta_j| \leq D$, and there is at most one index $i \in \{1, 2, \ldots, 3B\}$ such that $\tilde{f}(\omega^i) \neq \hat{a}_i$, then add $\tilde{f}$ to the list $L$.

If the error is in $(\hat{a}_1, \ldots, \hat{a}_B)$, then the sequence $(\hat{a}_{B+1}, \hat{a}_{B+2}, \ldots, \hat{a}_{3B})$ is free of errors, so Algorithm 2.2 in Step 3 will return $f$, and $f$ will be added into the list $L$.

**Step 4:** For $\ell = B + 1, B + 2, \ldots, 2B$,

$4(a)$: substitute $\hat{a}_\ell$ by a symbol $\alpha$ in the matrix $H_{\ell-B}$ (see (8)); use the fraction free Berlekamp/Massey algorithm [Giesbrecht, Kaltofen, and Lee 2002; Kaltofen and Yuhasz 2013] to compute the determinant of $H_{\ell-B}$ and denote it by $\Delta_\ell(\alpha)$;
Here $\Delta_\ell(\alpha)$ is a univariate polynomial of the form $(-1)^{B+1}\alpha^{B+1} + \tilde{\Delta}_\ell(\alpha)$ with $\deg(\Delta_\ell(\alpha)) < B + 1$;

4(b): compute all solutions of the equation $\Delta_\ell(\alpha) = 0$ in $K$; denote the solution set as $\{\xi_1, \ldots, \xi_b\}$;

4(c): for $k = 1, \ldots, b$,

4(c)\(i\): substitute $\hat{a}_\ell$ by $\xi_k$;

4(c)\(ii\): use Berlekamp/Massey algorithm to compute the the minimal linear generator of the new sequence $(\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_{3B})$ and denote it by $\Lambda(z)$;

4(c)\(iii\): if $\deg(\Lambda(z)) \leq B$, repeat Step 2.

If $\hat{a}_\ell \neq f(\omega^\ell)$ with $\ell \in \{B + 1, B + 2, \ldots, 2B\}$, then we substitute $\hat{a}_\ell$ by a symbol $\alpha$ and compute the roots $\{\xi_1, \ldots, \xi_b\}$ of $\Delta_\ell(\alpha)$ in $K$. The correct value $f(\omega^\ell)$ is in the set $\{\xi_1, \ldots, \xi_b\}$. Thus for every root $\xi_k$ ($k = 1, \ldots, b$), we replace $\hat{a}_\ell$ with $\xi_k$ and use Berlekamp/Massey algorithm to check if the new sequence $(\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_{3B})$ is generated by some polynomial of degree $\leq B$. If so, then we apply Algorithm 2.2 on the updated sequence $(\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_{2B})$. In the end, Step 4 will add $f$ into the list $L$ in case that $B + 1 \leq \ell \leq 2B$.

Step 5: Return the list $L$.

**Proposition 2.1.** The output list of Algorithm 2.3 contains $\leq B^2 + B + 2$ polynomials.

*Proof.* The Step 2 in Algorithm 2.3 produces $\leq 1$ polynomial and so is Step 3. In the Step 4 of Algorithm 2.3, because $\Delta_\ell(\alpha)$ has degree $B + 1$, the equation $\Delta_\ell(\alpha) = 0$ has $\leq B + 1$ solutions in $K$, therefore this step produces $\leq B(B + 1)$ polynomials. Thus the output list of Algorithm 2.3 contains $\leq 2 + B(B + 1)$ polynomials. □

### 2.2. Correcting 2 Errors

In this section, we give a list-interpolation algorithm to recover $f(x)$ (see (7)) from $4B$ evaluations that contain 2 errors. Recall that $B$ is an upper bound on the sparsity of $f(x)$ and $D$ is an upper bound on the absolute values of the term degrees of $f(x)$. We will use Algorithm 2.3 as a subroutine.

Let $\omega \in K \setminus \{0\}$ such that: (1) $\omega$ has order $\geq 2D + 1$, and (2) $\omega^{i_1} \neq \omega^{i_2}$ for all $1 \leq i_1 < i_2 \leq 4B$. For $i = 1, 2, \ldots, 4B$, let $\hat{a}_i$ be the output of the black box probed at input $\omega^i$. Let $\hat{a}_{\ell_1}$ and $\hat{a}_{\ell_2}$ be the 2 errors and $\ell_1 < \ell_2$. The problem can be covered by the following four cases:

Case 1: $1 \leq \ell_1 \leq B$;

Case 2: $3B + 1 \leq \ell_2 \leq 4B$;

Case 3: $B + 1 \leq \ell_1 < \ell_2 \leq 2B$ or $2B + 1 \leq \ell_1 < \ell_2 \leq 3B$

Case 4: $B + 1 \leq \ell_1 \leq 2B$ and $2B + 1 \leq \ell_2 \leq 3B$. 

First, we try the Algorithm 2.3 on the sequences \((\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_{3B})\) and \((\hat{a}_{B+1}, \hat{a}_{B+2}, \ldots, \hat{a}_{4B})\), which can list interpolate \(f(x)\) if either Case 2 or Case 1 happens. Next, we use the Algorithm 2.2 on the sequences \((\hat{a}_1, \ldots, \hat{a}_{2B})\) and \((\hat{a}_{2B+1}, \ldots, \hat{a}_{4B})\), which will return \(f(x)\) if Case 3 happens. For Case 4, we substitute the two erroneous values \(\hat{a}_{\ell_1}\) and \(\hat{a}_{\ell_2}\) by two symbols \(\alpha_1\) and \(\alpha_2\) respectively. Then the pair of correct values \((f(\omega^{\ell_1}), f(\omega^{\ell_2}))\) is a solution of the following Pham system (see Lemma 2.2 and Lemma 2.3):

\[
\det(H_{\ell_1-B}) = 0, \quad \det(H_{\ell_2-B}) = 0,
\]

where \(H_{\ell_1-B}\) and \(H_{\ell_2-B}\) are Hankel matrices defined as (8). As the Pham systems (10) is zero-dimensional (see Lemma 2.3), we compute the solution set \(\{(\xi_{1,1}, \xi_{2,1}), \ldots, (\xi_{1,n}, \xi_{2,n})\}\) of (10). Then, for \(k = 1, \ldots, b\), we substitute \((\hat{a}_{\ell_1}, \hat{a}_{\ell_2})\) by \((\xi_{1,k}, \xi_{2,k})\) and apply Algorithm 2.2 on the updated sequence \((\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_{2B})\); this results in a list of candidates for \(f\) if Case 4 happens.

The following Lemma shows that the determinants arising in (10) have the Pham property, using diagonals in place of anti-diagonals.

**Lemma 2.2.** Let \(A\) be an \(n \times n\) matrix with the following properties:

1) for \(i = 1, \ldots, n\), \(A[i, i] = \alpha_1\);

2) for some fixed \(k \in \{1, \ldots, n-1\}\) and for \(i = 1, \ldots, n-k\), \(A[i, i+k] = \alpha_2\);

3) all other entries of \(A\) elements are in the field of scalars \(K\).

Then \(\det(A) = \alpha_1^n + Q(\alpha_1, \alpha_2)\) where \(Q(\alpha_1, \alpha_2)\) is a polynomial of total degree \(\leq n - 1\).

**Proof.** The matrix \(A\) is of the form:

\[
A = \begin{bmatrix}
\alpha_1 & \alpha_2 & \ast \\
\cdot & \cdot & \cdot \\
\cdot & \cdot & \alpha_2 \\
\ast & \cdot & \cdot \\
\alpha_1
\end{bmatrix}.
\]

We prove by induction on \(n\). It is trivial if \(n = 1\). Assume that the conclusion holds for \(n - 1\). By minor expansion on the first column of \(A\), we have

\[
\det(A) = \alpha_1(\alpha_1^{n-1} + Q_1(\alpha_1, \alpha_2)) + Q_2(\alpha_1, \alpha_2)
\]

where \(Q_2(\alpha_1, \alpha_2)\) has total degree \(\leq n - 1\). By induction hypothesis, \(Q_1(\alpha_1, \alpha_2)\) has total degree \(\leq n - 2\). Let \(Q = \alpha_1 \cdot Q_1 + Q_2\). The proof is complete. \(\square\)

**Lemma 2.3.** The Pham system

\[
\begin{align*}
\alpha_1^{n_1} + Q_1(\alpha_1, \alpha_2) &= 0, \quad \text{deg}(Q_1) \leq n_1 - 1 \\
\alpha_2^{n_2} + Q_2(\alpha_1, \alpha_2) &= 0, \quad \text{deg}(Q_2) \leq n_2 - 1
\end{align*}
\]

has at most \(n_1n_2\) solutions, where \(Q_1\) and \(Q_2\) are two polynomials in \(K[\alpha_1, \alpha_2]\).
Proof. See e.g. [Cox, Little, and O’Shea 2015, Chapter 5, Section 3, Theorem 6]. □

Example 2.2. Let $B = 3$. With $4B = 12$ evaluations $\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_{12}$ obtained from the black box for $f$ at inputs $\omega, \omega^2, \ldots, \omega^{12}$, we have the following $9 \times 4$ matrix:

$$H = \begin{bmatrix} \hat{a}_1 & \hat{a}_2 & \hat{a}_3 & \hat{a}_4 \\ \hat{a}_2 & \hat{a}_3 & \hat{a}_4 & \hat{a}_5 \\ \hat{a}_3 & \hat{a}_4 & \hat{a}_5 & \hat{a}_6 \\ \hat{a}_4 & \hat{a}_5 & \hat{a}_6 & \hat{a}_7 \\ \hat{a}_5 & \hat{a}_6 & \hat{a}_7 & \hat{a}_8 \\ \hat{a}_6 & \hat{a}_7 & \hat{a}_8 & \hat{a}_9 \\ \hat{a}_7 & \hat{a}_8 & \hat{a}_9 & \hat{a}_{10} \\ \hat{a}_8 & \hat{a}_9 & \hat{a}_{10} & \hat{a}_{11} \\ \hat{a}_9 & \hat{a}_{10} & \hat{a}_{11} & \hat{a}_{12} \end{bmatrix} \in K^{9 \times 4}$$

Suppose there are two errors $\hat{e}_{\ell_1}, \hat{e}_{\ell_2} (\ell_1 < \ell_2)$ in the evaluations. If $\ell_1 \in \{1, 2, 3\}$, then the Algorithm 2.3 can recover $f(x)$ from the last $3B$ evaluations $(\hat{a}_4, \hat{a}_5, \ldots, \hat{a}_{12})$. Similarly, $f(x)$ can also be recovered from $(\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_9)$ by the Algorithm 2.3 if $\ell_2 \in \{10, 11, 12\}$. Next, if $\ell_1, \ell_2 \in \{4, 5, 6\}$ or $\ell_1, \ell_2 \in \{7, 8, 9\}$, then the Algorithm 2.2 can recover $f(x)$ from $(\hat{a}_7, \ldots, \hat{a}_{12})$ or $(\hat{a}_1, \ldots, \hat{a}_6)$.

It is remained to consider the case that $\ell_1 \in \{4, 5, 6\}$ and $\ell_2 \in \{7, 8, 9\}$. We substitute $\hat{e}_{\ell_1}, \hat{e}_{\ell_2}$ by $\alpha_1, \alpha_2$ respectively. Then the determinants of the matrices $H_{\ell_1-3}$ and $H_{\ell_2-3}$ can be written as:

$$\text{det}(H_{\ell_1-3}) = -\alpha_1^4 + Q_1(\alpha_1, \alpha_2), \text{ deg } Q_1 \leq 3$$
$$\text{det}(H_{\ell_2-3}) = -\alpha_2^4 + Q_2(\alpha_1, \alpha_2), \text{ deg } Q_2 \leq 3$$

(12)

where $H_{\ell_1-3}, H_{\ell_2-3}$ are Hankel matrices defined as (8) and where $Q_1$ and $Q_2$ are bivariate polynomials in $\alpha_1$ and $\alpha_2$. We compute the roots $(\xi_{1,k}, \xi_{2,k})_{k \geq 1}$ of the system (12) in $K$ and the pair correct values $(f(\omega^{\xi_{1,k}}), f(\omega^{\xi_{2,k}}))$ is one of the roots. For each root $(\xi_{1,k}, \xi_{2,k})$, we substitute $\hat{e}_{\ell_1}, \hat{e}_{\ell_2}$ by $\xi_{1,k}, \xi_{2,k}$ respectively, and check if the matrix $H$ has rank $B = 3$. If so, then run Algorithm 2.2 on the updated sequence $(\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_6)$. In the end, we obtain a list of sparse polynomials that contains $f(x)$.

Algorithm 2.4. A list-interpolation algorithm for power-basis sparse polynomial with evaluations containing at most 2 errors.

Input: • A black box representation of a polynomial $f \in K[x, x^{-1}]$ where $K$ is a field of scalars. The black box for $f$ returns the same (erroneous) output when probed multiple times at the same input.
• An upper bound $B$ on the sparsity of $f$.
• An upper bound $D \geq \max_j |\delta_j|$, where $\delta_j$ are term degrees of $f$.
• $\omega \in K \setminus \{0\}$ satisfying:
  • $\omega$ has order $\geq 2D + 1$;
  • $\omega^{i_1} \neq \omega^{i_2}$ for all $1 \leq i_1 < i_2 \leq 4B$.
• An algorithm to compute all roots $\in K$ of polynomials in $K[x]$.
Output: ▶ An empty list or a list of sparse polynomials \{f^{[1]}, \ldots, f^{[M]}\} with each \(f^{[k]}\) (1 \(\leq k \leq M\) satisfying:
  ▶ \(f^{[k]}\) has sparsity \(\leq B\) and has term degrees \(\delta_j\) with \(|\delta_j| \leq D\),
  ▶ \(f^{[k]}\) is represented by its term degrees and coefficients;
  ▶ there are \(\leq 2\) indices \(i_1, i_2 \in \{1, 2, \ldots, 4B\}\) such that \(f^{[k]}(\omega^{i_1}) \neq \hat{a}_{i_1}\) and \(f^{[k]}(\omega^{i_2}) \neq \hat{a}_{i_2}\) where \(\hat{a}_{i_1}\) and \(\hat{a}_{i_2}\) are the outputs of the black box probed at inputs \(\omega^{i_1}\) and \(\omega^{i_2}\) respectively;
  ▶ \(f\) is contained in the list.

Step 1: For \(i = 1, 2, \ldots, 4B\), get the output \(\hat{a}_i\) of the black box for \(f\) at input \(\omega^i\).

Step 2: Take \((\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_{3B})\) and \((\hat{a}_{B+1}, \hat{a}_{B+2}, \ldots, \hat{a}_{4B})\) as the evaluations at the first step of Algorithm 2.3 and get two lists \(L_1, L_2\). Let \(L\) be the union of \(L_1\) and \(L_2\).

If either \((\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_{3B})\) or \((\hat{a}_{B+1}, \hat{a}_{B+2}, \ldots, \hat{a}_{4B})\) contains \(\leq 1\) error, the Algorithm 2.3 can compute a list of sparse polynomials containing \(f(x)\).

Step 3: Use Algorithm 2.2 on the sequences \((\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_{2B})\) and \((\hat{a}_{2B+1}, \hat{a}_{2B+2}, \hat{a}_{4B})\). If Algorithm 2.2 returns a sparse polynomial \(\bar{f}\) of sparsity \(\leq B\) and has term degrees \(\delta_j\) with \(|\delta_j| \leq D\), then add \(\bar{f}\) into the list \(L\).

If either \((\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_{2B})\) or \((\hat{a}_{2B+1}, \hat{a}_{2B+2}, \hat{a}_{4B})\) is error-free, the Algorithm 2.2 will return \(f(x)\).

Step 4: For every polynomial \(\bar{f}\) in the list \(L\), if there are \(\geq 3\) indices \(i \in \{1, 2, \ldots, 4B\}\) such that \(\bar{f}(\omega^i) \neq \hat{a}_i\) then delete \(\bar{f}\) from \(L\).

Step 5: For \(\ell_1 = B + 1, \ldots, 2B\) and \(\ell_2 = 2B + 1, \ldots, 3B\),

5(a): substitute \(\hat{a}_{\ell_1}\) by \(\alpha_1\) and \(\hat{a}_{\ell_2}\) by \(\alpha_2\) in the Hankel matrices \(H_{\ell_1-B}\) and \(H_{\ell_2-B}\) (see (8)); let \(\Delta_{\ell_1}(\alpha_1, \alpha_2) = \det(H_{\ell_1-B})\) and \(\Delta_{\ell_2}(\alpha_1, \alpha_2) = \det(H_{\ell_2-B})\).

Here, we also use the fraction free Berlekamp/Massey algorithm [Giesbrecht, Kaltofen, and Lee 2002; Kaltofen and Yuhasz 2013] to compute the determinants of \(H_{\ell_1-B}\) and \(H_{\ell_2-B}\).

5(b): compute all solutions of the Pham system \(\{\Delta_{\ell_1}(\alpha_1, \alpha_2) = 0, \Delta_{\ell_2}(\alpha_1, \alpha_2) = 0\}\) in \(K^2\); denote the solution set as \(\{(\xi_{1,1}, \xi_{2,1}), \ldots, (\xi_{1,b}, \xi_{2,b})\}\).

One may use a Sylvester resultant algorithm and the root finder in \(K[x]\) to accomplish this task in polynomial time.

5(c): for \(k = 1, \ldots, b\),

5(c)i: substitute \(\hat{a}_{\ell_1}\) by \(\xi_{1,k}\) and \(\hat{a}_{\ell_2}\) by \(\xi_{2,k}\);
5(c)ii: use Berlekamp/Massey algorithm to compute the minimal linear generator of the new sequence \((\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_{4B})\) and denote it by \(\Lambda(z)\);
5(c)iii: if \(\deg(\Lambda(z)) \leq B\), use Algorithm 2.2 on the updated sequence \((\hat{a}_1, \hat{a}_2, \ldots, \hat{a}_{2B})\); if Algorithm 2.2 returns a sparse polynomial \(\bar{f}\) of sparsity \(\leq B\) and has term degrees \(\delta_j\) with \(|\delta_j| \leq D\), and there are \(\leq 2\) indices \(i_1, i_2 \in \{1, 2, \ldots, 4B\}\) such that \(\bar{f}(\omega^{i_1}) \neq \hat{a}_{i_1}\) and \(\bar{f}(\omega^{i_2}) \neq \hat{a}_{i_2}\), then add \(\bar{f}\) into the list \(L\);
If the two errors are \( \hat{a}_{\ell_1} \) and \( \hat{a}_{\ell_2} \) with \( \ell_1 \in \{B+1, \ldots, 2B\} \) and \( \ell_2 \in \{2B+1, \ldots, 3B\} \), we substitute \( \hat{a}_{\ell_1} \) and \( \hat{a}_{\ell_2} \) by two symbols \( \alpha_1 \) and \( \alpha_2 \) respectively. As the pair of correct values \( (f(\omega^{\ell_1}), f(\omega^{\ell_2})) \) is a solution of the system \( \{\Delta_{\ell_1}(\alpha_1, \alpha_2) = 0, \Delta_{\ell_2}(\alpha_1, \alpha_2) = 0\} \), Step 5 will add \( f \) into the list \( L \).

Step 6: Return the list \( L \).

**Proposition 2.4.** The output list of Algorithm 2.4 contains \( \leq B^4 + 2B^3 + 3B^2 + 2B + 6 \) polynomials.

**Proof.** In Algorithm 2.4, only Step 2, Step 3, and Step 5 produce new polynomials. By Proposition 2.1, both the lists \( L_1 \) and \( L_2 \) obtained at Step 2 contain \( \leq B^2 + B + 2 \) polynomials. Step 3 produces \( \leq 2 \) polynomials. For Step 5 of Algorithm 2.4, the Pham system \( \{\Delta_{\ell_1}(\alpha, \beta) = 0, \Delta_{\ell_2}(\alpha, \beta) = 0\} \) has \( \leq (B+1)^2 \) solutions, so this step produces \( \leq B^2(3B+1)^2 \) polynomials. Therefore the output list contains \( \leq B^2(3B+1)^2 + 2(B^2 + B + 2) + 2 \) polynomials. \( \square \)

### 2.3. Correcting \( E \) Errors

Recall that \( f(x) \) is a sparse univariate polynomial of the form \( \sum_{j=1}^{t} c_j x^{\delta_j} \) (see (7)) with \( t \leq B \) and \( \forall j, |\delta_j| \leq D \). We show how to list interpolate \( f(x) \) from \( N \) evaluations containing \( \leq E \) errors, where

\[
N = \left\lfloor \frac{4}{3}E + 2 \right\rfloor B.
\]  

Let \( \theta = \lfloor E/3 \rfloor \). Choose \( \omega_1, \ldots, \omega_{\theta+1}, \omega_{\theta+1} \in K \setminus \{0\} \) such that:

1. \( \omega_\sigma \) has order \( \geq 2D + 1 \) for all \( 1 \leq \sigma \leq \theta + 1 \), and
2. \( \omega_{\sigma_1}^{\delta_1} \neq \omega_{\sigma_2}^{\delta_2} \) for any \( 1 \leq \sigma_1 < \sigma_2 \leq \theta + 1 \) and \( 1 \leq i_1 < i_2 \leq 4B \).

Let \( \hat{a}_{\sigma, i} \) denote the output of the black box at input \( \omega_{\sigma}^i \) for \( \sigma = 1, \ldots, \theta + 1 \) and \( i = 1, \ldots, 4B \).

If \( E \mod 3 = 0 \) then \( N = (E/3)4B + 2B \). The problem is reduced to one the following situations: (1) the last block \( (\hat{a}_{\theta+1,1}, \hat{a}_{\theta+1,2}, \ldots, \hat{a}_{\theta+1,2B}) \) of length \( 2B \) is free of error, or (2) there is some block \( (\hat{a}_{\sigma,1}, \hat{a}_{\sigma,2}, \ldots, \hat{a}_{\sigma,4B}) \) with \( 1 \leq \sigma \leq \theta \) which contains \( \leq 2 \) errors. These two situations can be respectively dealt with the Algorithm 2.2 and Algorithm 2.4.

If \( E \mod 3 = 1 \) then \( N = 4B\theta + 3B \). The problem is reduced to one the following situations: (1) the last block \( (\hat{a}_{\theta+1,1}, \hat{a}_{\theta+1,2}, \ldots, \hat{a}_{\theta+1,3B}) \) of length \( 3B \) has \( \leq 1 \) error, or (2) there is some block \( (\hat{a}_{\sigma,1}, \hat{a}_{\sigma,2}, \ldots, \hat{a}_{\sigma,4B}) \) with \( 1 \leq \sigma \leq \theta \) which contains \( \leq 2 \) errors. Therefore by applying the Algorithm 2.3 on \( (\hat{a}_{\theta+1,1}, \hat{a}_{\theta+1,2}, \ldots, \hat{a}_{\theta+1,3B}) \) and the Algorithm 2.4 on \( (\hat{a}_{\sigma,1}, \hat{a}_{\sigma,2}, \ldots, \hat{a}_{\sigma,4B}) \), we can list interpolate \( f(x) \).

If \( E \mod 3 = 2 \) then \( E = 3\theta + 2 \) and \( N = (\theta + 1)4B \). So there is some \( \sigma \in \{1, \ldots, \theta + 1\} \) such that the block \( (\hat{a}_{\sigma,1}, \hat{a}_{\sigma,2}, \ldots, \hat{a}_{\sigma,4B}) \) of length \( 4B \) contains \( \leq 2 \) errors, and we can use the Algorithm 2.4 on this block to list interpolate \( f(x) \).

**Remark 2.1.** We apply the Algorithm 2.4 on every block \( (\hat{a}_{\sigma,1}, \hat{a}_{\sigma,2}, \ldots, \hat{a}_{\sigma,4B}) \) for all \( \sigma \in \{1, \ldots, \lfloor E/3 \rfloor\} \), which will result in \( \leq \lfloor E/3 \rfloor (B^4 + 2B^3 + 3B^2 + 2B + 6) \) polynomials according to Proposition 2.4. The length of the last block depends on the value of \( E \), and we have the following different upper bounds on the number of resulting polynomials:
(1) \((E/3)(B^4 + 2B^3 + 3B^2 + 2B + 6) + 1\), if \(E \mod 3 = 0\);

(2) \([E/3] (B^4 + 2B^3 + 3B^2 + 2B + 6) + B^2 + B + 2\), if \(E \mod 3 = 1\) (see Proposition 2.1);

(3) \((\lfloor E/3 \rfloor + 1) (B^4 + 2B^3 + 3B^2 + 2B + 6)\), if \(E \mod 3 = 2\).

By Descartes’ rule of signs (see e.g. [Bochnak, Coste, and Roy 1998, Proposition 1.2.14]), the approach for correcting \(E\) errors will produce a single polynomial if \(K = \mathbb{R}, N \geq 2B + 2E\) and \(\omega_\sigma > 0, \forall \sigma\). However, if \(N < 2B + 2E\) then there can be \(\geq 2\) valid sparse interpolants. We give an example to illustrate this.

**Example 2.3.** Choose \(\omega > 0\). Let \(B\) be an upper bound on the sparsity of \(f\) and \(E\) be an upper bound on the number of errors in the evaluations. Let

\[
h = \prod_{i=0}^{2B-2} (x - \omega^i),
\]

and \(f^{[1]}\) be the sum of odd degree terms of \(h\) and \(f^{[2]}\) be the negative of the sum of even degree terms of \(h\). Clearly, we have \(h = f^{[1]} - f^{[2]}\) and \(f^{[1]}(\omega^i) = f^{[2]}(\omega^i)\) for \(i = 0, 1, \ldots, 2B - 2\). Moreover, both \(f^{[1]}\) and \(f^{[2]}\) have sparsity \(\leq B\) as \(\deg(h) = 2B - 1\). Consider a sequence \(a\) consisting of the following \(2B + 2E - 1\) values:

\[
a^{(1)} = (f^{[1]}(\omega^0), f^{[1]}(\omega^1), \ldots, f^{[1]}(\omega^{2B-2})),
\]

\[
a^{(2)} = (f^{[1]}(\omega^{2B-1}), f^{[1]}(\omega^{2B}), \ldots, f^{[1]}(\omega^{2B+E-2})),
\]

\[
a^{(3)} = (f^{[2]}(\omega^{2B+E-1}), f^{[2]}(\omega^{2B+E}), \ldots, f^{[2]}(\omega^{2B+2E-2})),
\]

that is, \(\hat{a} = (a^{(1)}, a^{(2)}, a^{(3)})\). If all the errors are in \(a^{(3)}\) then \(f^{[1]}\) is a valid interpolant. Alternatively, if all the errors are in \(a^{(2)}\) then \(f^{[2]}\) is a valid interpolant. Therefore, from these \(2B + 2E - 1\) values, we have at least \(2\) valid interpolants.

We remark that one of the valid interpolants, \(f^{[1]}\) and \(f^{[2]}\), must have \(B\) terms since otherwise uniqueness is guaranteed by Descartes’s rule of signs. In this example, both \(f^{[1]}\) and \(f^{[2]}\) have \(B\) terms because the polynomial \(h\) has \(2B\) terms. Indeed, \(\deg(h) = 2B - 1\) implies that \(h\) has \(\leq 2B\) terms, and by Descartes’ rule of signs, \(h\) has \(\geq 2B\) terms because it has \(2B - 1\) positive real roots. Therefore \(h\) is a dense polynomial. However, with the following substitutions

\[x = y^k, \omega = \bar{\omega}^k\text{ for some }k \gg 1,\]

we have again a counter example where \(h, f^{[1]}, \text{ and } f^{[2]}\) are sparse with respect to the new variable \(y\).
3. Sparse Interpolation in Chebyshev Basis with Error Correction

3.1. Correcting One Error

Let \( K \) be a field of scalars with characteristic \( \neq 2 \). Let \( f(x) \in K[x] \) be a polynomial represented by a black box. Assume that \( f(x) \) is a sparse polynomial in Chebyshev-1 basis of the form:

\[
f(x) = \sum_{j=1}^{t} c_j T_{\delta_j}(x) \in K[x], \quad 0 \leq \delta_1 < \delta_2 < \cdots < \delta_t = \deg(f), \forall j, 1 \leq j \leq t: c_j \neq 0,
\]

where \( T_{\delta_j}(x) \) are Chebyshev polynomials of the First kind of degree \( \delta_j \). We want to recover the term degrees and coefficients of the polynomial \( g \). Assume that \( c_j \) are correct, then \( f \) can be represented as a sparse Laurent polynomial:

\[
g(y) = f\left(\frac{y + y^{-1}}{2}\right) = \sum_{j=1}^{t} \frac{c_j}{2} (y^{\delta_j} + y^{-\delta_j})
\]  

Therefore the problem is reduced to recover the term degrees and coefficients of the polynomial \( g(y) \). Let \( \omega \in \mathbb{K} \) such that \( \omega \) has order \( \geq 4D + 1 \).

For \( i = 1, 2, \ldots, 3B \), let \( \hat{a}_{2i-1} \) be the output of the black box probed at input \( \gamma_{2i-1} = (\omega^{2i-1} + \omega^{-(2i-1)})/2 \). Note that \( g(\omega^i) = g(\omega^{-i}) \) for any integer \( i \). For odd integers \( r \in \{2k-1 \mid k = 1, \ldots, B\} \), let \( G_r \in K^{(B+1) \times (B+1)} \) be the following Hankel-Toeplitz matrix:

\[
G_r = \begin{bmatrix}
\hat{a}_{|r+2(i+j)|}^{B}_{i,j=0} & \hat{a}_{|r+2(i-j)|}^{B}_{i,j=0}
\end{bmatrix}_{i,j=0}^{B}.
\]  

If all the values involved in the matrix \( G_r \) are correct, then \( \det(G_r) = 0 \) [Arnold and Kaltofen 2015, Lemma 3.1].

If the \( 2B \) evaluations \( \{\hat{a}_{2i-1}\}_{i=1}^{2B} \) are free of errors, then one can use Prony’s algorithm to recover \( g(y) \) (and \( f(x) \)) from the following sequence [Kaltofen and Pernet 2014, Lemma 1]:

\[
\hat{a}_{-2(2B-1)-1}, \hat{a}_{-2(2B-2)-1}, \ldots, \hat{a}_{-1}, \hat{a}_1, \ldots, \hat{a}_{2(2B-1)-1}, \hat{a}_{2(2B)-1}.
\]  

Now we show how to list interpolate \( f(x) \) from \( 3B \) evaluations \( \{\hat{a}_{2i-1}\}_{i=1}^{3B} \) containing \( \leq 1 \) error.

Assume that \( \hat{a}_{2\ell-1} \) is the error, that is, \( \hat{a}_{2\ell-1} \neq f(\gamma_{2\ell-1}) = g(\omega^{2\ell-1}) \). The problem can be reduced to three cases:

Case 1: \( 1 \leq \ell \leq B \);

Case 2: \( B + 1 \leq \ell \leq 2B \);

Case 3: \( 2B + 1 \leq \ell \leq 3B \).
For Case 3, we can recover $f(x)$ from the sequence $(\hat{a}_{2i-1})_{i=-(2B-1)}^{2B}$. For the Case 1 and Case 2, we substitute $\hat{a}_{2\ell-1}$ by a symbol $\alpha$. Let

$$
\Delta_{2\ell-1}(\alpha) = \begin{cases} 
\det(G_{2\ell-1}), & \text{if } 1 \leq \ell \leq B, \\
\det(G_{2(\ell-B)-1}), & \text{if } B + 1 \leq \ell \leq 2B,
\end{cases}
$$

where $G_{2\ell-1}$ and $G_{2(\ell-B)-1}$ are defined as in (16) and $\Delta_{2\ell-1}(\alpha)$ is a univariate polynomial of degree $B + 1$ in $\alpha$ (see Lemma 3.1). By [Arnold and Kaltofen 2015, Lemma 3.1], the correct value $f(\gamma_{2\ell-1})$ is a solution of the equation $\Delta_{2\ell-1}(\alpha) = 0$. So we compute all solutions $\{\xi_1, \ldots, \xi_b\}$ of $\Delta_{2\ell-1}(\alpha) = 0$ in $K$. For each solution $\xi_k(1 \leq k \leq b)$ we replace $\hat{a}_{2\ell-1}$ by $\xi_k$ and try Prony’s algorithm on the updated sequence $(\hat{a}_{2i-1})_{i=-(2B-1)}^{2B}$. In the end, we will get a list of polynomials with $f(x)$ being contained.

**Lemma 3.1.** Let $r \in \{2k - 1 \mid k = 1, \ldots, B\}$ and $G_r = [\hat{a}_{|r+2(i+j)|} + \hat{a}_{|r+2(i-j)|}]_{i,j=0}^B$. If $\hat{a}_r$ or $\hat{a}_{r+2B}$ is substituted by a symbol $\alpha$ in $G_r$, then the determinant of $G_r$ is a univariate polynomial of degree $B + 1$ in $\alpha$.

**Proof.** First, we show that if $\hat{a}_{r+2B}$ is substituted by $\alpha$, then the matrix $G_r$ has the form:

$$
\begin{bmatrix}
* & \alpha+ \ast \\
\ast & \ast \\
\alpha+ & \ast \\
\ast & \ast
\end{bmatrix}
$$

Since $r \in \{2k - 1 \mid k = 0, \ldots, B\}$ and $i, j \in \{0, 1, \ldots, B\}$, we have

$$
|r + 2(i + j)| = r + 2B \Rightarrow i + j = B,
|r + 2(i - j)| = r + 2B \Rightarrow i = B, j = 0 \text{ or } i = 0, j = B.
$$

Therefore, either $|r + 2(i + j)| = r + 2B$ or $|r + 2(i - j)| = r + 2B$ implies $i + j = B$, so $\hat{a}_{r+2B}$ only appears on the anti-diagonal of the matrix $G_r$. Conversely, every element on the anti-diagonal of $G_r$ is equal to $\hat{a}_{r+2B} + \hat{a}_{r+2(i-j)}$, for some $i, j \in \{0, 1, \ldots, B\}$. Thus $G_r$ has the form (18) and its determinant is a univariate polynomial of degree $B + 1$ in $\alpha$.

Now we consider the case that $\hat{a}_r$ is substituted by $\alpha$. Similarly, because $r \in \{2k - 1 \mid k = 1, \ldots, B\}$ and $i, j \in \{0, 1, \ldots, B\}$, we have

$$
|r + 2(i + j)| = r \Rightarrow i = j = 0,
|r + 2(i - j)| = r \Rightarrow i = j \text{ or } i = j - r \text{ if } j \geq r.
$$

Therefore, if $r > B$ then $i = j$ in (19), so $\hat{a}_r$ only appears on the main diagonal of $G_r$. On the other hand, every element on the main diagonal of $G_r$ is equal to $\hat{a}_{r+2(i+i)} + \hat{a}_r$ for some $i \in \{0, 1, \ldots, t\}$. Hence, if $r > B$ then the determinant of $G_r$ is a polynomial of degree $B + 1$ in $\alpha$. Assume that $r \leq B$. From (19), we see that after substituting $\hat{a}_r$ by $\alpha$, the matrix $G_r$
has the form:

$$\begin{bmatrix}
\alpha + \ast & \cdots & \alpha + \ast & \ast \\
\vdots & \ddots & \vdots & \vdots \\
\ast & \cdots & \alpha + \ast & \ast \\
\ast & \cdots & \ast & \alpha + \ast \\
\end{bmatrix}.$$  (20)

According to Lemma 2.2, the determinant of the matrix (20) is a univariate polynomial of degree $B + 1$ in $\alpha$. □

**Example 3.1.** For $B = 3$, we have $3B = 9$ evaluations $\{\hat{a}_{2\ell-1}\}_{i=1}^{3B}$ obtained from the black box for $f$ at inputs $\gamma_i = (\omega^{2i-1} + \omega^{-(2i-1)})/2$. We construct the following $6 \times 4$ matrix:

$$G = \begin{bmatrix}
2\hat{a}_1 & \hat{a}_3 + \hat{a}_1 & \hat{a}_5 + \hat{a}_3 & \hat{a}_7 + \hat{a}_5 \\
2\hat{a}_3 & \hat{a}_5 + \hat{a}_1 & \hat{a}_7 + \hat{a}_1 & \hat{a}_9 + \hat{a}_3 \\
2\hat{a}_5 & \hat{a}_7 + \hat{a}_3 & \hat{a}_9 + \hat{a}_1 & \hat{a}_{11} + \hat{a}_1 \\
2\hat{a}_7 & \hat{a}_9 + \hat{a}_5 & \hat{a}_{11} + \hat{a}_3 & \hat{a}_{13} + \hat{a}_1 \\
2\hat{a}_9 & \hat{a}_{11} + \hat{a}_7 & \hat{a}_{13} + \hat{a}_5 & \hat{a}_{15} + \hat{a}_3 \\
2\hat{a}_{11} & \hat{a}_{13} + \hat{a}_9 & \hat{a}_{15} + \hat{a}_7 & \hat{a}_{17} + \hat{a}_5 \\
\end{bmatrix} \in \mathbb{K}^{6 \times 4}.
$$

For $r = 1, 3, 5$, the matrices $G_r$ are $4 \times 4$ submatrices of the matrix $G$. The matrix $G_1$ consists of the first 4 rows of $G$. If we substitute $\hat{a}_1$ or $\hat{a}_7$ by a symbol $\alpha$, then the determinant of $G_1$ is univariate polynomial of degree 4 in $\alpha$. The matrix $G_3$ consists of the second to the fifth row of $G$ and the determinant of $G_3$ becomes a univariate polynomial of degree 4 in $\alpha$ if $\hat{a}_3$ or $\hat{a}_9$ is substituted by $\alpha$. Similarly, the matrix $G_5$ consists of the last 4 rows of $G$. Substituting $\hat{a}_5$ or $\hat{a}_{11}$ by $\alpha$, $\det(G_5)$ is a univariate polynomial of degree 4 in $\alpha$.

Suppose there is one error $\hat{a}_{2\ell-1} \neq f(\gamma_{2\ell-1})$ in the $3B$ evaluations. Here is how we correct this single error for all possible $\ell$’s:

1. if $\ell \in \{1, 2, 3\}$, then substitute $\hat{a}_{2\ell-1}$ by $\alpha$ and compute the roots of $\det(G_{2\ell-1})$, and the roots are candidates for $f(\gamma_{2\ell-1})$;
2. if $\ell \in \{4, 5, 6\}$, then substitute $\hat{a}_{2\ell-1}$ by $\alpha$ and compute the roots of $\det(G_{2(\ell-3)-1})$, and the roots are candidates for $f(\gamma_{2\ell-1})$;
3. if $\ell \in \{7, 8, 9\}$, then $f(x)$ can be recovered by applying Prony’s algorithm on the sequence $(\hat{a}_{2\ell-1})_{i=5}^{6}$.

**Algorithm 3.1.** A list-interpolation algorithm for Chebyshev-1 sparse polynomials with evaluations containing at most one error.

*Input:*  
- A black box representation of a polynomial $f \in \mathbb{K}[x]$ where $\mathbb{K}$ is a field of scalars with characteristic $\neq 2$ and $f$ is a linear combination of Chebyshev-1 polynomials. The black box for $f$ returns the same (erroneous) output when probed multiple times at the same input.
- An upper bound $B$ of the sparsity of $f$.
• An upper bound $D$ of the degree of $f$.
• $\omega \in K \setminus \{0\}$ has order $\geq 4D + 1$.
• An algorithm that computes all roots $\in K$ of a polynomial $\in K[x]$.

Output: • An empty list or a list of sparse polynomials $\{f^{[1]}, \ldots, f^{[M]}\}$ with each $f^{[k]}$ $(1 \leq k \leq M)$ satisfying:
  • $f^{[k]}$ has sparsity $\leq B$ and degree $\leq D$;
  • $f^{[k]}$ is represented by its Chebyshev-1 term degrees and coefficients;
  • there is $\leq 1$ index $i \in \{1, 2, \ldots, 3B\}$ such that $f^{[k]}(\gamma_{2i-1}) \neq \hat{a}_{2i-1}$ where
    $\gamma_i = (\omega^{2i-1} + \omega^{-(2i-1)})/2$ and
    $\hat{a}_{2i-1}$ is the output of the black box probed at input $\gamma_{2i-1}$;
  • $f$ is contained in the list.

Step 1: For $i = 1, 2, \ldots, 3B$, get the output $\hat{a}_i$ of the black box for $f$ at input $\gamma_i = (\omega^{2i-1} + \omega^{-(2i-1)})/2$. Let $L$ be an empty list.

Step 2: Use Algorithm 2.2 on the sequence $(\hat{a}_{2i-1})_{i=1}^{3B}$. If Algorithm 2.2 returns a polynomial of the following form: $\sum_{j=1}^{t} c_j/2 (\omega^{-\delta_j x^{2i\Delta}} + \omega^{\delta_j x^{-2i\Delta}})$ with $c_j \in K$, $t \leq B$, $\delta_j \leq D$, then let $\hat{f} = \sum_{j=1}^{t} c_j T_{\delta_j}(x)$. If there is $\leq 1$ index $i \in \{1, \ldots, 3B\}$ such that $\hat{f}(\gamma_{2i-1}) \neq \hat{a}_{2i-1}$, then add $\hat{f}$ to the list $L$.

Step 2 will add $f$ to the list $L$ if the error is in $\{\hat{a}_{2i-1}\}_{i=2B+1}^{3B}$.

Step 3: For $\ell = 1, \ldots, B$,

3(a): substitute $\hat{a}_{2\ell-1}$ by a symbol $\alpha$ in the matrix $G_{2\ell-1}$; compute the determinant of $G_{2\ell-1}$ and denote it by $\Delta_{2\ell-1}(\alpha)$.

According to Lemma 3.1, $\Delta_{2\ell-1}(\alpha)$ is a univariate polynomial of degree $B + 1$ in $\alpha$.

3(b): compute all solutions of the equation $\Delta_{2\ell-1}(\alpha) = 0$ in $K$; denote the solution set as $\{\xi_1, \ldots, \xi_b\}$.

3(c): for $k = 1, \ldots, b$,

3(c)i: substitute $\hat{a}_{2\ell-1}$ by $\xi_k$;
3(c)ii: use Berlekamp/Massey algorithm to compute the the minimal linear generator of the new sequence $(\hat{a}_{2i-1})_{i=-3B+1}^{3B}$ and denote it by $\Lambda(z)$;
3(c)iii: if $\deg(\Lambda(z)) \leq 2B$, repeat Step 2.

If the error is $\hat{a}_{2\ell-1}$ with $1 \leq \ell \leq B$, that is $\hat{a}_{2\ell-1} \neq f(\gamma_{2\ell-1})$, then we substitute $\hat{a}_{2\ell-1}$ by a symbol $\alpha$. As the correct value $f(\gamma_{2\ell-1})$ is a solution of $\Delta_{2\ell-1}(\alpha) = 0$, that is $f(\gamma_{2\ell-1}) = \xi_k$ for some $k \in \{1, \ldots, b\}$, Step 3 will add $f$ into the list $L$.

Step 4: For $\ell = B + 1, \ldots, 2B$,

4(a): substitute $\hat{a}_{2\ell-1}$ by a symbol $\alpha$ in the matrix $G_{2(\ell-1)-1}$; compute the determinant of $G_{2(\ell-1)-1}$ and denote it by $\Delta_{2\ell-1}(\alpha)$.

According to Lemma 3.1, $\Delta_{2\ell-1}(\alpha)$ is a univariate polynomial of degree $B + 1$ in $\alpha$.
4(b): compute all solutions of the equation $\Delta_{2\ell-1}(\alpha) = 0$ in $K$; denote the solution set as $\{\xi_1, \ldots, \xi_{b'}\}$;
4(c): for $k = 1, \ldots, b'$,
   4(c)i: substitute $\hat{a}_{2\ell-1}$ by $\xi_k$;
   4(c)ii: use Berlekamp/Massey algorithm to compute the minimal linear generator of the new sequence $(\hat{a}_{2i-1})_{i=-3B+1}^{3B}$ and denote it by $\Lambda(z)$;
   4(c)iii: if $\deg(\Lambda(z)) \leq 2B$, repeat Step 2.

If the error is $\hat{a}_{2\ell-1} (B + 1 \leq \ell \leq 2B)$, that is $\hat{a}_{2\ell-1} \neq f(\gamma_{2\ell-1})$, we also substitute $\hat{a}_{2\ell-1}$ by a symbol $\alpha$. As the solution set $\{\xi_1, \ldots, \xi_{b'}\}$ of $\Delta_{2\ell-1}(\alpha) = 0$ contains $f(\gamma_{2\ell-1})$, Step 4 will add $f$ into the list $L$.

Step 5: Return the list $L$.

**Proposition 3.2.** The output list of Algorithm 3.1 contains $\leq 2B^2 + 2B + 1$ polynomials.

**Proof.** The Step 2 in Algorithm 3.1 produces $\leq 1$ polynomial, and both Step 3 and Step 4 produce $\leq B(B+1)$ polynomials. Hence the final output list has $\leq 1+2B(B+1)$ polynomials. $\square$

### 3.2. Correcting $E$ Errors

The settings for $f(x)$ are the same as in Section 3.1. We show how to list interpolate $f(x)$ from $N$ evaluations containing $\leq E$ errors, where

$$N = \left\lceil \frac{3}{2} E + 2 \right\rceil B. \tag{21}$$

Let $\theta = \lfloor E/2 \rfloor$. Choose $\omega_1, \ldots, \omega_{\theta}, \omega_{\theta+1} \in K \setminus \{0\}$ such that $\omega_\sigma$ has order $\geq 4D + 1$ for $1 \leq \sigma \leq \theta + 1$.

If $E$ is even then $N = (E/2)3B + 2B$. The problem is reduced to one the following situations: (1) the last block $(\hat{a}_{\theta+1,2i-1})_{i=1}^{3B}$ of length $2B$ is free of errors, or (2) there is some block $(\hat{a}_{\sigma,2i-1})_{i=1}^{3B}$ with $1 \leq \sigma \leq E/2$ of length $3B$ contains $\leq 1$ errors. These two situations can be respectively dealt with the Algorithm 2.2 and Algorithm 3.1.

If $E$ is odd then $E = 2\theta + 1$ and $N = (\theta+1)3B$. Thus, there is some block $(\hat{a}_{\sigma,1}, \ldots, \hat{a}_{\sigma,3B})$ with $1 \leq \sigma \leq \theta + 1$ of length $3B$ contains $\leq 1$ error; we can use the Algorithm 3.1 on this block to list interpolate $f(x)$.

**Remark 3.1.** For every $\sigma \in \{1, \ldots, \lfloor E/2 \rfloor\}$, we apply Algorithm 3.1 on the block $(\hat{a}_{\sigma,2i-1})_{i=1}^{3B}$ which will result in $\leq \lfloor E/2 \rfloor (2B^2 + 2B + 1)$ polynomials by Proposition 3.2. The length of the last block depends on the value of $E$, and we have following different upper bounds on the number of resulting polynomials:

1. $(E/2)(2B^2 + 2B + 1) + 1$, if $E$ is even;
2. $(\lfloor E/2 \rfloor + 1)(2B^2 + 2B + 1)$, if $E$ is odd.
Due to Obrechkoff’s theorem, a generalization of Descartes’ rule of signs to orthogonal polynomials [Dimitrov and Rafaeli 2009, Theorem 1.1], our approach for correcting E errors gives a unique valid sparse interpolant when \( K = \mathbb{R}, N \geq 2B + 2E \) and \( \omega_\sigma > 1 \) [Arnold and Kaltofen 2015, Corollary 2.4]. Similar to the case of power basis, if \( N < 2B + 2E \) then there can be \( \geq 2 \) valid sparse interpolants in Chebyshev-1 basis as shown by the following example.

**Example 3.2.** Choose \( \omega > 1 \). The polynomials \( h, f^{[1]} \) and \( f^{[2]} \), given in Example 2.3, can be represented in Chebyshev-1 basis using the following formula [Fraser 1965, P. 303], [Cody 1970, P. 412], [Mathar 2006, Eq. (2)]:

\[
 x^d = \frac{1}{2^{d-1}} \sum_{j=0}^{d} \left( \frac{d}{(d-j)/2} \right) \prod_{j \text{ is even}} \begin{cases} T_j(x) & \text{if } j \geq 1, \\
 \frac{1}{2} & \text{if } j = 0. \end{cases}
 \] (22)

Moreover, the formula (22) implies that \( f^{[1]} \) is a linear combination of the odd degree Chebyshev-1 polynomials \( T_{2j-1}(x) \) \((j = 1, 2, \ldots, B)\), and \( f^{[2]} \) is a linear combination of the even degree Chebyshev-1 polynomials \( T_{2j-2}(x) \) \((j = 1, 2, \ldots, B)\), which means both \( f^{[1]} \) and \( f^{[2]} \) have sparsity \( \leq B \) in Chebyshev-1 basis as well. Therefore, \( f^{[1]} \) and \( f^{[2]} \) are also valid interpolants in Chebyshev-1 basis for the \( 2B + 2E - 1 \) evaluations given in (14) (if we assume \( B \) is an upper bound on the sparsity of the black-box polynomial \( f \) and \( E \) is an upper bound on the number of errors in the evaluations).

Again, we remark that one of the valid interpolants, \( f^{[1]} \) and \( f^{[2]} \), must have sparsity \( B \) since otherwise uniqueness is a consequence of the Obrechkoff’s theorem [Dimitrov and Rafaeli 2009, Theorem 1.1]. In this example, \( h \) also has \( 2B \) terms in Chebyshev-1 basis because \( \deg(h) = 2B - 1 \) and \( h \) has \( 2B - 1 \) real roots \( \omega^i > 1, i = 1, \ldots, 2B - 1 \). Thus both \( f^{[1]} \) and \( f^{[2]} \) have sparsity \( B \) in Chebyshev-1 basis. One can also make \( h, f^{[1]} \) and \( f^{[2]} \) sparse with respect to Chebyshev-1 basis by the following substitutions:

\[
 x = T_k(y), \ \omega = T_k(\bar{\omega}) \text{ for some } k \gg 1.
\]

For \( K = \mathbb{C} \), we usually choose \( \omega \) as a root of unity. But then we may need \( 2B(2E + 1) \) evaluations to get a unique interpolant. Here is an example from [Kaltofen and Pernet 2014, Theorem 3], simply by changing the power basis to Chebyshev-1 basis.

**Example 3.3.** Consider the following two polynomials:

\[
 f_1(x) = \frac{1}{t} \sum_{j=0}^{t-1} T_{2j \frac{m}{t}}(x)
\]

\[
 f_2(x) = -\frac{1}{t} \sum_{j=0}^{t-1} T_{(2j+1) \frac{m}{t}}(x),
\]

where \( m \geq 2t(2E + 1) - 1 \) and \( 2t \) divides \( m \). Let \( \omega \) be a primitive \( m \)-th root of unity. Let \( b = (0, \ldots, 0, 1, 0, \ldots, 0) \in K^{2t-1} \).
The evaluations of $f_1$ at $\frac{\omega^i + \omega^{-i}}{2}$ for $i = 1, 2, \ldots, 2t(2E + 1) - 1$ are

$$\left( b, 1, \ldots, b, 1, b \right) \in K^{2t(2E+1)-1}.$$

The evaluations of $f_2$ at $\frac{\omega^i + \omega^{-i}}{2}$ for $i = 1, 2, \ldots, 2t(2E + 1) - 1$ are

$$\left( b, -1, \ldots, b, -1, b \right) \in K^{2t(2E+1)-1}.$$

Suppose we probe the black box for $f$ at $\frac{\omega^i + \omega^{-i}}{2}$ with $i = 1, 2, \ldots, 2t(2E + 1) - 1$ sequentially, and obtain the following sequence of evaluations:

$$\hat{a} = \left( b, 1, \ldots, b, 1, b, -1, \ldots, b, -1, b \right) \in K^{2t(2E+1)-1}.$$

Assume $B = t$ and there are $E$ errors in the sequence $\hat{a}$. Then both $f_1$ and $f_2$ are valid interpolants for $\hat{a}$. More specifically, $f_1$ is a valid interpolant for $\hat{a}$ if the $E$ errors are $\hat{a}_{2t}, \hat{a}_{2t+2}, \ldots, \hat{a}_{2t+E}$; $f_2$ is a valid interpolant for $\hat{a}$ if the $E$ errors are $\hat{a}_{2t(E+1)}, \hat{a}_{2t(E+2)}, \ldots, \hat{a}_{2t(2E)}$.

**Remark 3.2.** Polynomials in Chebyshev-2, Chebyshev-3 and Chebyshev-4 bases can be transformed into Laurent polynomials using the formulas given in [Imamoglu, Kaltofen, and Yang 2018, Sec. 1, (7)-(9)]. Therefore, our approach to list-interpolate black-box polynomials in Chebyshev-1 bases also works for black-box polynomials in Chebyshev-2, Chebyshev-3 and Chebyshev-4 bases.

**References**


## A. Appendix

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\hat{a}_i$</td>
<td>the output of the black box for $f$ at input $\omega^i$</td>
</tr>
<tr>
<td>$\alpha$</td>
<td>a symbol that substitute the single error in a block of $3B$ outputs of the black box for $f$</td>
</tr>
<tr>
<td>$\alpha_1, \alpha_2$</td>
<td>symbols that substitute the two errors in a block of $4B$ outputs of the black box for $f$</td>
</tr>
<tr>
<td>$B \geq t$, $B \geq t$</td>
<td>an upper bound on the sparsity of $f$</td>
</tr>
<tr>
<td>$b$</td>
<td>number of solutions to polynomial equation(s) for hypothetical errors</td>
</tr>
<tr>
<td>$\beta$</td>
<td>$=(\omega + 1/\omega)/2$, evaluation point of Chebyshev-1 polynomials</td>
</tr>
<tr>
<td>$c_j$</td>
<td>the coefficient of the $j$-th term of $f$</td>
</tr>
<tr>
<td>$D \geq</td>
<td>\delta_j</td>
</tr>
<tr>
<td>$\delta_j$</td>
<td>the $j$-th term degree of $f$</td>
</tr>
<tr>
<td>$\Delta$</td>
<td>a matrix determinant</td>
</tr>
<tr>
<td>$E$</td>
<td>an upper bound on the number of errors that is input to the algorithm</td>
</tr>
<tr>
<td>$f$</td>
<td>the black-box polynomial</td>
</tr>
<tr>
<td>$\gamma_i$</td>
<td>$=(\omega^i + 1/\omega^i)/2$, inputs of the black box for $f$ if $f$ is in Chebyshev bases</td>
</tr>
<tr>
<td>$G_r \in K^{(B+1)\times(B+1)}$, $G_r \in K^{(B+1)\times(B+1)}$, $G_r \in K^{(B+1)\times(B+1)}$</td>
<td>the Hankel+Toeplitz matrix with $\hat{a}_{r+i+j}$ on its $(i+1)$-th row and $(j+1)$-th column</td>
</tr>
<tr>
<td>$H_r \in K^{(B+1)\times(B+1)}$, $H_r \in K^{(B+1)\times(B+1)}$, $H_r \in K^{(B+1)\times(B+1)}$</td>
<td>the Hankel matrix with $\hat{a}<em>{r+i-1}, \hat{a}</em>{r+i}, \ldots, \hat{a}_{r+i-1+B}$ on its $i$-th row</td>
</tr>
<tr>
<td>$K$</td>
<td>a field of scalars with characteristic $\neq 2$</td>
</tr>
<tr>
<td>$\xi_i$</td>
<td>candidates for the correct value $f(\omega^i)$ if $\hat{a}_i$ is assumed to be an error</td>
</tr>
<tr>
<td>$\xi_{1,i}, \xi_{2,i}$</td>
<td>candidates for the pair of correct values $f(\omega^{\ell_1}), f(\omega^{\ell_2})$ if $\hat{a}<em>{\ell_1}$ and $\hat{a}</em>{\ell_2}$ are assumed to be errors</td>
</tr>
<tr>
<td>$\ell$</td>
<td>the error location in the outputs of the black box for $f$ if $E = 1$</td>
</tr>
<tr>
<td>$\ell_1, \ell_2$</td>
<td>the error locations in the outputs of the black box for $f$ if $E = 2$</td>
</tr>
<tr>
<td>$L$</td>
<td>the output list of our list decoding algorithms</td>
</tr>
<tr>
<td>$\Lambda$</td>
<td>the term locator polynomial</td>
</tr>
<tr>
<td>$M$</td>
<td>the number of the output polynomials of our error-correcting algorithms</td>
</tr>
<tr>
<td>$N$</td>
<td>the number of the evaluations by the black box for $f$</td>
</tr>
<tr>
<td>$\omega$</td>
<td>a non-zero number in $K$, evaluation base point for the black-box polynomial $f$ when only one block of evaluations are needed</td>
</tr>
<tr>
<td>$\omega_\sigma = \sigma = 1, 2, \ldots, \theta + 1$, $\omega_\sigma = \sigma = 1, 2, \ldots, \theta + 1$, $\omega_\sigma = \sigma = 1, 2, \ldots, \theta + 1$</td>
<td>non-zero numbers in $K$, evaluation base points for the black box polynomial $f$ when multiple blocks of evaluations are needed</td>
</tr>
<tr>
<td>$p_j$</td>
<td>$1 \leq j \leq t$, the roots of the term locator polynomial $\Lambda$</td>
</tr>
<tr>
<td>$t$</td>
<td>the actual number of terms of $f$</td>
</tr>
<tr>
<td>$\theta = \lfloor E/3 \rfloor$ if the black-box polynomial $f$ is in power basis, or $= \lfloor E/2 \rfloor$ if the black-box polynomial $f$ is in Chebyshev bases</td>
<td></td>
</tr>
<tr>
<td>$\zeta_i$</td>
<td>distinct, algorithm-dependent arguments in $K$</td>
</tr>
</tbody>
</table>