On Extensions of the Ben-Or/Tiwari and Prony Algorithms

Mark Giesbrecht **George Labahn** **Wen-shin Lee**
Symbolic Computation Group
University of Waterloo

May 18, 2002

In 1795, Gaspard Clair Franois Marie Riche de Prony gave a two-step method for fitting sums of exponential functions.

It is closely related to the modern Ben-Or/Tiwari multivariate sparse interpolation algorithm.

Essai expérimental et analytique sur les lois de la dilatabilité et sur celles de la force expansive de la vapeur de l'eau et de la vapeur de l'alkool, à diff'erentes températures.

J. de l' École Polytechnique 1:24–76, 1795.

For a function $f : \mathbb{R} \to \mathbb{R}$, and $t \in \mathbb{Z}_{>0}$, find c_i, a_i such that

$$f(x) = \sum_{i=1}^{t} c_i e^{a_i x}$$

Sparse Interpolation with Ben-Or/Tiwari and Prony's algorithms

de Prony	Ben-Or/Tiwari
Interpolate: $f(x) = \sum_{i=1}^{t} c_i e^{a_i x}$	Interpolate: $f(x_1,\dots,x_n) = \sum_{i=1}^{t} c_i x_1^{d_{1,i}} \cdots x_n^{d_{n,i}}$
1. Solve λ_j, $i = 0,\dots,t-1$: $\sum_{j=0}^{t-1} \lambda_j f(i+j) = -f(i+t)$ 2. e^{a_i} are zeros of $\Lambda = z^t + \lambda_{t-1} z^{t-1} + \cdots + \lambda_0$	1. Compute[†] the minimal Λ that generates[*] $\{f(p_1^i,\dots,p_n^i)\}_{i=0}^{2t-1}$ 2. $p_1^{d_{1,i}} \cdots p_n^{d_{n,i}}$ are zeros of $\Lambda = z^t + \lambda_{t-1} z^{t-1} + \cdots + \lambda_0$
3. Determine c_i from e^{a_i} and evaluations of f	3. Determine c_i from $p_1^{d_{1,i}} \cdots p_n^{d_{n,i}}$ and evaluations of f

† Berlekamp/Massey algorithm

$*$ $p_1,\dots,p_n$ distinct primes

Ben-Or/Tiwari Algorithm and Its Early Termination

- Compute minimal generator of linearly recurring sequence

$$f(p_1,\ldots,p_n),\ f(p_1^2,\ \ldots,\ p_n^2),\ \ldots,\ f(p_1^i,\ \ldots,\ p_n^i),\ \ldots$$

by Berlekamp/Massey algorithm with distinct random p_j.

> **In the Berlekamp/Massey algorithm:**
> The algorithm processes elements from a field, computing a "discrepancy" Δ from the actual minimal polynomial. When $\Delta = 0$ at $i > 2L$, with high probability the minimal generator $\Lambda(z)$ is determined and $i = 2t+1$, where t the number of terms in f.

- Recover non-zero terms in f by finding roots of $\Lambda(z)$.
- Locate coefficients for non-zero terms in f.

See *"Early termination in Ben-Or/Tiwari sparse interpolation and a hybrid of Zippel's algorithm"* E. Kaltofen, W.-s. Lee, and A. A. Lobo. ISSAC 2000.

Under the standard power basis, this algorithm interpolates f and is sensitive to the number of terms. Early termination is based on the correctness of the algorithm when p_j are evaluated symbolically.

In $f(x_1,\ldots,x_n) = \sum_{i=1}^{t} c_i x_1^{e_{1,i}} \cdots x_n^{e_{n,i}}$, the support in powers of x_j is revealed as powers of p_j as f evaluated at

$$X = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} p_1^i \\ p_2^i \\ \vdots \\ p_n^i \end{bmatrix}, \quad \text{for } i \geq 1.$$

Polynomial Evaluations in Any Given Power Basis

When $f(x_1,\ldots,x_n)=\sum_{i=1}^{t} c_i y_1^{\delta_{1,i}}\cdots y_n^{\delta_{n,i}}$, where

$$\underbrace{\begin{bmatrix} a_{1,1} & a_{1,2} & \ldots & a_{1,n} \\ a_{2,1} & a_{2,2} & \ldots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,1} & a_{n,2} & \ldots & a_{n,n} \end{bmatrix}}_{A} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} + \underbrace{\begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_n \end{bmatrix}}_{S} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

The support in powers of y_j can be revealed if x_j make y_j behave as powers of p_j. That is, f evaluated at $x_1,\ldots,x_n$ such that

$$\begin{bmatrix} a_{1,1} & a_{1,2} & \ldots & a_{1,n} \\ a_{2,1} & a_{2,2} & \ldots & a_{2,n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,1} & a_{n,2} & \ldots & a_{n,n} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_n \end{bmatrix} = \begin{bmatrix} p_1^i \\ p_2^i \\ \vdots \\ p_n^i \end{bmatrix}, \quad \text{for } i \geq 1.$$

Sparse Interpolation Algorithm in Any Given Power Basis

- A power basis is given as A and S.
- Perform Ben-Or/Tiwari algorithm (its early termination) to

$$f(x_1,\ldots,x_n) \text{ evaluated at } A^{-1}\begin{bmatrix} p_1^i - s_1 \\ \vdots \\ p_n^i - s_n \end{bmatrix} \quad \text{for } i \geq 1.$$

Example: Interpolate $f(x_1,x_2)$ in power basis of $2x_1+x_2+1$, $3x_1+2x_2-5$: pick p_1, p_2 and perform Ben-Or/Tiwari on

$$f(2p_1-p_2-7, -3p_1+2p_2+13),$$
$$\vdots$$
$$f(2p_1^i-p_2^i-7, -3p_1^i+2p_2^i+13),$$
$$\vdots$$

Since $A^{-1}\begin{bmatrix} p_1^i - s_1 \\ p_2^i - s_2 \end{bmatrix} = \begin{bmatrix} 2 & -1 \\ -3 & 2 \end{bmatrix}\begin{bmatrix} p_1^i - 1 \\ p_2^i + 5 \end{bmatrix} = \begin{bmatrix} 2p_1^i - p_2^i - 7 \\ -3p_1^i + 2p_2^i + 13 \end{bmatrix}.$

Sparsest Shifts

For polynomial f in a power basis of $y_1, \ldots, y_n$ given as A and S, $[\theta_1, \ldots, \theta_n]^{\mathrm{T}}$ is a sparsest shift if the number of terms τ is minimized in

$$f(x_1, \ldots, x_n) = \sum_{i=1}^{\tau} c_i (y_1 + \theta_1)^{\delta_{1,i}} \cdots (y_n + \theta_n)^{\delta_{n,i}}.$$

Computing Sparsest Shifts in the Standard Power Basis

The case $A = I_n$, we leave $[\theta_1, \ldots, \theta_n]^{\mathrm{T}}$ as symbols:

$$\begin{bmatrix} 1 & 0 & \ldots & 0 \\ 0 & 1 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_n \end{bmatrix} = \begin{bmatrix} p_1^i \\ p_2^i \\ \vdots \\ p_n^i \end{bmatrix}, \quad \text{for } i \geq 1.$$

Perform the fraction-free Berlekamp/Massey algorithm on

$$f(p_1-\theta_1,\ldots,p_n-\theta_n),\ldots,f(p_1^i-\theta_1,\ldots,p_n^i-\theta_n),\ldots$$

The fraction-free Berlekamp/Massey algorithm:
This algorithm processes elements from an integral domain. Now the "discrepancies" $\Delta(\theta_1,\ldots,\theta_n)$ are polynomials in $\theta_1,\ldots,\theta_n$. We find the first $(\theta_1,\ldots,\theta_n)$ such that $\Delta(s_1,\ldots,s_n)=0$ at $i>2L$, which occurs at $i=2\tau+1$. A sparsest shift will force the sparse interpolation terminate in its shifted power basis.

When $f\in\mathbb{Q}[x]$, we have efficient probabilistic algorithm that can interpolate f and compute θ at the same time, that is, f is interpolated with respect to a sparsest shifted power basis.

See *"Algorithms for computing the sparsest shifts for polynomials via the Berlekamp/Massey algorithm"* M. Giesbrecht, E. Kaltofen, and W.-s. Lee. To appear, ISSAC 2002.

Computing Sparsest Shifts in Any Given Power Basis

- A power basis is given as A and $S = [s_1, \ldots, s_n]^{\mathrm{T}}$.
- Perform the fraction-free Berlekamp/Massey algorithm on $f(x_1, \ldots, x_n)$ evaluated at

$$A^{-1} \begin{bmatrix} p_1^i - s_1 - \theta_1 \\ p_2^i - s_2 - \theta_2 \\ \vdots \\ p_n^i - s_n - \theta_n \end{bmatrix} \quad \text{for } i \geq 1,$$

and find the first $(\theta_1, \ldots, \theta_n)$ such that

$$\Delta(\theta_1, \ldots, \theta_n) = 0 \text{ and } i > 2L.$$

The Sparsifying Linear Transforms

For polynomial f, A is a sparsifying linear transform if

$$AX = Y = [y_1, \ldots, y_n]^{\mathrm{T}} \text{ and } f(x_1, \ldots, x_n) = g(y_1, \ldots, y_n),$$

where g is sparse.

Assume $A = [a_{i,j}]_{i,j=1}^n$ non-singular, and leave $a_{i,j}$ as symbols. In general, if we evaluate f at

$$A^{-1} \begin{bmatrix} p_1^i \\ p_2^i \\ \vdots \\ p_n^i \end{bmatrix} \text{ for } i \geq 0,$$

the symbolic $a_{i,j}$ could appear in the denominator of a rational function. Instead of the fraction-free Berlekamp/Massey algorithm, we need to apply the original BM algorithm which could cause **extreme intermediate expression swell.**

Whenever $\det A$ is a value in the coefficient domain, the fraction-free Berlekamp/Massey algorithm can still be employed. Especially if A is triangular with all diagonals 1:

$$\begin{bmatrix} 1 & * & \dots & * \\ 0 & 1 & \dots & * \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & 1 \end{bmatrix} \text{ or } \begin{bmatrix} 1 & 0 & \dots & 0 \\ * & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ * & \dots & * & 1 \end{bmatrix}.$$

Interesting Cases To Consider:

- A is banded.
- The entries of A are integers or rational numbers.
- A does not have full rank.