Sparse Polynomial Interpolation

Wen-shin Lee

November 8 2004
National Taiwan University

Outline

Black box sparse polynomial interpolation

Early termination strategy

Sparse representation of polynomials

Symbolic-numeric sparse interpolation

Future directions

Outline

- Black box sparse polynomial interpolation

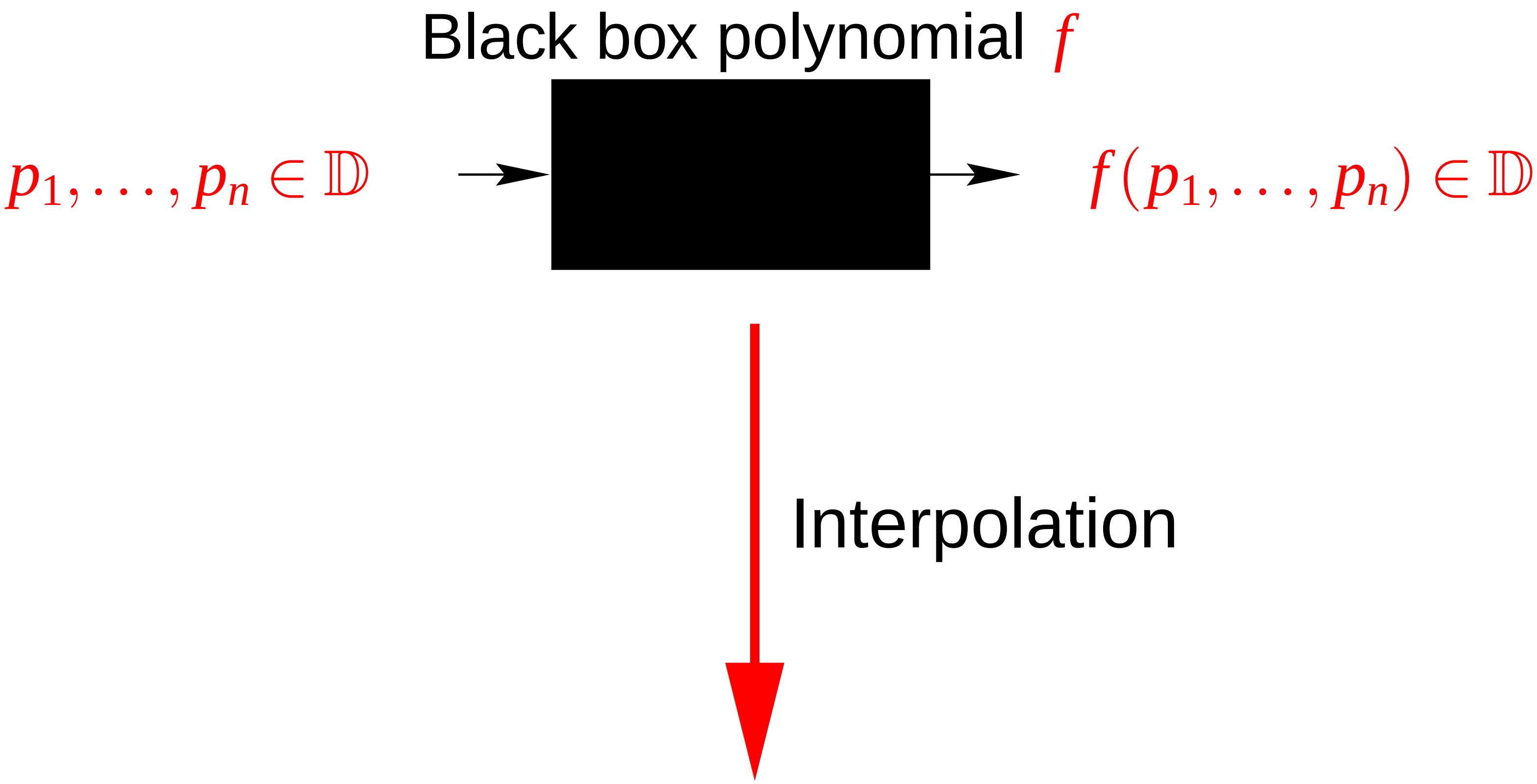
  Early termination strategy

  Sparse representation of polynomials

  Symbolic-numeric sparse interpolation

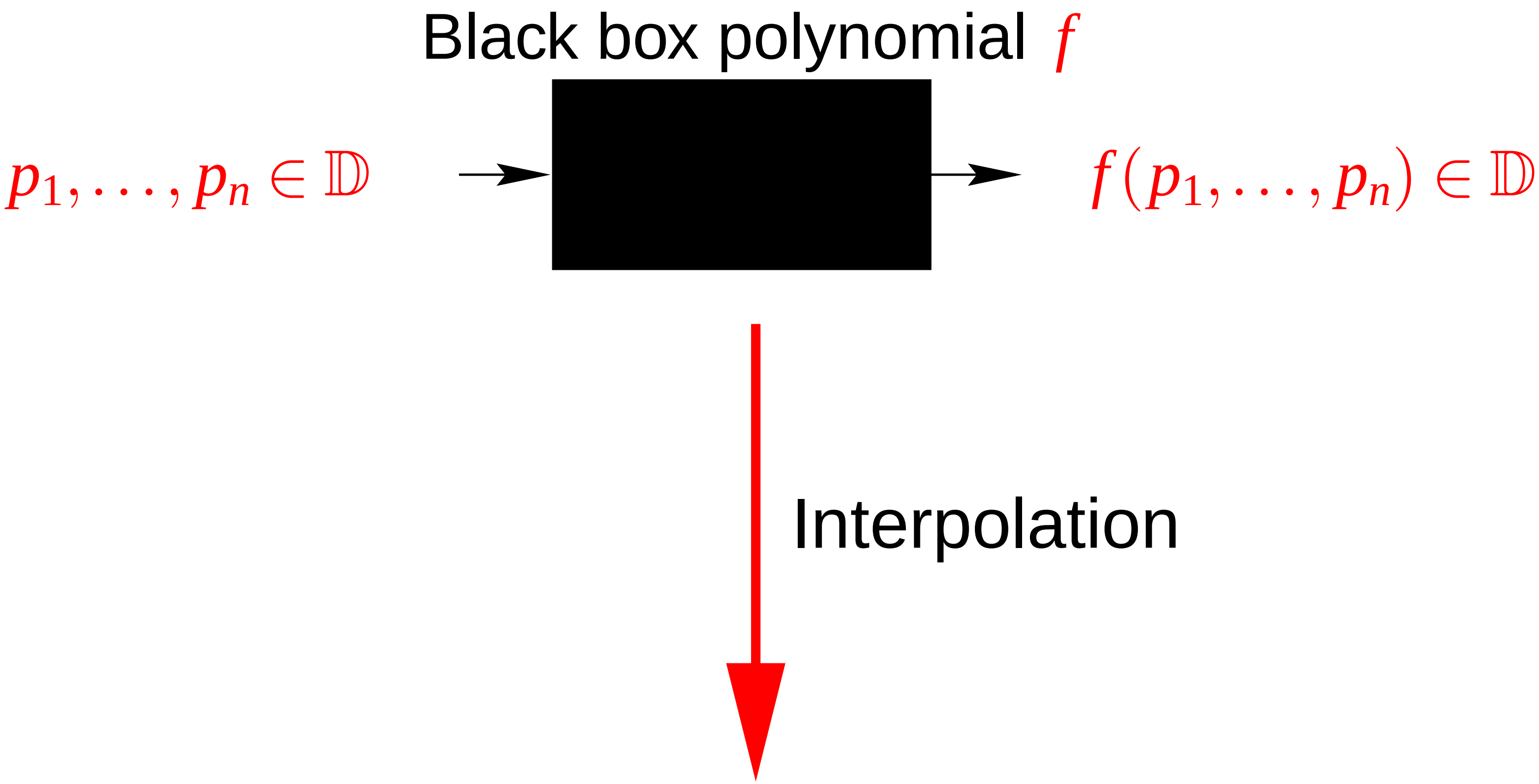
  Future directions

Black box polynomial interpolation



$$f(x_1, \ldots, x_n) = \sum_{j=1}^{t} c_j x_1^{d_{j1}} \cdots x_n^{d_{jn}} \in \mathbb{D}[x_1, \ldots, x_n]$$

Black box polynomial interpolation



$$f(x_1, \ldots, x_n) = \sum_{j=1}^{t} c_j x_1^{d_{j_1}} \cdots x_n^{d_{j_n}} \in \mathbb{D}[x_1, \ldots, x_n]$$

What if $f(x_1, \ldots, x_n)$ is sparse?

When $f(x_1,\ldots,x_n) = \sum_{j=1}^{t} c_j x_1^{d_{j1}} \cdots x_n^{d_{jn}}$ is sparse:

e.g. $f(x_1,x_2) = 3x_1^{1000}x_2^{150} - 17x_1^{999}x_2^{500}$

Zippel's probabilistic interpolation (1979)

Need: $D_k \geq \deg_{x_k} f$ for $1 \leq k \leq n$

Ben-Or/Tiwari deterministic algorithm (1988)

Need: $T \geq t$

Zippel's probabilistic algorithm: $f(x,y) = 3x^5y^3 + 2x^5 + y^2 + 5$

Zippel's probabilistic algorithm: $f(x,y) = 3x^5y^3 + 2x^5 + y^2 + 5$

- Pick a random a, interpolate $f(x,a) = C_5(a)x^5 + C_0(a)$.

Zippel's probabilistic algorithm: $f(x,y) = 3x^5y^3 + 2x^5 + y^2 + 5$

- Pick a random a, interpolate $f(x,a) = C_5(a)x^5 + C_0(a)$.

$$C_1(a) = \cdots = C_4(a) = 0$$

$$\implies C_1(y) = \cdots = C_4(y) = 0$$ with high probability

Zippel's probabilistic algorithm: $f(x,y) = 3x^5y^3 + 2x^5 + y^2 + 5$

- Pick a random a, interpolate $f(x,a) = C_5(a)x^5 + C_0(a)$.

$$C_1(a) = \cdots = C_4(a) = 0 \implies C_1(y) = \cdots = C_4(y) = 0 \text{ with high probability}$$

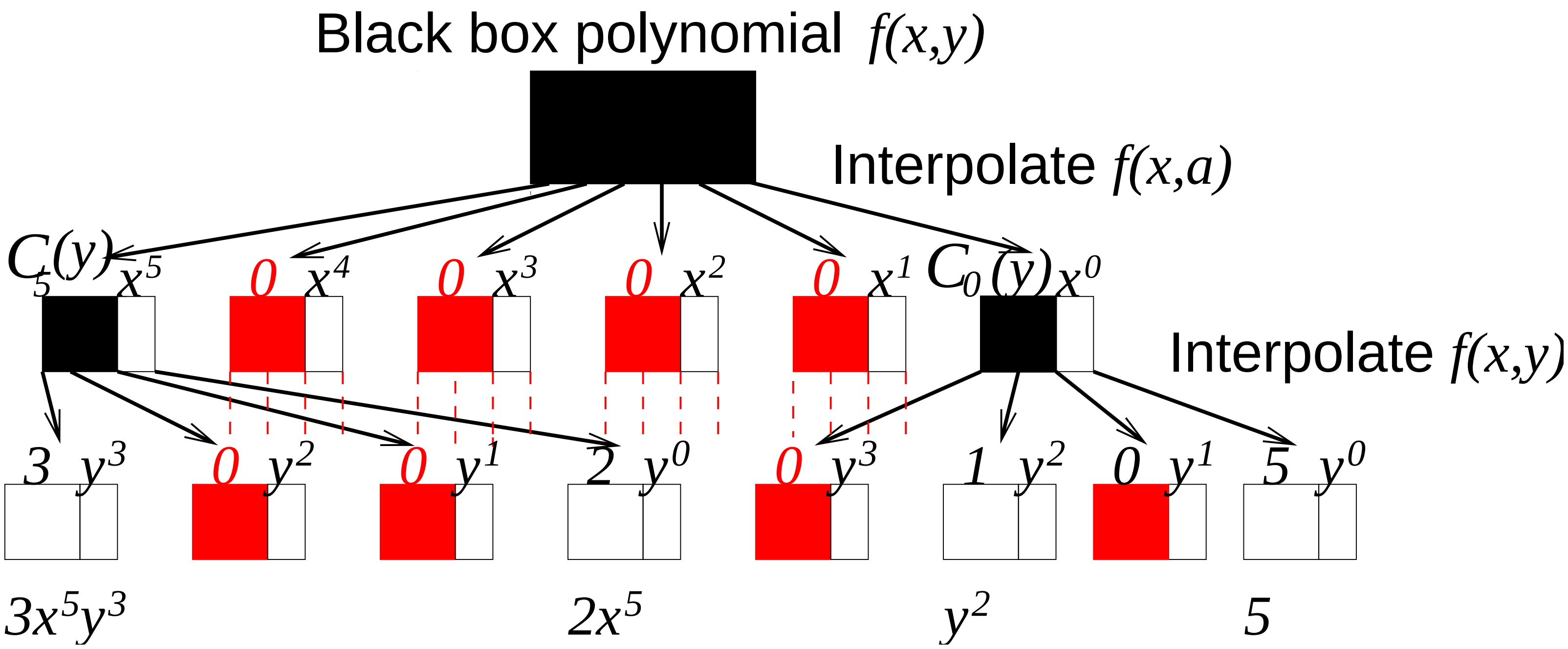
- Interpolate $C_5(y)$ and $C_0(y)$.

Zippel's probabilistic algorithm: $f(x,y) = 3x^5y^3 + 2x^5 + y^2 + 5$

- Pick a random a, interpolate $f(x,a) = C_5(a)x^5 + C_0(a)$.

$$C_1(a) = \cdots = C_4(a) = 0$$
$$\Longrightarrow C_1(y) = \cdots = C_4(y) = 0 \text{ with high probability}$$

- Interpolate $C_5(y)$ and $C_0(y)$.



Ben-Or/Tiwari algorithm: $f(x,y) = x^5 y + 0.1 x y^{13} - 0.5 x y^{11} + 2.2 x^4 y^4$

Ben-Or/Tiwari algorithm: $f(x,y) = x^5y + 0.1xy^{13} - 0.5xy^{11} + 2.2x^4y^4$

- Pick p_1, p_2 relatively prime: say $p_1 = 2,\ p_2 = 3.$

Ben-Or/Tiwari algorithm: $f(x,y) = x^5y + 0.1xy^{13} - 0.5xy^{11} + 2.2x^4y^4$

- Pick p_1, p_2 relatively prime: say $p_1 = 2$, $p_2 = 3$.
  Evaluate $a_0 = f(2^0, 3^0), a_1 = f(2,3), a_2 = f(2^2, 3^2), \ldots$.

Ben-Or/Tiwari algorithm: $f(x,y) = x^5y + 0.1xy^{13} - 0.5xy^{11} + 2.2x^4y^4$

- Pick p_1, p_2 relatively prime: say $p_1 = 2$, $p_2 = 3$.
  Evaluate $a_0 = f(2^0, 3^0), a_1 = f(2,3), a_2 = f(2^2, 3^2), \ldots$.
  sequence $a_0, a_1, a_2, \ldots$ is linearly generated

Ben-Or/Tiwari algorithm: $f(x,y) = x^5y + 0.1xy^{13} - 0.5xy^{11} + 2.2x^4y^4$

- Pick p_1, p_2 relatively prime: say $p_1 = 2$, $p_2 = 3$.
  Evaluate $a_0 = f(2^0, 3^0), a_1 = f(2,3), a_2 = f(2^2, 3^2), \ldots$.

  sequence $a_0, a_1, a_2, \ldots$ is linearly generated

- Compute the minimal generator $G(z)$ by Berlekamp/Massey algorithm or solving a Hankel system.

Ben-Or/Tiwari algorithm: $f(x,y) = x^5y + 0.1xy^{13} - 0.5xy^{11} + 2.2x^4y^4$

- Pick p_1, p_2 relatively prime: say $p_1 = 2,\ p_2 = 3$.
  Evaluate $a_0 = f(2^0, 3^0), a_1 = f(2,3), a_2 = f(2^2, 3^2), \ldots$.

  sequence $a_0, a_1, a_2, \ldots$ is linearly generated

- Compute the minimal generator $G(z)$ by Berlekamp/Massey algorithm or solving a Hankel system.

$$
\begin{aligned}
G(z) = & z^4 - 3544332z^3 - 1134650042820z^2 \\
& - 1573008457549248z - 1405550128432803844
\end{aligned}
$$

Ben-Or/Tiwari algorithm: $f(x,y) = x^5y + 0.1xy^{13} - 0.5xy^{11} + 2.2x^4y^4$

- Pick p_1, p_2 relatively prime: say $p_1 = 2,\ p_2 = 3.$
  Evaluate $a_0 = f(2^0, 3^0), a_1 = f(2,3), a_2 = f(2^2, 3^2), \ldots.$

  sequence $a_0, a_1, a_2, \ldots$ is linearly generated

- Compute the minimal generator $G(z)$ by Berlekamp/Massey algorithm or solving a Hankel system.

  $$G(z) = z^4 - 3544332z^3 - 1134650042820z^2 - 1573008457549248z - 1405550128432803 84$$

- Roots of $G(z) = 0$ are non-zero terms in f at $x = p_1, y = p_2$:

Ben-Or/Tiwari algorithm: $f(x,y) = x^5y + 0.1xy^{13} - 0.5xy^{11} + 2.2x^4y^4$

- Pick p_1, p_2 relatively prime: say $p_1 = 2,\ p_2 = 3$.
  Evaluate $a_0 = f(2^0, 3^0), a_1 = f(2,3), a_2 = f(2^2, 3^2), \ldots$.

  sequence $a_0, a_1, a_2, \ldots$ is linearly generated

- Compute the minimal generator $G(z)$ by Berlekamp/Massey algorithm or solving a Hankel system.

$$G(z) = z^4 - 3544332z^3 - 1134650042820z^2 - 1573008457549248z - 140555012843280384$$

- Roots of $G(z) = 0$ are non-zero terms in f at $x = p_1, y = p_2$:

$$96 = 2^5 \cdot 3, \quad 1296 = 2^4 \cdot 3^4, \quad 354294 = 2 \cdot 3^{11}, \quad 3188646 = 2 \cdot 3^{13}$$

$$\Downarrow \qquad \Downarrow \qquad \Downarrow \qquad \Downarrow$$

$$x^5y \qquad x^4y^4 \qquad xy^{11} \qquad xy^{13}$$

Ben-Or/Tiwari algorithm: $f(x,y) = x^5y + 0.1xy^{13} - 0.5xy^{11} + 2.2x^4y^4$

- Pick p_1, p_2 relatively prime: say $p_1 = 2$, $p_2 = 3$.
  Evaluate $a_0 = f(2^0, 3^0), a_1 = f(2,3), a_2 = f(2^2, 3^2), \ldots$.

  sequence $a_0, a_1, a_2, \ldots$ is linearly generated

- Compute the minimal generator $G(z)$ by Berlekamp/Massey algorithm or solving a Hankel system.

$$G(z) = z^4 - 3544332z^3 - 1134650042820z^2 - 1573008457549248z - 14055501284328038 4$$

- Roots of $G(z) = 0$ are non-zero terms in f at $x = p_1, y = p_2$:

$$96 = 2^5 \cdot 3, \quad 1296 = 2^4 \cdot 3^4, \quad 354294 = 2 \cdot 3^{11}, \quad 3188646 = 2 \cdot 3^{13}$$

$$\Downarrow \qquad \Downarrow \qquad \Downarrow \qquad \Downarrow$$

$$x^5y \qquad x^4y^4 \qquad xy^{11} \qquad xy^{13}$$

- Recover coefficients by solving a Vandermonde system.

Outline

- Black box sparse polynomial interpolation
- **Early termination strategy**
- Sparse representation of polynomials
- Symbolic-numeric sparse interpolation
- Future directions

Without $T \geq t$ and $D_k \geq \deg_{x_k} f$

Guess and check

Early termination strategy

Early termination in Newton interpolation

- Interpolate $f(x)$ on random $p_0,\ p_1,\ p_2, \ldots \in S$

$$f^{[i]}(x) = c_0 + c_1(x-p_0) + \cdots + c_i(x-p_0)\cdots(x-p_{i-1})$$

Early termination in Newton interpolation

- Interpolate $f(x)$ on random $p_0,\ p_1,\ p_2, \ldots \in S$

$$f^{[i]}(x) = c_0 + c_1(x - p_0) + \cdots + c_i(x - p_0)\cdots(x - p_{i-1})$$

- When $c_i = 0$, $f = f^{[i]}$ and $i = \deg f + 2$ with high probability.

Early termination in Newton interpolation

- Interpolate $f(x)$ on random $p_0,\ p_1,\ p_2,\ldots \in S$

$$f^{[i]}(x) = c_0 + c_1(x-p_0) + \cdots + c_i(x-p_0)\cdots(x-p_{i-1})$$

- When $c_i = 0$, $f = f^{[i]}$ and $i = \deg f + 2$ with high probability.

Probability analysis

If $f^{[i-1]} = f^{[i]} = \cdots = f^{[i+\eta]}$, then $f^{[i]} = f$ with probability at least

$$1-(i+1)\left(\frac{\deg(f)}{\#(S)}\right)^{\eta}$$

Early termination Ben-Or/Tiwari sparse interpolation

(Kaltofen, Lee, Lobo 2000)

Interpolate: $f = \sum_{j=1}^{t} c_j x_1^{d_{j1}} \cdots x_n^{d_{jn}}$

- With distinct random $p_1, \ldots, p_n$, compute minimal linear generator Λ of $f(p_1, \ldots, p_n),\ f(p_1^2, \ldots, p_n^2), \ldots, f(p_1^i, \ldots, p_n^i), \ldots$.

> Berlekamp/Massey algorithm: compute "discrepancy" Δ_i.
> When $\Delta_i = 0$ at $i > 2L$, $i = 2t + 1$ and Λ is determined with high probability.

- Recover terms in f by finding roots of Λ.
- Locate coefficients c_j in f.

- Any power basis! (Giesbrecht, Kaltofen, Lee 2003)

Early termination sparse interpolation in non-standard bases

(Kaltofen, Lee 2003)

- Pochhammer (rising factorial) basis:

$$f(x) = \sum_{j=1}^{t} c_j x^{\overline{d}_j}$$

$$x^{\overline{n}} = x(x+1)\cdots(x+n-1)$$

- Chebyshev basis:

$$f(x) = \sum_{j=1}^{t} c_j T_{d_j}(x)$$

$$T_0(x) = 1, \quad T_1(x) = x$$
$$n \geq 2: \quad T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x)$$

Outline

- Black box sparse polynomial interpolation
- Early termination strategy
- • Sparse representation of polynomials
- Symbolic-numeric sparse interpolation
- Future directions

Sparse shifts

Consider the polynomial:

$$f(x_1, \ldots, x_n) = \sum_{j=1}^{t} c_j x_1^{d_{j1}} \cdots x_n^{d_{jn}}$$

in shifted basis $y_k = x_k + s_k$: $$= \sum_{j=1}^{\tau} \gamma_j \underbrace{(x_1 + s_1)}_{y_1}^{\delta_{j1}} \cdots \underbrace{(x_n + s_n)}_{y_n}^{\delta_{jn}}$$

τ depends on $s = (s_1, \ldots, s_n)$

Questions:

- Find a sparsest shift of f within set S:

  $s = (s_1, \ldots, s_n) \in S$ and τ is minimized.

- T-sparse shifts of f within set S:

  $s = (s_1, \ldots, s_n) \in S$ and $\tau \leq T$.

Sparse shift example:

$$f(x_1,x_2) = x_1^5x_2^2 - 6x_1^5x_2 + 9x_1^5 + 10x_1^4x_2^2 - 60x_1^4x_2 + 90x_1^4 + 40x_1^3x_2^2 - 240x_1^3x_2 + 360x_1^3 + 80x_1^2x_2^2 - 480x_1^2x_2 + 720x_1^2 + 80x_1x_2^2 - 480x_1x_2 + 720x_1 + 32x_2^2 - 192x_2 + 289$$

$$= (\underbrace{x_1+2}_{y_1})^5(\underbrace{x_2-3}_{y_2})^2 + 1$$

$(2,-3)$ is a sparsest shift of $f(x_1,x_2)$

Early termination Ben-Or/Tiwari sparse interpolation

$\sum_{j=1}^{t} c_j x_1^{d_{1,j}} \cdots x_n^{d_{n,j}}$ $\quad\quad$ $\sum_{j=1}^{\tau} \gamma_j (x_1 + s_1)^{\delta_{1,j}} \cdots (x_n + s_n)^{\delta_{n,j}}$

$\downarrow \Delta_i$ $\quad$ Berlekamp/Massey $\quad$ $\downarrow \Delta_i$

$\Delta_{2t+1} = 0$ $\quad$ symbolic $x_1, \ldots, x_n$ $\quad$ $\Delta_{2\tau+1} = 0$

$\downarrow$

Leave shifts s_k as symbols: $s_k \longrightarrow z_k$

Compute sparsest shifts $s = (s_1, \ldots, s_n)$: solve first $\Delta_i(z) = 0$
for symbolic $x_1, \ldots, x_n$

Minimize: $i = 2\tau + 1$

Compute sparsest shifts in the standard power basis

(Giesbrecht, Kaltofen, Lee 2002)

- Run the sparse interpolation with the shift as a symbol

  Perform fraction-free Berlekamp/Massey algorithm on

$$f(y_1 - z_1, \ldots, y_n - z_n), \ldots, f(y_1^i - z_1, \ldots, y_n^i - z_n), \ldots$$

> The fraction-free Berlekamp/Massey algorithm:
> $\Delta_i(z_1, \ldots, z_n, y_1, \ldots, y_n)$ are polynomials in $z_1, \ldots, z_n, y_1, \ldots, y_n$.

- Solve $z_1, \ldots, z_n$ in $\boxed{\Delta_i = 0}$ for all $y_1, \ldots, y_n$, which minimizes i.

When $f = f(x)$: symbolic; project y; $f(x) \in \mathbb{Q}[x]$.

Relevant developments in (Giesbrecht, Kaltofen, Lee 2003)

- How many sparse representations can a polynomial have?

  for univariate $f(x) = \sum_{j=1}^{t} c_j x^{d_j} = \sum_{j=1}^{\tau} \gamma_j x^{\delta_j}$

  $$t + \tau > \deg(f) + 1$$

- Sparsest shifts in any power basis

- Sparsest shifts for a set of polynomials $f_1, \ldots, f_m \in D[x_1, \ldots, x_n]$:

  consider $G(x_1, \ldots, x_n, z_0) = f_1 + z_0 f_2 + \cdots + z_0^{m-1} f_{m-1} + z_0^{m-1} f_m$.

- Sparse shifts in Chebyshev, Pochhammer bases

Outline

- Black box and sparse polynomials
- Early termination strategy
- Sparse representation of polynomials
- Symbolic-numeric sparse interpolation
- Future directions

Sparse interpolation of a black box polynomial

Black box $f = \sum_{j=1}^{t} c_j x_1^{d_{j,1}} \cdots x_n^{d_{j,n}}$

$p_1, \ldots, p_n$

$f(p_1, \ldots, p_n)$

Sparse Interpolation

$$\tilde{f} = \sum_{j=1}^{t} \tilde{c}_j x_1^{d_{j,1}} \cdots x_n^{d_{j,n}}$$

Determine $\tilde{c}_j$:

1. exactly
2. approximately, to a fixed precision

Example Black box $10x^6y^8 - 6x^{10} - 5x^8y - 4y^7$

	Exact	Approximate
Input	$\omega_x = \exp(\frac{2\pi i}{31})$ (31st-PRU)	$\mathrm{evalf}(\omega_x) = 0.9795299413 + 0.2012985201I$
	$\omega_y = \exp(\frac{2\pi i}{37})$ (37th-PRU)	$\mathrm{evalf}(\omega_y) = 0.9856159104 + 0.1690008203I$
Compute	$f(\omega_x^i, \omega_y^i)$	$\mathrm{evalf}(f(\mathrm{evalf}(\omega_x^i), \mathrm{evalf}(\omega_y^i))), i = 0, 1, \ldots, 8$
Output	$10x^6y^8$	$(10.00000006 - 0.8543610430 \times 10^{-8}I)x^6y^8$
	$-6x^{10}$	$(-6.000000235 - 0.1390185436 \times 10^{-6}I)x^{10}$
	$-5x^8y$	$(-4.999999825 + 0.1968676105 \times 10^{-6}I)x^8y$
	$-4y^7$	$(-3.999999997 - 0.493054565410^{-7}I)y^7$

Gaspard Clair Franois Marie Riche de Prony

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_j, μ_j such that

$$F(x) = \sum_{j=1}^{t} c_j e^{\mu_j x}$$

Methods: Prony (1795) $\sim$ Ben-Or/Tiwari (1988)

A sum of exponential functions $F(x) = \sum_{j=1}^{t} c_j e^{\mu_j x} = \sum_{j=1}^{t} c_j b_j^x$	A polynomial $f(x_1, \ldots, x_n) = \sum_{j=1}^{t} c_j x_1^{d_{j,1}} \cdots x_n^{d_{j,n}}$
1. Solve λ_j, $i = 0, \ldots, t-1$: $\sum_{j=0}^{t-1} \lambda_j F(i+j) = -F(i+t)$ 2. $e^{\mu_j} = b_j$ are zeros of $\Lambda = z^t + \lambda_{t-1} z^{t-1} + \cdots + \lambda_0$	1. Compute† the minimal Λ that generates* $\{f(p_1^i, \ldots, p_n^i)\}_{i=0}^{2t-1}$ 2. $p_1^{d_{j,1}} \cdots p_n^{d_{j,n}}$ are zeros of $\Lambda = z^t + \lambda_{t-1} z^{t-1} + \cdots + \lambda_0$
3. Determine c_j from $e^{\mu_i} = b_j$ and evaluations of F	3. Determine c_j from $p_1^{d_{j,1}} \cdots p_n^{d_{j,n}}$ and evaluations of f

† Berlekamp/Massey algorithm

$*$ $p_1, \ldots, p_n$ relatively prime

Numerical challenges in Prony's method

Ill-conditioned Hankel system

$$\underbrace{\begin{bmatrix} F(0) & F(1) & \dots & F(t-1) \\ F(1) & F(2) & \dots & F(t) \\ \vdots & \vdots & \ddots & \vdots \\ F(t-1) & F(t) & \dots & F(2t-2) \end{bmatrix}}_{H_{0,t-1}} \begin{bmatrix} \lambda_0 \\ \lambda_1 \\ \vdots \\ \lambda_{t-1} \end{bmatrix} = - \begin{bmatrix} F(t) \\ F(t+1) \\ \vdots \\ F(2t-1) \end{bmatrix}$$

Root-finding sensitive to perturbations in λ_j

$$\Lambda = z^t + \lambda_{t-1} z^{t-1} + \cdots + \lambda_0 = 0$$

Further challenge in Ben-Or/Tiwari algorithm

Recover multivariate terms in the target polynomial

Generalized eigenvalue reformulation (Golub, Milanfar, and Varah 1999)

$$H_{0,t-1} = \underbrace{\begin{bmatrix} 1 & \dots & 1 \\ b_1 & \dots & b_t \\ \vdots & \vdots & \vdots \\ b_1^{t-1} & \dots & b_t^{t-1} \end{bmatrix}}_{V} \underbrace{\begin{bmatrix} c_1 & 0 & \dots & 0 \\ 0 & c_2 & \dots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & c_t \end{bmatrix}}_{D} \underbrace{\begin{bmatrix} 1 & b_1 & \dots & b_1^{t-1} \\ 1 & b_2 & \dots & b_2^{t-1} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & b_t & \dots & b_t^{t-1} \end{bmatrix}}_{V^T}$$

$$\underbrace{\begin{bmatrix} F(1) & \dots & F(t) \\ \vdots & \ddots & \vdots \\ F(t) & \dots & F(2t-1) \end{bmatrix}}_{H_{1,t}} = VDBV^T \quad \text{with } B = \begin{bmatrix} b_1 & 0 & \dots & 0 \\ 0 & b_2 & \dots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \ddots & 0 & b_t \end{bmatrix}$$

$$V^{-1} H_{0,t-1} V^{-T} = D, \qquad V^{-1} H_{1,t} V^{-T} = DB$$

$\Longrightarrow H_{1,t} v = b H_{0,t-1} v$ has solutions $b_1, \dots, b_t$ for b.

Sparse interpolation via generalized eigenvalues

$$f(x) = \sum_{j=1}^{t} c_j x^{d_j}$$

$$\underbrace{\begin{bmatrix} f(p^0) & f(p) & \cdots & f(p^{t-1}) \\ f(p) & f(p^2) & \cdots & f(p^t) \\ \vdots & \vdots & \ddots & \vdots \\ f(p^{t-1}) & f(p^t) & \cdots & f(p^{2t-2}) \end{bmatrix}}_{H_{0,t-1}} v = z \underbrace{\begin{bmatrix} f(p) & f(p^{t+1}) & \cdots & f(p^t) \\ f(p^2) & f(p^3) & \cdots & f(p^{t+1}) \\ \vdots & \vdots & \ddots & \vdots \\ f(p^t) & f(p^{t+1}) & \cdots & f(p^{2t-1}) \end{bmatrix}}_{H_{1,t}} v$$

- Solutions for $z : \tilde{b}_1, \tilde{b}_2, \ldots, \tilde{b}_t$ approximate $p^{d_1}, p^{d_2}, \ldots, p^{d_t}$.

Multivariate case

$$f(x_1,\ldots,x_n) = \sum_{j=1}^{t} c_j x_1^{d_{j,1}} \cdots x_n^{d_{j,n}}$$

- Variable by variable (sometimes called "peeling method.")
- Everything at once (numerically better)

Evaluate each variable x_k at powers of ω_k

$$\omega_k^i = \exp(2\pi i / p_k) \text{ and } f(\omega_1^i,\ldots,\omega_n^i)$$

$p_1,\ldots,p_n$ relatively prime.

Recall: In the original Ben-Or/Tiwari algorithm, evaluate $f(p_1^i,\ldots,p_n^i)$ for $p_1,\ldots,p_n$ relatively prime.

The number of terms?

- Binary search

  Guess an upper bound $\tau \geq t$, double τ if fails.

- Early termination heuristic

  Cabay-Meleshko algorithm: a fast procedure estimates the condition number of a Hankel matrix $H_{0,N}$ for any N.

Approximate sparse interpolation in non-standard bases

Simultaneous diagonalizations

The general eigenvalue reformulation can be applied to interpolation systems M, N containing:

$$F^{-1}MG^{-1} = \bar{D}, F^{-1}NG^{-1} = \bar{D}\bar{B} \text{ with } \bar{D}, \bar{B} \text{ diagonal.}$$

- Chebyshev basis
- Factorial bases

Outline

- Black box and sparse polynomials
- Early termination strategy
- Sparse representation of polynomials
- Symbolic-numeric sparse interpolation
- Future directions

Future directions

- Shift/transform equivalence of polynomials
- Reduce volumes of Newton polytopes via changing bases
- Simplify a system of linear PDEs
- ...