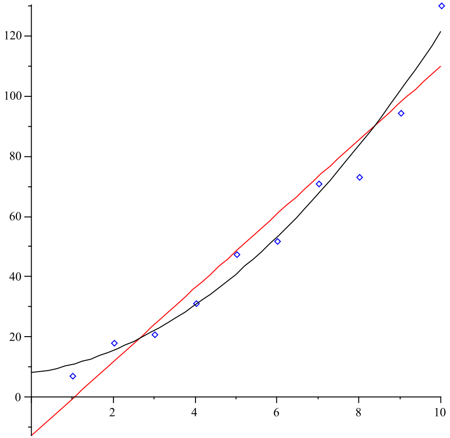
Cleaning-Up Data With Errors: When Symbolic-Numeric Sparse Interpolation Meets Error-Correcting Codes

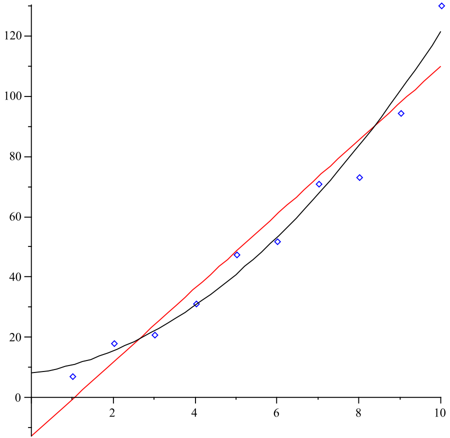
Erich L. Kaltofen
North Carolina State University
`google, bing->kaltofe`

Model Discovery Example



Linear or quadratic best fit?

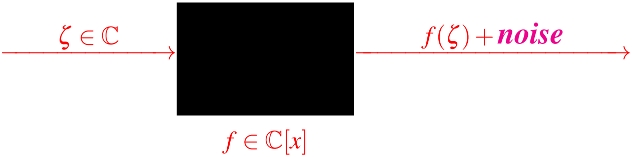
Model Discovery Example



Linear or quadratic best fit?
How many points are needed to discover a sparse model,

e.g., $2.5x^8 + x^2 - 5.7x$?

The Numeric Sparse Interpolation Problem
[Giesbrecht, Labahn, Lee 2003]

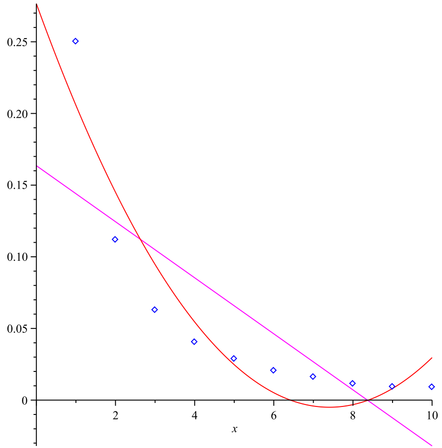


By sampling black box, compute T-sparse representation

$$f(x) = \sum_{j=1}^{T} c_j x^{e_j}, \quad 0 \neq c_j \in \mathbb{C}, e_j \in \mathbb{Z}$$

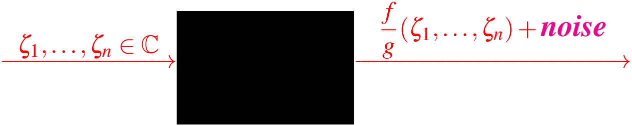
Note: T, e_j are not known (otherwise, a least squares problem)
Number of sample points $O(T)$, not $O(\deg(f))$

Rational Model Discovery Example



What if best model is $\dfrac{2.5x^7y^{10}+1.3}{x^2-y^9}$?

Sparse rational fitting [Kaltofen,Yang,Zhi SNC'07]



$f, g \in \mathbb{C}[x_1, \ldots, x_n], \mathrm{GCD}(f, g) = 1$

By sampling black box, compute sparse representation

$$\frac{\sum_{j=1}^{T_f} \tilde{a}_j x_1^{d_{j,1}} \cdots x_n^{d_{j,n}}}{\sum_{m=1}^{T_g} \tilde{b}_m x_1^{e_{m,1}} \cdots x_n^{e_{m,n}}} = \frac{\tilde{f}}{\tilde{g}}, \quad \tilde{a}_j \neq 0, \tilde{b}_k \neq 0$$

Note: Terms are **not** known.

Sparse rational fitting [Kaltofen,Yang,Zhi SNC'07]

$f, g \in \mathbb{C}[x_1,\ldots,x_n], \mathrm{GCD}(f,g) = 1$

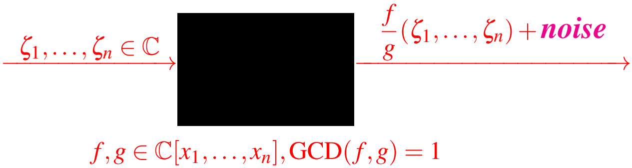
By sampling black box, compute sparse representation

$$\frac{\sum_{j=1}^{T_f} \tilde{a}_j x_1^{d_{j,1}} \cdots x_n^{d_{j,n}}}{\sum_{m=1}^{T_g} \tilde{b}_m x_1^{e_{m,1}} \cdots x_n^{e_{m,n}}} = \frac{\tilde{f}}{\tilde{g}}, \quad \tilde{a}_j \neq 0, \tilde{b}_k \neq 0$$

Note: Terms are **not** known.

Unfinished business: Prove that $O(n(T_f + T_g))$ points suffice
[$O(n(T_f + T_g)^2)$ points proven sufficient in '07]

Sparse rational fitting [Kaltofen,Yang,Zhi SNC'07]



By sampling black box, compute sparse representation

$$\frac{\sum_{j=1}^{T_f} \tilde{a}_j x_1^{d_{j,1}} \cdots x_n^{d_{j,n}}}{\sum_{m=1}^{T_g} \tilde{b}_m x_1^{e_{m,1}} \cdots x_n^{e_{m,n}}} = \frac{\tilde{f}}{\tilde{g}}, \quad \tilde{a}_j \neq 0, \tilde{b}_k \neq 0$$

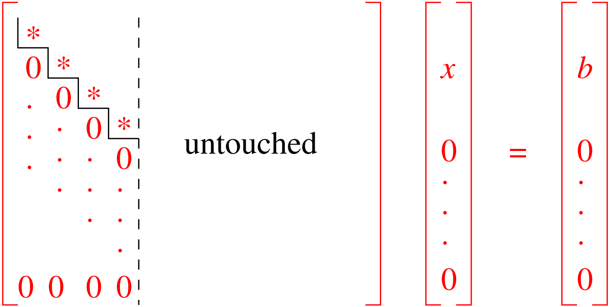
Note: Terms are **not** known.

Ingredients: 1. univariate dense rational function interpolation based on structured total least norm fitting
2. Zippel lifting + our numeric Zippel-Schwartz Lemma

Linear system solving in sublinear complexity
[Storjohann and Mulders 2004]

Solve a linear system $Ax = b$ without inspecting all entries

Consistency (assume initial columns give full rank):



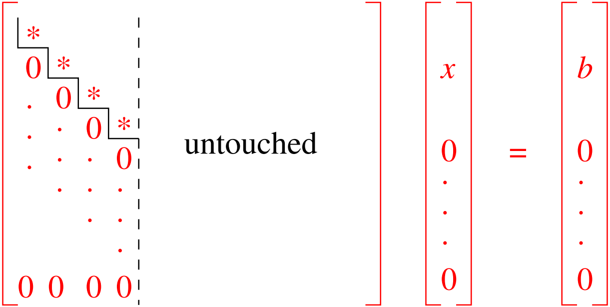
partial row echelon form
(* = pivot elements)

a solution
vector

Linear system solving in sublinear complexity [Storjohann and Mulders 2004]

Solve a linear system $Ax = b$ without inspecting all entries

Consistency (assume initial columns give full rank):



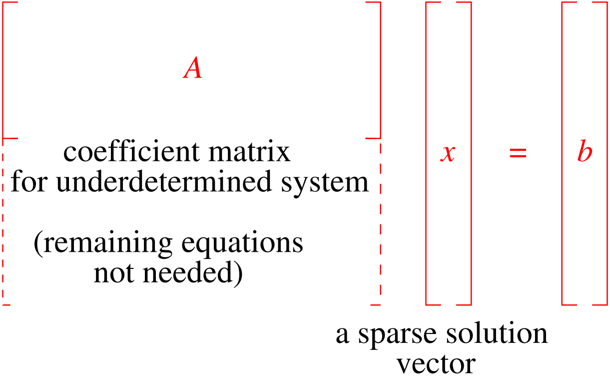
partial row echelon form (* = pivot elements)

a solution vector

New problem for “BIG DATA:” Row/column selection in linear algebra

Candès-Tao Column Selection via Sparsity Constraint

Assume solution x is sparse:



Compute x with minimal ℓ_1-norm $|x_1| + \cdots + |x_n|$ via linear programming

For x with T non-zero entries need $O(T^2)$ equations for certain A (RIP: *restricted isometry property*): reconstruct model from few observations

Example: Zippel lifting (Exact Case)

Given the black box of the rational function f/g

$$f = x_1^3 + 3x_1x_2^2, \quad g = 2x_1^3 + 3x_2,$$

and the degree bounds $\bar{d}_f = 4$, $\bar{d}_g = 4$. Suppose

$$f_1 = f(x_1, \alpha) = b_1x_1^3 + b_2x_1, \quad g_1 = g(x_1, \alpha) = b_3x_1^3 + b_4.$$

and $\bar{D}_f^{[2]} = \{1, x_2^2\}$, $\bar{D}_g^{[2]} = \{1, x_2\}$, where $b_1, b_2, b_3, b_4 \in \mathsf{K} \setminus \{0\}$.

Example: Zippel lifting (Exact Case)

Given the black box of the rational function f/g

$$f = x_1^3 + 3x_1x_2^2, \quad g = 2x_1^3 + 3x_2,$$

and the degree bounds $\bar{d}_f = 4$, $\bar{d}_g = 4$. Suppose

$$f_1 = f(x_1, \alpha) = b_1x_1^3 + b_2x_1, \quad g_1 = g(x_1, \alpha) = b_3x_1^3 + b_4.$$

and $\bar{D}_f^{[2]} = \{1, x_2^2\}$, $\bar{D}_g^{[2]} = \{1, x_2\}$, where $b_1, b_2, b_3, b_4 \in \mathsf{K} \setminus \{0\}$.

The sets of the possible terms of f and g are

$$D_{f,2} = \{x_1, x_1^3, x_1x_2^2\}, \quad D_{g,2} = \{1, x_1^3, x_1^3x_2, x_2\}.$$

Example continued

f and g can be represented as

$$f = y_1 x_1 + y_2 x_1^3 + y_3 x_1 x_2^2, \quad g = z_1 + z_2 x_1^3 + z_3 x_1^3 x_2 + z_4 x_2.$$

Pick random points $\xi_1, \xi_2 \in \mathsf{K}$ and compute the values:

$$-\gamma_\ell = \frac{f(\xi_1^\ell, \xi_2^\ell)}{g(\xi_1^\ell, \xi_2^\ell)} \in \mathsf{K} \cup \{\infty\},\ \ell = 0, 1, \ldots, L-1.$$

$$\Downarrow$$

$$\underbrace{\begin{bmatrix} 1 & 1 & 1 & \gamma_0 & \gamma_0 & \gamma_0 & \gamma_0 \\ \xi_1 & \xi_1^3 & \xi_1 \xi_2^2 & \gamma_1 & \gamma_1 \xi_1^3 & \gamma_1 \xi_1^3 \xi_2 & \gamma_1 \xi_2 \\ \xi_1^2 & (\xi_1^3)^2 & (\xi_1 \xi_2^2)^2 & \gamma_2 & \gamma_2 (\xi_1^3)^2 & \gamma_2 (\xi_1^3 \xi_2)^2 & \gamma_2 \xi_2^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \xi_1^{L-1} & (\xi_1^3)^{L-1} & (\xi_1 \xi_2^2)^{L-1} & \gamma_{L-1} & \gamma_{L-1} (\xi_1^3)^{L-1} & \gamma_{L-1} (\xi_1^3 \xi_2^2)^{L-1} & \gamma_{L-1} \xi_2^{L-1} \end{bmatrix}}_{G} \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ z_1 \\ z_2 \\ z_3 \\ z_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

Exact Probabilistic Analysis

L is the number of required evaluation points.

Unique sparse fraction from **consecutive powers of random points**
if $L \geq |D_{f,i}| \cdot |D_{g,i}| = O(T_f T_g)$, *with high probability*

Kaltofen July 14, 2013: $L \geq |D_{f,i}| + |D_{g,i}| - 1 = O(T_f + T_g)$
evaluations at **random** points suffice, *with high probability*

Exact Probabilistic Analysis

L is the number of required evaluation points.

Unique sparse fraction from **consecutive powers of random points** if $L \geq |D_{f,i}| \cdot |D_{g,i}| = O(T_f T_g)$, *with high probability*

Kaltofen July 14, 2013: $L \geq |D_{f,i}| + |D_{g,i}| - 1 = O(T_f + T_g)$ evaluations at **random** points suffice, *with high probability*

Compare with Candès-Tao's $O(T^2)$

Numeric stability because algorithm works on original data, not derived data

Exact Probabilistic Analysis

L is the number of required evaluation points.

Unique sparse fraction from **consecutive powers of random points** if $L \geq |D_{f,i}| \cdot |D_{g,i}| = O(T_f T_g)$, *with high probability*

Kaltofen July 14, 2013: $L \geq |D_{f,i}| + |D_{g,i}| - 1 = O(T_f + T_g)$ evaluations at **random** points suffice, *with high probability*

Compare with Candès-Tao's $O(T^2)$

Numeric stability because algorithm works on original data, not derived data

Linear system solved by Olshevsky-Shokrollahi STOC 1999 displacement operators

Idea of argument

1. evaluate f/g at $(x_1, \ldots, x_n) \leftarrow (v_1^\ell, \ldots, v_n^\ell)$,
$\ell \leftarrow 0, 1, 2, \ldots, 2T_f T_g - 1$, v_i symbolic variables

Let Φ, Ψ be a sparse interpolant: then

$$(\frac{f}{g} - \frac{\Phi}{\Psi})(v_1^\ell, \ldots, v_n^\ell) = 0 \Rightarrow (\underbrace{f\Psi - \Phi g}_{\leq 2T_f T_g \text{ terms}})(v_1^\ell, \ldots, v_n^\ell) = 0.$$

Term values $(v_1^\ell)^{e_1} \cdots (v_n^\ell)^{e_n} = (v_1^{e_1} \cdots v_n^{e_n})^\ell$ so coefficient vector nullifies (transposed) Vandermonde matrix $\Rightarrow f\Psi - \Phi g = 0$

[Ben-Or, Tiwari 1984; Kaltofen, Yang, Zhi 2007]

Idea of argument

1. evaluate f/g at $(x_1, \ldots, x_n) \leftarrow (v_1^\ell, \ldots, v_n^\ell)$,
$\ell \leftarrow 0, 1, 2, \ldots, 2T_f T_g - 1$, v_i symbolic variables

Let Φ, Ψ be a sparse interpolant: then

$$\left(\frac{f}{g} - \frac{\Phi}{\Psi}\right)(v_1^\ell, \ldots, v_n^\ell) = 0 \Rightarrow \underbrace{(f\Psi - \Phi g)}_{\leq 2T_f T_g \text{ terms}}(v_1^\ell, \ldots, v_n^\ell) = 0.$$

Term values $(v_1^\ell)^{e_1} \cdots (v_n^\ell)^{e_n} = (v_1^{e_1} \cdots v_n^{e_n})^\ell$ so coefficient vector nullifies (transposed) Vandermonde matrix $\Rightarrow f\Psi - \Phi g = 0$

[Ben-Or, Tiwari 1984; Kaltofen, Yang, Zhi 2007]

2. evaluate f/g at $(x_1, \ldots, x_n) \leftarrow (v_{1,\ell}, \ldots, v_{n,\ell}), \ell \leftarrow 0, 1, 2, \ldots,$
$v_{i,\ell}$ symbolic variables

$0 \leq \ell \leq 2T_f T_g - 1$ suffices: substitute $v_{i,\ell} \leftarrow v_i^\ell$ to get full rank (transposed) Vandermonde coefficient matrix

Idea of argument concluded

2. evaluate f/g at $(x_1,\ldots,x_n) \leftarrow (v_{1,\ell},\ldots,v_{n,\ell})$,
$\ell \leftarrow 0,1,2,\ldots,2T_f T_g - 1$, $v_{i,\ell}$ symbolic variables

Idea of argument concluded

2. evaluate f/g at $(x_1, \ldots, x_n) \leftarrow (v_{1,\ell}, \ldots, v_{n,\ell})$,
$\ell \leftarrow 0, 1, 2, \ldots, 2T_f T_g - 1$, $v_{i,\ell}$ symbolic variables

3. There are $r = T_f + T_g$ unknowns, the coefficients of Φ and Ψ
Select $1 \leq \theta \leq r-1$ full rank **rows** among the $2T_f T_g - 1$:

$$\Psi(v_{1,\ell_\theta}, \ldots, v_{n,\ell_\theta}) - (f/g)(v_{1,\ell_\theta}, \ldots, v_{n,\ell_\theta})\Phi(v_{1,\ell_\theta}, \ldots, v_{n,\ell_\theta}) = 0$$

Main idea:
all equations look the same for a new set of variables $v_{1,\ell_\theta}, \ldots, v_{n,\ell_\theta}$
Therefore, can use the first $r-1$ evaluations

Idea of argument concluded

2. evaluate f/g at $(x_1,\ldots,x_n) \leftarrow (v_{1,\ell},\ldots,v_{n,\ell})$,
$\ell \leftarrow 0,1,2,\ldots,2T_fT_g-1$, $v_{i,\ell}$ symbolic variables

3. There are $r = T_f + T_g$ unknowns, the coefficients of Φ and Ψ
Select $1 \le \theta \le r-1$ full rank **rows** among the $2T_fT_g-1$:

$$\Psi(v_{1,\ell_\theta},\ldots,v_{n,\ell_\theta}) - (f/g)(v_{1,\ell_\theta},\ldots,v_{n,\ell_\theta})\Phi(v_{1,\ell_\theta},\ldots,v_{n,\ell_\theta}) = 0$$

Main idea:
all equations look the same for a new set of variables $v_{1,\ell_\theta},\ldots,v_{n,\ell_\theta}$
Therefore, can use the first $r-1$ evaluations

4. Random evaluations at scalars by Schwartz-Zippel Lemma
(standard trick, but technically challenging for poles)

Remark for Arne: Vectors of functions

$$\left[\frac{f^{\langle 1\rangle}}{g},\ldots,\frac{f^{\langle s\rangle}}{g}\right] \in \mathsf{K}(x_1,\ldots,x_n)^s, \quad g \neq 0.$$

$L = T_g + \max_{1\le\sigma\le s} T_{f^{\langle\sigma\rangle}}$ evaluations suffice **at random points**

[Kaltofen, Yang, ISSAC'14]

$T_{f^{\langle\sigma\rangle}} =$ upper bound for number of terms in $f^{\langle\sigma\rangle}$

$T_g =$ upper bound for number of terms in g

Remark for Arne: Vectors of functions

$$\left[\frac{f^{\langle 1\rangle}}{g},\ldots,\frac{f^{\langle s\rangle}}{g}\right] \in \mathsf{K}(x_1,\ldots,x_n)^s, \quad g \neq 0.$$

$L = T_g + \max_{1\le\sigma\le s} T_{f^{\langle\sigma\rangle}}$ evaluations suffice **at random points**

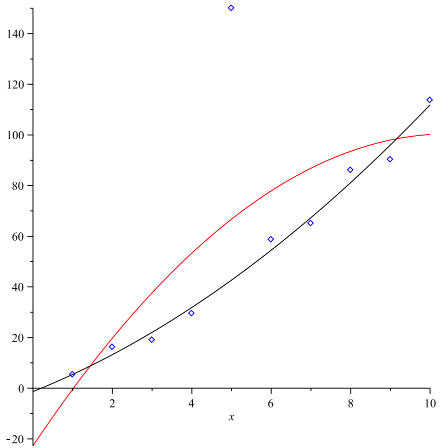
[Kaltofen, Yang, ISSAC'14]

$T_{f^{\langle\sigma\rangle}} =$ upper bound for number of terms in $f^{\langle\sigma\rangle}$
$T_g =$ upper bound for number of terms in g

Clément Pernet says $L = \lceil T_g/s \rceil + \max_{1\le\sigma\le s} T_{f^{\langle\sigma\rangle}}$ should suffice, at least for $n = 1$ and **dense** functions, i.e., rational vector reconstruction without or with errors, but randomness is still required

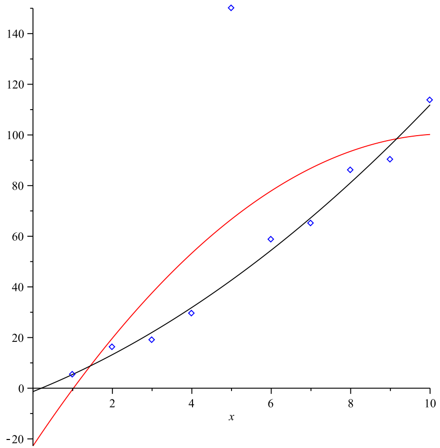
Note: errors are **not** in random locations: e.g., in transmissions, errors come in bursts

Outlier Example



How to identify the outlier?
Note again: sparse model is unknown

Outlier Example



How to identify the outlier?
Note again: sparse model is unknown
Outlier location and sparse interpolation can be related:
$\Longrightarrow$ **numeric** error correcting Reed-Solomon decoding

Curve/surface fitting problems

4 possible functions: polynomial, rational function;
univariate, multivariate

2+ representations: dense, **sparse** (several bases)

4 settings: exact (interpolation), with noise (least squares);
exact **with errors** (error correcting decoding),
with noise and **outliers**

2 different sparse interpolation algorithms: Zippel,
Prony/Blahut/Ben-Or&Tiwari

2 different Reed-Solomon decoders: Blahut, Berlekamp-Welch

Reed-Solomon error correcting codes

4 possible functions: polynomial, rational function;
univariate, multivariate

2 representations: dense, sparse

4 settings: exact (interpolation), with noise (least squares);
exact with errors (error correcting decoding),
with noise and outliers

2 different sparse interpolation algorithms: Zippel,
Prony/Blahut/Ben-Or&Tiwari

2 different Reed-Solomon decoders: Blahut, Berlekamp-Welch

Comer, Kaltofen, Pernet ISSAC 2012; Kaltofen, Pernet ISSAC 2014

4 different sparse models: polynomial, rational function; univariate, multivariate

4 settings: exact, with noise;
exact with errors, with noise and outliers

2 different sparse interpolation algorithms: Zippel,
Prony/Blahut/Ben-Or&Tiwari

Must correct outliers at the same time as determining sparse support;
need a new decoder!

2 different Reed-Solomon decoders: Blahut, Berlekamp-Welch

Prony 1795/Blahut 1979 Theorem

Idea #1:
Let $f(x) = c_1 x^{e_1}$
The **linear generator** for

$$a_i = f(\omega^i) = c_1 \omega^{e_1 i}, \quad i = 0, 1, 2, \ldots$$

is $\lambda - \omega^{e_1}$: $a_{i+1} - \omega^{e_1} a_i = 0$

Prony 1795/Blahut 1979 Theorem

Idea #1:
Let $f(x) = c_1 x^{e_1}$
The **linear generator** for

$$a_i = f(\omega^i) = c_1 \omega^{e_1 i}, \quad i = 0, 1, 2, \ldots$$

is $\lambda - \omega^{e_1}$: $a_{i+1} - \omega^{e_1} a_i = 0$

Let $g(x) = c_2 x^{e_2}$
The linear generator for

$$a_i + b_i = f(\omega^i) + g(\omega^i) = c_1 \omega^{e_1 i} + c_2 \omega^{e_2 i}, \quad i = 0, 1, 2, \ldots$$

is $\mathrm{LCM}(\lambda - \omega^{e_1}, \lambda - \omega^{e_2}) = (\lambda - \omega^{e_1})(\lambda - \omega^{e_2})$ for $\omega^{e_1} \neq \omega^{e_2}$

Basic Sparse Interpolation

$$f(x) = \sum_{j=1}^{T} c_j x^{e_j} \quad \text{[Prony: } x = \exp(y)\text{]}$$

Step 1: Compute $a_i = f(\omega^i)$ for $i = 0, \ldots, 2T - 1$

Step 2: Compute the linear generator $\Gamma(\lambda) = \prod_{j=1}^{T} (\lambda - \omega^{e_j})$ by the Berlekamp/Massey algorithm

Step 3: Compute exponents e_j from roots ω^{e_j} of Γ

Step 4: Compute c_j from transposed Vandermonde system

Note: need $2T$ values, not $\deg(f) + 1$ values

Sparse polynomial codes [Kaltofen,Pernet ISSAC 2014]

Let ω be a primitive m-th root of unity, $2T$ divides m

$$\begin{aligned} f_1(x) &= \frac{1}{T}\sum_{i=0}^{T-1} x^{2i\frac{m}{2T}} = \frac{1}{T}\cdot\frac{x^m-1}{x^{\frac{m}{T}}-1}, \\ f_2(x) &= -\frac{1}{T}\sum_{i=0}^{T-1} x^{(2i+1)\frac{m}{2T}} = -\frac{x^{\frac{m}{2T}}}{T}\cdot\frac{x^m-1}{x^{\frac{m}{T}}-1}. \end{aligned}$$

$$(f_1(\omega^i))_{i=0}^{m+2T-2} = \underbrace{0,\ldots,0}_{T-1},1,\underbrace{0,\ldots,0}_{T-1},\quad 1,\ldots,\quad 1,\underbrace{0,\ldots,0}_{T-1},1,\underbrace{0,\ldots,0}_{T-1}\,|\quad 1$$

$$(f_2(\omega^i))_{i=0}^{m+2T-2} = \underbrace{0,\ldots,0}_{T-1},1,\underbrace{0,\ldots,0}_{T-1},-1,\ldots-1,\underbrace{0,\ldots,0}_{T-1},1,\underbrace{0,\ldots,0}_{T-1}\,|-1$$

has only $\delta = m/(2T)$ differences: cannot decode from

$< m+2T = 2T(\frac{m}{2T}+1) = \boxed{2T(\delta+1)}$ elements with $\boxed{E = \frac{\delta}{2}}$ errors.

Decoding $2T(2E+1)$ elements [Comer,Kaltofen,Pernet'12]

$E+1$ *segments of* $2T$ *evaluations are error free and yield the same unique Prony/Blahut sparse interpolant: majority vote!*

Decoding $2T(2E+1)$ elements [Comer,Kaltofen,Pernet'12]

$E+1$ segments of $2T$ evaluations are error free and yield the same unique Prony/Blahut sparse interpolant: majority vote!

List-decoding $2T(E+1)$ elements [Comer,Kaltofen,Pernet'12]

Return a list of valid interpolants (with $\leq E$ errors) from each segment of $2T$ evaluations

One segment is error free, so the original sparse polynomial must be in the list

Sparse polynomial codes at real points

Let $\Phi, \Psi \in \mathbb{R}[x]$.
Φ and Ψ both have sparsity $\leq T$.
$(\xi_1, \beta_1), \ldots (\xi_{2T+2E}, \beta_{2T+2E})$ be distinct interpolation points
with $\boxed{\forall i \colon\ \xi_i > 0}$

Suppose $\Phi(\xi_i) = \beta_i$ for all $i \notin \lambda_i, \ldots, \lambda_k,\ k \leq E,$
$\Psi(\xi_j) = \beta_j$ for all $j \notin \mu_i, \ldots, \mu_\ell,\ \ell \leq E\colon$
no more than E interpolation errors for Φ and Ψ

Then $\Phi = \Psi$: $\Phi - \Psi$ has sparsity $\leq 2T$ and is zero at $2T$ distinct positive reals [Descartes's Sign Rule].

Sparse polynomial codes at real points

Let $\Phi, \Psi \in \mathbb{R}[x]$.
Φ and Ψ both have sparsity $\leq T$.
$(\xi_1, \beta_1), \ldots (\xi_{2T+2E}, \beta_{2T+2E})$ be distinct interpolation points
with $\boxed{\forall i\colon\ \xi_i > 0}$

Suppose $\Phi(\xi_i) = \beta_i$ for all $i \notin \lambda_i, \ldots, \lambda_k,\ k \leq E,$
$\Psi(\xi_j) = \beta_j$ for all $j \notin \mu_i, \ldots, \mu_\ell,\ \ell \leq E:$
no more than E interpolation errors for Φ and Ψ

Then $\Phi = \Psi$: $\Phi - \Psi$ has sparsity $\leq 2T$ and is zero at $2T$ distinct positive reals [Descartes's Sign Rule].

Sparse decoding: how to compute Φ fast?
Note: Our list-decoding algorithms yield unique solution from $\leq 2T(E+1)$ evaluations.

Better list-decoders [Kaltofen, Pernet ISSAC 2014]

Find an arithmetic progression $r+is, r \geq 0, s \geq 1$ such that $f(\omega^{r+is})$ are clean for all $i = 0, \ldots, 2T-1$
Interpolate the T-sparse polynomial $f(\omega^r x^s)$

Example 1: $E = 1$ and $4T - 1 < 2T(E+1)$ elements
Difficult case: error at location $2T$: $f(\omega^{2T-1})$

Use $r = 0, s = 2$: $f(\omega^0), f(\omega^2), \ldots, f(\omega^{2(T-1)})$

Better list-decoders [Kaltofen, Pernet ISSAC 2014]

Find an arithmetic progression $r+is, r \geq 0, s \geq 1$ such that $f(\omega^{r+is})$ are clean for all $i = 0, \ldots, 2T-1$
Interpolate the T-sparse polynomial $f(\omega^r x^s)$

Example 1: $E = 1$ and $4T-1 < 2T(E+1)$ elements
Difficult case: error at location $2T$: $f(\omega^{2T-1})$

Use $r = 0, s = 2$: $f(\omega^0), f(\omega^2), \ldots, f(\omega^{2(T-1)})$
Example 2 by computer: $T = 5, E = 10$: $f(\omega^i), i = 0, ..., \mathbf{73}$ suffice
$[2T(E+1) = \mathbf{110}]$

Better list-decoders [Kaltofen, Pernet ISSAC 2014]

Find an arithmetic progression $r+is, r \geq 0, s \geq 1$ such that $f(\omega^{r+is})$ are clean for all $i = 0, \ldots, 2T-1$
Interpolate the T-sparse polynomial $f(\omega^r x^s)$

Example 1: $E = 1$ and $4T-1 < 2T(E+1)$ elements
Difficult case: error at location $2T$: $f(\omega^{2T-1})$

Use $r = 0, s = 2$: $f(\omega^0), f(\omega^2), \ldots, f(\omega^{2(T-1)})$
Example 2 by computer: $T = 5, E = 10$: $f(\omega^i), i = 0, ..., \mathbf{73}$ suffice
$[2T(E+1) = \mathbf{110}]$

My phone call in April'14: *"Clément, you have designed an algorithm that has at least **cubic running time** in its input size; but hey, it's polynomial-time as required for list-decoders*"

The Szerekes and Erdős-Turán Conjectures 1936

$r(k,n)$ is the length of the longest subsequence of $1,2,3,\ldots,n$ that contains no k-term arithmetic progression (here: $k = 2T$)

$$\forall k \geq 3: \lim_{n\to\infty} \frac{r(k,n)}{n} = 0$$ [Szemeredi ICM 1974]

Szerekes's Conjecture:

$$r\left(k, \frac{(k-2)k^i + 1}{k-1}\right) = (k-1)^i$$

Disproved by Salem and Spencer in 1950, also needed in matrix multiplication algorithms.

The Szerekes and Erdős-Turán Conjectures 1936

$r(k,n)$ is the length of the longest subsequence of $1,2,3,\ldots,n$ that contains no k-term arithmetic progression (here: $k = 2T$)

$$\forall k \geq 3\colon \lim_{n\to\infty} \frac{r(k,n)}{n} = 0$$ [Szemeredi ICM 1974]

Szerekes's Conjecture:

$$r\left(k, \frac{(k-2)k^i+1}{k-1}\right) = (k-1)^i$$

Disproved by Salem and Spencer in 1950, also needed in matrix multiplication algorithms.

List-decoding for Chebyshev basis done with Andrew Arnold yesterday.

Multivariate Generalization [Kaltofen, June 2014]

$$f(x,y) = \sum_{j=1}^{T} c_j x^{d_j} y^{e_j}, \quad c_j \neq 0$$

The infinite 2-dimensional array

$$[f(\omega^i, \zeta^\ell)]_{i\geq 0, \ell \geq 0}$$

has an ideal of (scalar) linear generators

$$\Gamma = \text{Ideal-Product}_{j=1}^{T} \Big((x - \omega^{d_j})\mathsf{K}[x,y] + (y - \zeta^{e_j})\mathsf{K}[x,y] \Big)$$

with $\text{Set-of-Zeros}(\Gamma) = \{(\omega^{d_j}, \zeta^{e_j}) \mid j = 1, \ldots T\}$

Multivariate Generalization [Kaltofen, June 2014]

$$f(x,y) = \sum_{j=1}^{T} c_j x^{d_j} y^{e_j}, \quad c_j \neq 0$$

The infinite 2-dimensional array

$$[f(\omega^i, \zeta^\ell)]_{i\geq 0, \ell \geq 0}$$

has an ideal of (scalar) linear generators

$$\Gamma = \text{Ideal-Product}_{j=1}^{T}\Big((x-\omega^{d_j})\mathsf{K}[x,y] + (y-\zeta^{e_j})\mathsf{K}[x,y]\Big)$$

with Set-of-Zeros$(\Gamma) = \{(\omega^{d_j}, \zeta^{e_j}) \mid j = 1, \ldots T\}$

One can compute a triangular basis for Γ by
Shojiro Sakata's 1998 algorithm $\rightarrow$ Maple worksheet

Kaltofen and Yang ISSAC 2013, 2014

4 different sparse models: polynomial, rational function; univariate, multivariate

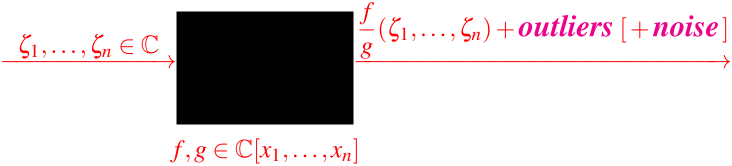
4 settings: exact, with noise;
exact with errors, with noise and outliers

2 different sparse interpolation algorithms: Zippel,
Prony/Blahut/Ben-Or&Tiwari

2 different Reed-Solomon decoders: Blahut, Berlekamp-Welch

A lucky coincidence:
Berlekamp-Welch decoding = rational function recovery
[Kaltofen and Pernet 2013; Boyer and Kaltofen SNC 2014]

Rational Recovery with Outliers [Kaltofen,Yang'13, '14]

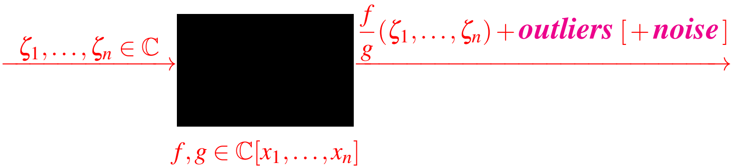


By sampling black box, compute sparse representation

$$\frac{\sum_{j=1}^{T_f} \tilde{a}_j x_1^{d_{j,1}} \cdots x_n^{d_{j,n}}}{\sum_{m=1}^{T_g} \tilde{b}_m x_1^{e_{m,1}} \cdots x_n^{e_{m,n}}} = \frac{\tilde{f}}{\tilde{g}}, \quad \tilde{a}_j \neq 0, \tilde{b}_m \neq 0$$

1 *Note:* Term exponents and outlier locations are **not** known.

Rational Recovery with Outliers [Kaltofen, Yang'13, '14]

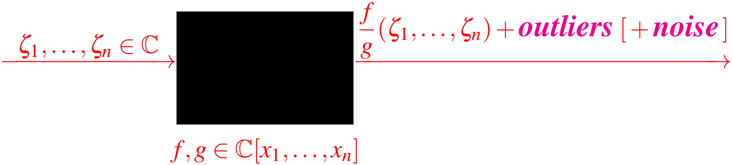


By sampling black box, compute sparse representation

$$\frac{\sum_{j=1}^{T_f} \tilde{a}_j x_1^{d_{j,1}} \cdots x_n^{d_{j,n}}}{\sum_{m=1}^{T_g} \tilde{b}_m x_1^{e_{m,1}} \cdots x_n^{e_{m,n}}} = \frac{\tilde{f}}{\tilde{g}}, \quad \tilde{a}_j \neq 0, \tilde{b}_m \neq 0$$

1 *Note:* Term exponents and outlier locations are **not** known.
Idea: compute $\frac{f_i(x_1,\ldots,x_i)\Lambda(x_1)}{g_i(x_1,\ldots,x_i)\Lambda(x_1)}$ à la Kaltofen, Yang, Zhi '07,
where $\Lambda(x_1) = (x_1 - \xi_{1,\lambda_1}) \cdots (x_1 - \xi_{1,\lambda_k})$ is the "error locator polyn."

Rational Recovery with Outliers [Kaltofen, Yang'13, '14]



By sampling black box, compute sparse representation

$$\frac{\sum_{j=1}^{T_f} \tilde{a}_j x_1^{d_{j,1}} \cdots x_n^{d_{j,n}}}{\sum_{m=1}^{T_g} \tilde{b}_m x_1^{e_{m,1}} \cdots x_n^{e_{m,n}}} = \frac{\tilde{f}}{\tilde{g}}, \quad \tilde{a}_j \neq 0, \tilde{b}_m \neq 0$$

1 *Note:* Term exponents and outlier locations are **not** known.
Idea: compute $\frac{f_i(x_1,\ldots,x_i)\Lambda(x_1)}{g_i(x_1,\ldots,x_i)\Lambda(x_1)}$ à la Kaltofen, Yang, Zhi '07,
where $\Lambda(x_1) = (x_1 - \xi_{1,\lambda_1}) \cdots (x_1 - \xi_{1,\lambda_k})$ is the "error locator polyn."
Possibly unreduced $\frac{f}{g}$, e.g, $\frac{x^{100} - y^{100}}{x - y}$:
??? *Candès-Tao or* Kaltofen-Nehring'11 for **univariate** base case

Experiments: Maple 16 on Intel i7 2.8GHz 8GB VAIO

Ex.	*Random Noise*	$\bar{d}_f, \bar{d}_g$	deg(f), deg(g)	T_f, T_g	n	$1/q$	E	N	*Time* (*secs.*)	Rel. Error
2	$10^{-5} \sim 10^{-3}$	5, 5	2, 2	3, 3	2	1/12	21	306	5.975	4.470e–8
3	$10^{-5} \sim 10^{-3}$	2, 5	1, 4	2, 4	3	1/15	13	561	13.56	4.704e–7
4	$10^{-6} \sim 10^{-4}$	8, 8	5, 2	10, 6	3	1/40	12	616	47.35	3.615e–6
5	$10^{-7} \sim 10^{-5}$	10 10	7, 7	10, 10	5	1/90	7	1508	197.7	5.090e–11
6	$10^{-7} \sim 10^{-5}$	15, 10	10, 3	15, 5	8	1/90	7	2423	273.7	7.401e–11
7	$10^{-7} \sim 10^{-5}$	10, 15	5, 13	4, 6	10	1/80	2	1289	24.68	8.091e–10
8	$10^{-7} \sim 10^{-5}$	25, 25	20, 20	7, 7	15	1/100	3	2890	137.5	2.902e–10
9	$10^{-8} \sim 10^{-6}$	35, 35	30, 30	6, 6	20	1/80	2	3881	230.4	5.495e–13
10	$10^{-8} \sim 10^{-6}$	45, 45	40, 40	6, 6	5	1/80	6	2080	219.1	3.688e–12
11	$10^{-8} \sim 10^{-6}$	85, 85	60, 60	7, 7	4	1/100	11	2787	1479.0	3.710e–13
12	$10^{-8} \sim 10^{-6}$	85, 85	80, 80	3, 3	5	1/30	4	1773	83.59	4.508e–12
13	$10^{-9} \sim 10^{-7}$	70, 0	40, 0	6, 1	15	1/70	2	2284	75.86	7.492e–18
14	$\mathbf{10^{-8} \sim 10^{-6}}$	**25, 25**	**20, 20**	**5, 5**	**102**	**1/80**	**1**	**10191**	**272.1**	**6.104e–12**

High Error Rate: ISSAC 2014

Ex.	*Random Noise*	*Outlier Error*	$\deg(f), \deg(g)$	t_f, t_g	n	$1/q$	N	*time*	*Rel. Error*
1	$10^{-6} \sim 10^{-4}$	$1 \sim 2$	3, 3	2, 3	2	**1/4**	203	2.371	2.83e–9
2	$10^{-7} \sim 10^{-5}$	$0.1 \sim 0.2$	10, 10	3, 5	2	**1/8**	558	14.80	3.16e–12
3	$10^{-7} \sim 10^{-5}$	$0.001 \sim 0.002$	10, 3	4, 3	3	**1/7**	822	15.09	2.92e–12
4	$10^{-7} \sim 10^{-5}$	$0.01 \sim 0.02$	5, 5	4, 4	5	**1/4**	960	10.58	4.72e–11
5	$10^{-8} \sim 10^{-6}$	$0.001 \sim 0.002$	10, 10	5, 4	7	**1/10**	1002	26.93	1.36e–13
6	$10^{-8} \sim 10^{-6}$	$0.1 \sim 0.2$	5, 8	1, 3	10	**1/5**	2010	37.58	2.25e–12
7	$10^{-6} \sim 10^{-4}$	$0.01 \sim 0.02$	10, 15	3, 3	4	**1/10**	1874	88.09	8.12e–12
8	$10^{-7} \sim 10^{-5}$	$0.01 \sim 0.02$	10, 10	3, 2	15	**1/5**	2786	44.66	6.32e–12
9	$10^{-8} \sim 10^{-6}$	$0.01 \sim 0.02$	8, 8	4, 3	30	**1/5**	4798	206.0	1.58e–13
10	$10^{-9} \sim 10^{-7}$	$0.01 \sim 0.02$	15, 15	3, 3	50	**1/4**	14717	2080	1.90e–16

High Error Rate: ISSAC 2014

Ex.	*Random Noise*	*Outlier Error*	$\deg(f), \deg(g)$	t_f, t_g	n	$1/q$	N	*time*	*Rel. Error*
1	$10^{-6} \sim 10^{-4}$	$1 \sim 2$	3, 3	2, 3	2	**1/4**	203	2.371	2.83e–9
2	$10^{-7} \sim 10^{-5}$	$0.1 \sim 0.2$	10, 10	3, 5	2	**1/8**	558	14.80	3.16e–12
3	$10^{-7} \sim 10^{-5}$	$0.001 \sim 0.002$	10, 3	4, 3	3	**1/7**	822	15.09	2.92e–12
4	$10^{-7} \sim 10^{-5}$	$0.01 \sim 0.02$	5, 5	4, 4	5	**1/4**	960	10.58	4.72e–11
5	$10^{-8} \sim 10^{-6}$	$0.001 \sim 0.002$	10, 10	5, 4	7	**1/10**	1002	26.93	1.36e–13
6	$10^{-8} \sim 10^{-6}$	$0.1 \sim 0.2$	5, 8	1, 3	10	**1/5**	2010	37.58	2.25e–12
7	$10^{-6} \sim 10^{-4}$	$0.01 \sim 0.02$	10, 15	3, 3	4	**1/10**	1874	88.09	8.12e–12
8	$10^{-7} \sim 10^{-5}$	$0.01 \sim 0.02$	10, 10	3, 2	15	**1/5**	2786	44.66	6.32e–12
9	$10^{-8} \sim 10^{-6}$	$0.01 \sim 0.02$	8, 8	4, 3	30	**1/5**	4798	206.0	1.58e–13
10	$10^{-9} \sim 10^{-7}$	$0.01 \sim 0.02$	15, 15	3, 3	50	**1/4**	14717	2080	1.90e–16

Idea: use Kaltofen-Pernet'13 dense univ. rat. fun. interp. with errors to locate errors only, and then Kaltofen-Yang-Zhi'07 on clean data, but at random points

Conclusion

What’s in the black box?
a complicated function that is noisily approximated
or a rational function that noisily evaluates?

Sparsity is a strong constraint

Error correcting coding is applicable to floats

Much remains to be done, e.g., reduce number of evaluations

Thank you!