Fast computation of minimal interpolation bases

Vincent Neiger [§,†,‡]

Claude-Pierre Jeannerod[§] Éric Schost[†] Gilles Villard[§]

[§]AriC, LIP, École Normale Supérieure de Lyon, France

[†]University of Waterloo, Ontario, Canada

[‡]Partially supported by the mobility grants *Explo'ra doc* from *Région Rhône-Alpes* / *Globalink Research Award - Inria* from *Mitacs & Inria* / *Programme Avenir Lyon Saint-Étienne*

JNCF, Cluny, France, November 2, 2015

Outline

Polynomial approximation

Hermite-Padé approximation

Input: $\mathbf{f} = (f_1, \ldots, f_m)$ polynomials over $\mathbb{K}$, order σ
Find $\mathbf{p} = (p_1, \ldots, p_m)$ polynomials such that

$$\begin{cases} p_1 f_1 + \cdots + p_m f_m \ = \ 0 \bmod X^\sigma \\ \text{minimal } \deg(\mathbf{p}) \end{cases}$$

Polynomial approximation

Hermite-Padé approximation

Input: $\mathbf{f} = (f_1, \ldots, f_m)$ polynomials over $\mathbb{K}$, order σ
Find $\mathbf{p} = (p_1, \ldots, p_m)$ polynomials such that

$$\begin{cases} p_1 f_1 + \cdots + p_m f_m \ = \ 0 \bmod X^\sigma \\ \text{minimal } \deg(\mathbf{p}) \end{cases}$$

M-Padé approximation (without multiplicities)

Input: $\mathbf{f} = (f_1, \ldots, f_m)$ polynomials over $\mathbb{K}$, points $x_1, \ldots, x_\sigma$
Find $\mathbf{p} = (p_1, \ldots, p_m)$ polynomials such that

$$\begin{cases} p_1(x_j) f_1(x_j) + \cdots + p_m(x_j) f_m(x_j) \ = \ 0 \text{ for all } j \\ \text{minimal } \deg(\mathbf{p}) \end{cases}$$

Same problem...

Define $\mathbf{p} \cdot \mathbf{f}$ in $\mathbb{K}^{1\times\sigma}$

- Hermite-Padé
  $\mathbf{p} \cdot \mathbf{f} = [\text{coefficients of } p_1 f_1 + \cdots + p_m f_m \text{ of degree } < \sigma]$
- M-Padé
  $\mathbf{p} \cdot \mathbf{f} = [\text{evaluations of } p_1 f_1 + \cdots + p_m f_m \text{ at points } x_1, \ldots, x_\sigma]$

$\rightsquigarrow$ Unified framework [Beckermann - Labahn, 2000]

- $\mathbf{p}$ interpolant for $\mathbf{f}$: $\mathbf{p} \cdot \mathbf{f} = 0$
- minimal $\deg(\mathbf{p})$

Same problem...

Define $\mathbf{p} \cdot \mathbf{f}$ in $\mathbb{K}^{1\times\sigma}$

- Hermite-Padé
  $\mathbf{p} \cdot \mathbf{f} = [\text{coefficients of } p_1 f_1 + \cdots + p_m f_m \text{ of degree } < \sigma]$
- M-Padé
  $\mathbf{p} \cdot \mathbf{f} = [\text{evaluations of } p_1 f_1 + \cdots + p_m f_m \text{ at points } x_1, \ldots, x_\sigma]$

$\rightsquigarrow$ Unified framework [Beckermann - Labahn, 2000]

- $\mathbf{p}$ interpolant for $\mathbf{f}$: $\mathbf{p} \cdot \mathbf{f} = 0$
- minimal $\deg(\mathbf{p})$ $\longrightarrow$ minimal $\deg_{\mathbf{w}}(\mathbf{p})$

Degree weights $\mathbf{w} = (w_1, \ldots, w_m)$

$$\deg_{\mathbf{w}}(\mathbf{p}) = \max(\deg(p_j) + w_j)$$

. . . different algorithms?

Hermite-Padé

Algorithm $\star$: small weights	Cost	Output
[Van Barel - Bultheel, 1991] ($\star$)	$\mathcal{O}(m^{(2)}\sigma^2)$	Basis
[Beckermann - Labahn, 1994]	$\mathcal{O}^{\sim}(m^{\omega}\sigma)$	Basis
[Zhou - Labahn, 2012] $\star$	$\mathcal{O}^{\sim}(m^{\omega-1}\sigma)$	Basis

General problem

Algorithm $\star$: small weights	Cost	Output
[Beckermann - Labahn, 1997/2000] ($\star$)	$\mathcal{O}(m^{(2)}\sigma^2)$	Basis
[Kötter; as in McEliece, 2003] ($\star$)	$\mathcal{O}(m^{(2)}\sigma^2)$	Basis

This talk

Algorithm

- solves the general problem
- for small weights
- cost bound $\tilde{\mathcal{O}}(m^{\omega-1}\sigma)$
- outputs a minimal basis

Focusing on M-Padé approximation without multiplicities

Outline

List-decoding: Sudan algorithm

given σ points $\{(x_1, y_1), \ldots, (x_\sigma, y_\sigma)\}$

f solution: $\deg f \leqslant k$ and $f(x_i) = y_i$ for $\geqslant \sigma - e$ points

[Sudan, 1997]

- Compute degree constraints m and b
- Interpolation step
  compute $Q(X, Y) = Q_0 + Q_1 Y + \cdots + Q_m Y^m$ such that
  - $Q_0, \ldots, Q_m$ have small weighted degree: $\deg Q_j < b - jk$
  - $Q(x_i, y_i) = 0$ for all points
- Root-finding step
  the solutions f are among the Y-roots of $Q(X, Y)$

Interpolation steps in related contexts

[Guruswami - Sudan, 1999]
List-decoding of Reed-Solomon codes,
further extends the error-correction bound

Compute $Q(X, Y) = Q_0 + Q_1 Y + \cdots + Q_m Y^m$ such that

- $Q_0, \ldots, Q_m$ have small weighted degree
- $Q(x_i, y_i) = 0$ with multiplicity μ for all points

Interpolation steps in related contexts

[Kötter - Vardy, 2003]
Soft-decision decoding of Reed-Solomon codes

$x_1, \ldots, x_n$ are not pairwise distinct
Compute $Q(X, Y) = Q_0 + Q_1 Y + \cdots + Q_m Y^m$ such that

- $Q_0, \ldots, Q_m$ have small weighted degree
- $Q(x_i, y_i) = 0$ with multiplicity μ_i for all points

Interpolation steps in related contexts

[Guruswami - Rudra, 2006]
List-decoding of folded Reed-Solomon codes:
extends the error-correction bound up to the information-theoretic limit

[Devet - Goldberg - Heninger, 2012]
Optimally robust Private Information Retrieval

Compute $Q(X, Y_1, \ldots, Y_s) = \sum_{(j_1,\ldots,j_s)\in\Gamma} Q_{j_1,\ldots,j_s} Y_1^{j_1} \cdots Y_s^{j_s}$ such that

- the $Q_{(j_1,\ldots,j_s)}$ have small weighted degree
- $Q(x_i, y_{i1}, \ldots, y_{is}) = 0$ with multiplicity μ for all points

Outline

M-Padé

M-Padé approximation

Input: $\mathbf{f} = (f_1, \ldots, f_m)$ polynomials, points $x_1, \ldots, x_\sigma$, weights $\mathbf{w}$
Find $\mathbf{p} = (p_1, \ldots, p_m)$ polynomials such that

$\mathbf{p}$ interpolant for $\mathbf{f}$:	$\mathbf{p} \cdot \mathbf{f} = 0$
$\mathbf{p}$ has minimal weighted-degree	$\deg_{\mathbf{w}}(\mathbf{p})$

where $\mathbf{p} \cdot \mathbf{f} = [\text{evaluations of } p_1 f_1 + \cdots + p_m f_m \text{ at points } x_1, \ldots, x_\sigma]$

and $\deg_{\mathbf{w}}(\mathbf{p}) = \max(\deg(p_j) + w_j)$

Iterative algorithm:

- cost quadratic in σ
- returns a basis of interpolants

Iterative algorithm [Beckermann-Labahn / Kötter]

1. $\mathbf{P} = \begin{bmatrix} -\mathbf{p}_1- \\ \vdots \\ -\mathbf{p}_m- \end{bmatrix} = \text{Identity in } \mathbb{K}[X]^{m\times m}$

2. *For i from 1 to σ:*

a. Compute evaluations $\begin{bmatrix} (\mathbf{p}_1 \cdot \mathbf{f})(x_i) \\ \vdots \\ (\mathbf{p}_m \cdot \mathbf{f})(x_i) \end{bmatrix} = (\mathbf{P} \cdot \mathbf{f})(x_i)$

b. Choose pivot π with smallest w_π such that $(\mathbf{p}_\pi \cdot \mathbf{f})(x_i) \neq 0$
Update pivot weight $w_\pi = w_\pi + 1$

c. Eliminate:

For $j \neq \pi$ do $\mathbf{p}_j = \mathbf{p}_j - \dfrac{(\mathbf{p}_j \cdot \mathbf{f})(x_i)}{(\mathbf{p}_\pi \cdot \mathbf{f})(x_i)}\mathbf{p}_\pi$ /* $\forall j \neq \pi,\ (\mathbf{p}_j \cdot \mathbf{f})(x_i) = 0$ */

$\mathbf{p}_\pi = (X - x_i)\mathbf{p}_\pi$ /* $(\mathbf{p}_\pi \cdot \mathbf{f})(x_i) = 0$ */

After i iterations: $\mathbf{P}$ basis of small interpolants for $(x_1, \ldots, x_i)$

M-Padé Iterative algorithm

Parameters: $\sigma = 8$ $m = 4$ $\mathbf{w} = [0, 2, 4, 6]$, base field $\mathbb{F}_{97}$

Input: $(24, 31, 15, 32, 83, 27, 20, 59)$ and $\mathbf{f} = (1, R, R^2, R^3)$

Iteration: $i = 1$ **Point:** $24, 31, 15, 32, 83, 27, 20, 59$

Weights

$$[0 \quad 2 \quad 4 \quad 6]$$

Basis

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Values

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 80 & 73 & 73 & 35 & 66 & 46 & 91 & 64 \\ 95 & 91 & 91 & 61 & 88 & 79 & 36 & 22 \\ 34 & 47 & 47 & 1 & 85 & 45 & 75 & 50 \end{bmatrix}$$

M-Padé Iterative algorithm

Parameters: $\sigma = 8$ $m = 4$ $\mathbf{w} = [0, 2, 4, 6]$, base field $\mathbb{F}_{97}$

Input: $(24, 31, 15, 32, 83, 27, 20, 59)$ and $\mathbf{f} = (1, R, R^2, R^3)$

Iteration: $i = 1$ **Point:** $24, 31, 15, 32, 83, 27, 20, 59$

Weights

$$[0 \quad 2 \quad 4 \quad 6]$$

Basis

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Values

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 80 & 73 & 73 & 35 & 66 & 46 & 91 & 64 \\ 95 & 91 & 91 & 61 & 88 & 79 & 36 & 22 \\ 34 & 47 & 47 & 1 & 85 & 45 & 75 & 50 \end{bmatrix}$$

M-Padé Iterative algorithm

Parameters: $\sigma = 8$ $m = 4$ $\mathbf{w} = [0, 2, 4, 6]$, base field $\mathbb{F}_{97}$

Input: $(24, 31, 15, 32, 83, 27, 20, 59)$ and $\mathbf{f} = (1, R, R^2, R^3)$

Iteration: $i = 1$ **Point:** $24, 31, 15, 32, 83, 27, 20, 59$

Weights $[0 \quad 2 \quad 4 \quad 6]$

Basis

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 17 & 1 & 0 & 0 \\ 2 & 0 & 1 & 0 \\ 63 & 0 & 0 & 1 \end{bmatrix}$$

Values

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 90 & 90 & 52 & 83 & 63 & 11 & 81 \\ 0 & 93 & 93 & 63 & 90 & 81 & 38 & 24 \\ 0 & 13 & 13 & 64 & 51 & 11 & 41 & 16 \end{bmatrix}$$

M-Padé Iterative algorithm

Parameters: $\sigma = 8$ $m = 4$ $\mathbf{w} = [0, 2, 4, 6]$, base field $\mathbb{F}_{97}$

Input: $(24, 31, 15, 32, 83, 27, 20, 59)$ and $\mathbf{f} = (1, R, R^2, R^3)$

Iteration: $i = 1$ **Point:** $24, 31, 15, 32, 83, 27, 20, 59$

Weights $[1 \quad 2 \quad 4 \quad 6]$

Basis

$$\begin{bmatrix} X + 73 & 0 & 0 & 0 \\ 17 & 1 & 0 & 0 \\ 2 & 0 & 1 & 0 \\ 63 & 0 & 0 & 1 \end{bmatrix}$$

Values

$$\begin{bmatrix} 0 & 7 & 88 & 8 & 59 & 3 & 93 & 35 \\ 0 & 90 & 90 & 52 & 83 & 63 & 11 & 81 \\ 0 & 93 & 93 & 63 & 90 & 81 & 38 & 24 \\ 0 & 13 & 13 & 64 & 51 & 11 & 41 & 16 \end{bmatrix}$$

M-Padé Iterative algorithm

Parameters: $\sigma = 8$ $m = 4$ $\mathbf{w} = [0, 2, 4, 6]$, base field $\mathbb{F}_{97}$

Input: $(24, 31, 15, 32, 83, 27, 20, 59)$ and $\mathbf{f} = (1, R, R^2, R^3)$

Iteration: $i = 2$ **Point:** $24, 31, 15, 32, 83, 27, 20, 59$

Weights

$$[1 \quad 2 \quad 4 \quad 6]$$

Basis

$$\begin{bmatrix} X + 73 & 0 & 0 & 0 \\ 17 & 1 & 0 & 0 \\ 2 & 0 & 1 & 0 \\ 63 & 0 & 0 & 1 \end{bmatrix}$$

Values

$$\begin{bmatrix} 0 & 7 & 88 & 8 & 59 & 3 & 93 & 35 \\ 0 & 90 & 90 & 52 & 83 & 63 & 11 & 81 \\ 0 & 93 & 93 & 63 & 90 & 81 & 38 & 24 \\ 0 & 13 & 13 & 64 & 51 & 11 & 41 & 16 \end{bmatrix}$$

M-Padé Iterative algorithm

Parameters: $\sigma = 8$ $m = 4$ $\mathbf{w} = [0, 2, 4, 6]$, base field $\mathbb{F}_{97}$

Input: $(24, 31, 15, 32, 83, 27, 20, 59)$ and $\mathbf{f} = (1, R, R^2, R^3)$

Iteration: $i = 2$ **Point:** $24, 31, 15, 32, 83, 27, 20, 59$

Weights

$$[1 \quad 2 \quad 4 \quad 6]$$

Basis

$$\begin{bmatrix} X + 73 & 0 & 0 & 0 \\ X + 90 & 1 & 0 & 0 \\ 56X + 16 & 0 & 1 & 0 \\ 12X + 66 & 0 & 0 & 1 \end{bmatrix}$$

Values

$$\begin{bmatrix} 0 & 7 & 88 & 8 & 59 & 3 & 93 & 35 \\ 0 & 0 & 81 & 60 & 45 & 66 & 7 & 19 \\ 0 & 0 & 74 & 26 & 96 & 55 & 8 & 44 \\ 0 & 0 & 2 & 63 & 80 & 47 & 90 & 48 \end{bmatrix}$$

M-Padé Iterative algorithm

Parameters: $\sigma = 8$ $m = 4$ $\mathbf{w} = [0, 2, 4, 6]$, base field $\mathbb{F}_{97}$

Input: $(24, 31, 15, 32, 83, 27, 20, 59)$ and $\mathbf{f} = (1, R, R^2, R^3)$

Iteration: $i = 2$ **Point:** $24, 31, 15, 32, 83, 27, 20, 59$

Weights

$$[2 \quad 2 \quad 4 \quad 6]$$

Basis

$$\begin{bmatrix} X^2 + 42X + 65 & 0 & 0 & 0 \\ X + 90 & 1 & 0 & 0 \\ 56X + 16 & 0 & 1 & 0 \\ 12X + 66 & 0 & 0 & 1 \end{bmatrix}$$

Values

$$\begin{bmatrix} 0 & 0 & 47 & 8 & 61 & 85 & 44 & 10 \\ 0 & 0 & 81 & 60 & 45 & 66 & 7 & 19 \\ 0 & 0 & 74 & 26 & 96 & 55 & 8 & 44 \\ 0 & 0 & 2 & 63 & 80 & 47 & 90 & 48 \end{bmatrix}$$

M-Padé Iterative algorithm

Parameters: $\sigma = 8$ $m = 4$ $\mathbf{w} = [0, 2, 4, 6]$, base field $\mathbb{F}_{97}$

Input: $(24, 31, 15, 32, 83, 27, 20, 59)$ and $\mathbf{f} = (1, R, R^2, R^3)$

Iteration: $i = 3$ **Point:** $24, 31, 15, 32, 83, 27, 20, 59$

Weights

$$[2 \quad 2 \quad 4 \quad 6]$$

Basis

$$\begin{bmatrix} X^2 + 42X + 65 & 0 & 0 & 0 \\ X + 90 & 1 & 0 & 0 \\ 56X + 16 & 0 & 1 & 0 \\ 12X + 66 & 0 & 0 & 1 \end{bmatrix}$$

Values

$$\begin{bmatrix} 0 & 0 & 47 & 8 & 61 & 85 & 44 & 10 \\ 0 & 0 & 81 & 60 & 45 & 66 & 7 & 19 \\ 0 & 0 & 74 & 26 & 96 & 55 & 8 & 44 \\ 0 & 0 & 2 & 63 & 80 & 47 & 90 & 48 \end{bmatrix}$$

M-Padé Iterative algorithm

Parameters: $\sigma = 8$ $m = 4$ $\mathbf{w} = [0, 2, 4, 6]$, base field $\mathbb{F}_{97}$

Input: $(24, 31, 15, 32, 83, 27, 20, 59)$ and $\mathbf{f} = (1, R, R^2, R^3)$

Iteration: $i = 3$ **Point:** $24, 31, 15, 32, 83, 27, 20, 59$

Weights

$$[3 \quad 2 \quad 4 \quad 6]$$

Basis

$$\begin{bmatrix} X^3 + 27X^2 + 17X + 92 & 0 & 0 & 0 \\ 54X^2 + 38X + 11 & 1 & 0 & 0 \\ 17X^2 + 91X + 54 & 0 & 1 & 0 \\ 66X^2 + 68X + 88 & 0 & 0 & 1 \end{bmatrix}$$

Values

$$\begin{bmatrix} 0 & 0 & 0 & 39 & 74 & 50 & 26 & 52 \\ 0 & 0 & 0 & 7 & 41 & 0 & 55 & 74 \\ 0 & 0 & 0 & 65 & 66 & 45 & 77 & 20 \\ 0 & 0 & 0 & 9 & 32 & 31 & 84 & 29 \end{bmatrix}$$

M-Padé Iterative algorithm

Parameters: $\sigma = 8$ $m = 4$ $\mathbf{w} = [0, 2, 4, 6]$, base field $\mathbb{F}_{97}$

Input: $(24, 31, 15, 32, 83, 27, 20, 59)$ and $\mathbf{f} = (1, R, R^2, R^3)$

Iteration: $i = 4$ **Point:** $24, 31, 15, 32, 83, 27, 20, 59$

Weights

$$[3 \quad 2 \quad 4 \quad 6]$$

Basis

$$\begin{bmatrix} X^3 + 27X^2 + 17X + 92 & 0 & 0 & 0 \\ 54X^2 + 38X + 11 & 1 & 0 & 0 \\ 17X^2 + 91X + 54 & 0 & 1 & 0 \\ 66X^2 + 68X + 88 & 0 & 0 & 1 \end{bmatrix}$$

Values

$$\begin{bmatrix} 0 & 0 & 0 & 39 & 74 & 50 & 26 & 52 \\ 0 & 0 & 0 & 7 & 41 & 0 & 55 & 74 \\ 0 & 0 & 0 & 65 & 66 & 45 & 77 & 20 \\ 0 & 0 & 0 & 9 & 32 & 31 & 84 & 29 \end{bmatrix}$$

M-Padé Iterative algorithm

Parameters: $\sigma = 8$ $m = 4$ $\mathbf{w} = [0, 2, 4, 6]$, base field $\mathbb{F}_{97}$

Input: $(24, 31, 15, 32, 83, 27, 20, 59)$ and $\mathbf{f} = (1, R, R^2, R^3)$

Iteration: $i = 4$ **Point:** $24, 31, 15, 32, 83, 27, 20, 59$

Weights

$$[3 \quad 3 \quad 4 \quad 6]$$

Basis

$$\begin{bmatrix} X^3 + 31X^2 + 27X + 3 & 36 & 0 & 0 \\ 54X^3 + 56X^2 + 56X + 36 & X + 65 & 0 & 0 \\ 56X^2 + 43X + 35 & 60 & 1 & 0 \\ 52X^2 + 33X + 60 & 68 & 0 & 1 \end{bmatrix}$$

Values

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 95 & 50 & 66 & 0 \\ 0 & 0 & 0 & 0 & 54 & 0 & 19 & 58 \\ 0 & 0 & 0 & 0 & 4 & 45 & 79 & 95 \\ 0 & 0 & 0 & 0 & 7 & 31 & 41 & 17 \end{bmatrix}$$

M-Padé Iterative algorithm

Parameters: $\sigma = 8$ $m = 4$ $\mathbf{w} = [0, 2, 4, 6]$, base field $\mathbb{F}_{97}$

Input: $(24, 31, 15, 32, 83, 27, 20, 59)$ and $\mathbf{f} = (1, R, R^2, R^3)$

Iteration: $i = 5$ **Point:** $24, 31, 15, 32, 83, 27, 20, 59$

Weights

$$[4 \quad 3 \quad 4 \quad 6]$$

Basis

$$\begin{bmatrix} X^4 + 45X^3 + 73X^2 + 90X + 42 & 36X + 19 & 0 & 0 \\ 81X^3 + 20X^2 + 9X + 20 & X + 67 & 0 & 0 \\ 2X^3 + 21X^2 + 41 & 35 & 1 & 0 \\ 52X^3 + 15X^2 + 79X + 22 & 0 & 0 & 1 \end{bmatrix}$$

Values

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 13 & 13 & 0 \\ 0 & 0 & 0 & 0 & 0 & 89 & 55 & 58 \\ 0 & 0 & 0 & 0 & 0 & 48 & 17 & 95 \\ 0 & 0 & 0 & 0 & 0 & 12 & 78 & 17 \end{bmatrix}$$

M-Padé Iterative algorithm

Parameters: $\sigma = 8$ $m = 4$ $\mathbf{w} = [0, 2, 4, 6]$, base field $\mathbb{F}_{97}$

Input: $(24, 31, 15, 32, 83, 27, 20, 59)$ and $\mathbf{f} = (1, R, R^2, R^3)$

Iteration: $i = 6$ **Point:** $24, 31, 15, 32, 83, 27, 20, 59$

Weights

$$[4 \quad 4 \quad 4 \quad 6]$$

Basis

$$\begin{bmatrix} X^4 + 19X^3 + 57X^2 + 44X + 26 & 74X + 43 & 0 & 0 \\ 81X^4 + 64X^3 + 51X^2 + 68X + 42 & X^2 + 40X + 34 & 0 & 0 \\ 3X^3 + 44X^2 + 54X + 64 & 6X + 49 & 1 & 0 \\ 28X^3 + 45X^2 + 44X + 52 & 50X + 52 & 0 & 1 \end{bmatrix}$$

Values

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 66 & 70 \\ 0 & 0 & 0 & 0 & 0 & 0 & 3 & 13 \\ 0 & 0 & 0 & 0 & 0 & 0 & 56 & 55 \\ 0 & 0 & 0 & 0 & 0 & 0 & 15 & 7 \end{bmatrix}$$

M-Padé Iterative algorithm

Parameters: $\sigma = 8$ $m = 4$ $\mathbf{w} = [0, 2, 4, 6]$, base field $\mathbb{F}_{97}$

Input: $(24, 31, 15, 32, 83, 27, 20, 59)$ and $\mathbf{f} = (1, R, R^2, R^3)$

Iteration: $i = 7$ **Point:** $24, 31, 15, 32, 83, 27, 20, 59$

Weights

$$[5 \quad 4 \quad 4 \quad 6]$$

Basis

$$\begin{bmatrix} X^5+96X^4+65X^3+68X^2+19X+62 & 74X^2+18X+13 & 0 & 0 \\ 6X^4+94X^3+44X^2+66X+32 & X^2+19X+10 & 0 & 0 \\ 55X^4+78X^3+75X^2+49X+39 & 2X+86 & 1 & 0 \\ 13X^4+81X^3+10X^2+34X+2 & 42X+29 & 0 & 1 \end{bmatrix}$$

Values

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 14 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 25 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 44 \end{bmatrix}$$

M-Padé Iterative algorithm

Parameters: $\sigma = 8$ $m = 4$ $\mathbf{w} = [0, 2, 4, 6]$, base field $\mathbb{F}_{97}$

Input: $(24, 31, 15, 32, 83, 27, 20, 59)$ and $\mathbf{f} = (1, R, R^2, R^3)$

Iteration: $i = 8$ **Point:** $24, 31, 15, 32, 83, 27, 20, 59$

Weights

$$[5 \quad 5 \quad 4 \quad 6]$$

Basis

$$\begin{bmatrix} X^5+12X^4+10X^3+34X^2+65X+2 & 60X^2+43X+67 & 0 & 0 \\ 6X^5+31X^4+27X^3+89X^2+18X+52 & X^3+57X^2+53X+89 & 0 & 0 \\ 2X^4+56X^3+42X^2+48X+15 & 72X^2+12X+30 & 1 & 0 \\ 40X^4+19X^3+14X^2+40X+49 & 53X^2+79X+74 & 0 & 1 \end{bmatrix}$$

Values

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Minimal interpolation bases

interpolation basis =
basis of the module of interpolants

$$\left\{\mathbf{p} = (p_1, \ldots, p_m) \in \mathbb{K}[X]^m \;\; / \;\; \mathbf{p} \cdot \mathbf{f} = 0\right\}$$

rank $m \rightsquigarrow$ matrix $\mathbf{P} = \begin{bmatrix} -\mathbf{p}_1- \\ \vdots \\ -\mathbf{p}_m- \end{bmatrix}$

w-*minimal* =
the tuple $(\deg_{\mathbf{w}}(\mathbf{p}_1), \ldots, \deg_{\mathbf{w}}(\mathbf{p}_m))$ is minimal

Example: an interpolant $\mathbf{p}$ such that $\deg(\mathbf{p}) \leqslant \sigma/m$ can be found in a **0**-minimal interpolation basis

Divide-and-conquer algorithm

Relying on fast polynomial matrix multiplication

Following [Beckermann - Labahn, 1994] for Hermite-Padé:

- $\mathbf{P}^{(1)}$ for first $\sigma/2$ points, $\mathbf{f}$, and weights $\mathbf{w}$
- $\mathbf{P}^{(2)}$ for last $\sigma/2$ points, updated $\mathbf{f}$, and updated weights $\mathbf{w}'$
- return $\mathbf{P} = \mathbf{P}^{(2)}\mathbf{P}^{(1)}$

Then $\mathbf{P}$ is a $\mathbf{w}$-minimal interpolation basis

Obstacle: degrees in $\mathbf{P}^{(1)}$ and $\mathbf{P}^{(2)}$ up to $\sigma/2$
$\rightsquigarrow$ $\mathbf{P} = \mathbf{P}^{(2)}\mathbf{P}^{(1)}$ computed in $\tilde{\mathcal{O}}(m^\omega \sigma)$

Solution for fast Hermite-Padé

Simultaneous Hermite-Padé: n equations in degree σ/n

Algorithm $\star$: small weights	Cost	Output
[Beckermann - Labahn, 1994]	$\mathcal{O}^{\sim}(m^{\omega}\sigma)$	Basis
[Giorgi - Jeannerod - Villard, 2003]	$\mathcal{O}^{\sim}(m^{\omega}\sigma/n)$	Basis

degrees in $\mathbf{P}^{(1)}$ and $\mathbf{P}^{(2)}$ up to $\leqslant \sigma/(2n)$
$\rightsquigarrow \mathbf{P} = \mathbf{P}^{(2)}\mathbf{P}^{(1)}$ computed in $\mathcal{O}^{\sim}(m^{\omega}\sigma/n)$

Solution for fast Hermite-Padé

Simultaneous Hermite-Padé: n equations in degree σ/n

Algorithm $\star$: small weights	Cost	Output
[Beckermann - Labahn, 1994]	$\tilde{\mathcal{O}}(m^\omega \sigma)$	Basis
[Giorgi - Jeannerod - Villard, 2003]	$\tilde{\mathcal{O}}(m^\omega \sigma/n)$	Basis
[Storjohann, 2006] $\star$	$\tilde{\mathcal{O}}(m^{\omega-1} \sigma)$	Partial basis
[Zhou - Labahn, 2012] $\star$	$\tilde{\mathcal{O}}(m^{\omega-1} \sigma)$	Basis

degrees in $\mathbf{P}^{(1)}$ and $\mathbf{P}^{(2)}$ up to $\leqslant \sigma/(2n)$
$\rightsquigarrow \mathbf{P} = \mathbf{P}^{(2)}\mathbf{P}^{(1)}$ computed in $\tilde{\mathcal{O}}(m^\omega \sigma/n)$

Obstacle solved: under $\star$, Storjohann's first transformation
one equation in degree σ $\longrightarrow$ m equations in degree σ/m

Unclear to me how to proceed similarly for the general problem

Ingredient 1: controlling the weights (1/3)

P with $m \times m$ entries of degree $\leqslant \sigma$: size up to $\Theta(m^2\sigma)$

$\rightsquigarrow$ compromises our target cost $\tilde{\mathcal{O}}(m^{\omega-1}\sigma)$

Ingredient 1: controlling the weights (1/3)

P with $m \times m$ entries of degree $\leqslant \sigma$: size up to $\Theta(m^2\sigma)$
$\rightsquigarrow$ compromises our target cost $\tilde{\mathcal{O}}(m^{\omega-1}\sigma)$

Assumption $\star$: weights **w** satisfy $|\mathbf{w}| = w_1 + \cdots + w_m \leqslant \sigma$

$\Rightarrow$ **P** has sum of row degrees $\leqslant 2\sigma$ (in particular, size $\mathcal{O}(m\sigma)$)

Example of degrees in **P** ($m = 4, \sigma = 16$)

$\mathbf{w} = (0,0,0,0)$

$$\begin{bmatrix} 4 & 3 & 3 & 3 \\ 4 & 4 & 3 & 3 \\ 4 & 4 & 4 & 3 \\ 4 & 4 & 4 & 4 \end{bmatrix} \quad \begin{bmatrix} 7 & 6 & 1 & 6 \\ 0 & 1 & & 0 \\ & & 2 & \\ 6 & 6 & 1 & 6 \end{bmatrix}$$

sum: $16 = \sigma$

$\mathbf{w} = (0,2,4,6)$

$$\begin{bmatrix} 7^7 & 4^6 & 2^6 & 0^6 \\ 7^7 & 5^7 & 2^6 & 0^6 \\ 7^7 & 5^7 & 3^7 & 0^6 \\ 7^7 & 5^7 & 3^7 & 1^7 \end{bmatrix} \quad \begin{bmatrix} 8^8 & 5^7 & 1^5 & \\ 8^8 & 6^8 & 1^5 & \\ & & 2^6 & \\ 0^0 & 1^3 & & 0^6 \end{bmatrix}$$

sum: $28 = \sigma + |\mathbf{w}|$

Ingredient 1: controlling the weights (2/3)

Divide-and-conquer algorithm:

- $\mathbf{P}^{(1)}$ for first $\sigma/2$ points, $\mathbf{f}$, and weights $\mathbf{w}$
- $\mathbf{P}^{(2)}$ for last $\sigma/2$ points, updated $\mathbf{f}$, and updated weights $\mathbf{w}'$
- return $\mathbf{P} = \mathbf{P}^{(2)}\mathbf{P}^{(1)}$

Under $\star$, fast multiplication with average row degree σ/m

$$\mathbf{P} = \mathbf{P}^{(2)}\mathbf{P}^{(1)} \text{ is computed in } \tilde{\mathcal{O}}(m^{\omega-1}\sigma)$$

Ingredient 1: controlling the weights (2/3)

Divide-and-conquer algorithm:

- $\mathbf{P}^{(1)}$ for first $\sigma/2$ points, $\mathbf{f}$, and weights $\mathbf{w}$
- $\mathbf{P}^{(2)}$ for last $\sigma/2$ points, updated $\mathbf{f}$, and updated weights $\mathbf{w}'$
- return $\mathbf{P} = \mathbf{P}^{(2)}\mathbf{P}^{(1)}$

Under $\star$, fast multiplication with average row degree σ/m

$$\mathbf{P} = \mathbf{P}^{(2)}\mathbf{P}^{(1)} \text{ is computed in } \tilde{\mathcal{O}}(m^{\omega-1}\sigma)$$

Obstacle: in recursive calls, σ becomes $\sigma/2$ but $\mathbf{w}$ is unchanged

$\Rightarrow$ need to preserve the assumption $\star$: $w_1 + \cdots + w_m \leqslant \sigma$

- Compute a minimal basis for $\mathbf{0}$
- Change of weights: recover a minimal basis for $\mathbf{w}$

Ingredient 1: controlling the weights (3/3)

Change of weights

Input: $\mathbf{0}$-minimal basis $\mathbf{P}$, weights $\mathbf{w}$
Output: $\mathbf{w}$-minimal basis $\mathbf{R}$
Cost: $\tilde{\mathcal{O}}(m^{\omega-1}(|\mathbf{d}| + |\mathbf{w}|))$, where $\mathbf{d}$ = row degrees of $\mathbf{P}$

- $\mathbf{R}$ and $\mathbf{P}$ bases: $\mathbf{R} = \mathbf{U}\mathbf{P}$ for some unimodular $\mathbf{U}$
- $\mathbf{P}$ is $\mathbf{0}$-minimal $\Rightarrow$ $\mathbf{U}$ has small degree:

$$\mathbf{d}\text{-weighted row degrees of } \mathbf{U} \leqslant \text{row degrees of } \mathbf{R}$$

Algorithm: Compute $\mathbf{U}, \mathbf{R}$ via a $\mathbf{d}|\mathbf{w}$-minimal nullspace basis

$$\begin{array}{c|c} \mathbf{d} & \mathbf{w} \\ \hline \mathbf{U} & \mathbf{R} \end{array} \begin{bmatrix} \mathbf{P} \\ \hline -\mathbf{Id} \end{bmatrix} = \mathbf{0}$$

[Zhou - Labahn - Storjohann, 2012]

Ingredient 2:

Divide-and-conquer algorithm for $\mathbf{w} = \mathbf{0}$:

- $\mathbf{P}^{(1)}$ for first $\sigma/2$ points, $\mathbf{f}$, and weights $\mathbf{0}$
- $\mathbf{P}^{(2)}$ for last $\sigma/2$ points, updated $\mathbf{f}$, and weights $\mathbf{0}$
- $\mathbf{R}^{(2)} =$ Change weights of $\mathbf{P}^{(2)}$ to $\deg(\mathbf{P}^{(1)})$
- return $\mathbf{P} = \mathbf{R}^{(2)}\mathbf{P}^{(1)}$

Fast multiplication with average row degree σ/m:
$\mathbf{P} = \mathbf{P}^{(2)}\mathbf{P}^{(1)}$ is computed in $\tilde{\mathcal{O}}(m^{\omega-1}\sigma)$

Ingredient 2: linearizing at small orders (1/2)

Divide-and-conquer algorithm for $\mathbf{w} = \mathbf{0}$:

- $\mathbf{P}^{(1)}$ for first $\sigma/2$ points, $\mathbf{f}$, and weights $\mathbf{0}$
- $\mathbf{P}^{(2)}$ for last $\sigma/2$ points, updated $\mathbf{f}$, and weights $\mathbf{0}$
- $\mathbf{R}^{(2)} =$ Change weights of $\mathbf{P}^{(2)}$ to $\deg(\mathbf{P}^{(1)})$
- return $\mathbf{P} = \mathbf{R}^{(2)}\mathbf{P}^{(1)}$

Fast multiplication with average row degree σ/m:
$\mathbf{P} = \mathbf{P}^{(2)}\mathbf{P}^{(1)}$ is computed in $\tilde{\mathcal{O}}(m^{\omega-1}\sigma)$ ... if $\sigma \geqslant m$

Base case: $\sigma = m$
Target cost: $\tilde{\mathcal{O}}(m^{\omega})$

sum of row degrees $\leqslant m \Rightarrow$ average degree in $\mathbf{P}$ is $\leqslant 1$
$\rightsquigarrow$ linearization in degree $\sigma = m$

Ingredient 2: linearizing at small orders (2/2)

minimal interpolation basis $\longleftrightarrow$ minimal linear relations between rows of $\mathcal{K}$

$$\mathcal{K} = \begin{bmatrix} \mathbf{Id} \cdot \mathbf{f} \\ X\mathbf{Id} \cdot \mathbf{f} \\ X^2\mathbf{Id} \cdot \mathbf{f} \\ \vdots \\ X^m\mathbf{Id} \cdot \mathbf{f} \end{bmatrix} = \begin{bmatrix} \mathbf{E} \\ \mathbf{ED} \\ \mathbf{ED}^2 \\ \vdots \\ \mathbf{ED}^m \end{bmatrix} \quad \text{where} \quad \mathbf{E} = \mathbf{Id} \cdot \mathbf{f} = \begin{bmatrix} f_1(x_1) & \cdots & f_1(x_m) \\ \cdots & & \\ f_m(x_1) & \cdots & f_m(x_m) \end{bmatrix}$$

$$\mathbf{D} = \mathrm{Diag}(x_1, \ldots, x_m)$$

Ingredient 2: linearizing at small orders (2/2)

minimal interpolation basis $\longleftrightarrow$ minimal linear relations between rows of $\mathcal{K}$

$$\mathcal{K} = \begin{bmatrix} \mathbf{Id} \cdot \mathbf{f} \\ X\mathbf{Id} \cdot \mathbf{f} \\ X^2\mathbf{Id} \cdot \mathbf{f} \\ \vdots \\ X^m\mathbf{Id} \cdot \mathbf{f} \end{bmatrix} = \begin{bmatrix} \mathbf{E} \\ \mathbf{ED} \\ \mathbf{ED}^2 \\ \vdots \\ \mathbf{ED}^m \end{bmatrix} \quad \text{where} \quad \mathbf{E} = \mathbf{Id} \cdot \mathbf{f} = \begin{bmatrix} f_1(x_1) & \cdots & f_1(x_m) \\ \cdots & & \\ f_m(x_1) & \cdots & f_m(x_m) \end{bmatrix}$$

$$\mathbf{D} = \mathrm{Diag}(x_1, \ldots, x_m)$$

- Find the first $\mathrm{rank}(\mathcal{K})$ linearly independent rows
  $\mathcal{O}(m^\omega \log m)$
- Compute minimal linear relations between rows
  $\mathcal{O}(m^\omega)$

Divide-and-conquer algorithm

Algorithm (Minimal interpolation basis, $\mathbf{w} = \mathbf{0}$)

- If $\sigma \leqslant m$:
  - compute $\mathbf{P}$ by linearization
- Else:
  - $\mathbf{P}^{(1)}$ for first $\sigma/2$ points, $\mathbf{f}$, and weights $\mathbf{0}$
  - $\mathbf{P}^{(2)}$ for last $\sigma/2$ points, updated $\mathbf{f}$, and weights $\mathbf{0}$
  - $\mathbf{R}^{(2)} =$ Change weights of $\mathbf{P}^{(2)}$ to $\deg(\mathbf{P}^{(1)})$
  - return $\mathbf{P} = \mathbf{R}^{(2)}\mathbf{P}^{(1)}$

- Small weights: cost bound

$$\mathcal{O}(m^{\omega-1}\mathrm{M}(\sigma)\log(\sigma)^2)$$

Divide-and-conquer algorithm

Algorithm (Minimal interpolation basis, $\mathbf{w} = \mathbf{0}$)

- If $\sigma \leqslant m$:
  - compute $\mathbf{P}$ by linearization
- Else:
  - $\mathbf{P}^{(1)}$ for first $\sigma/2$ points, $\mathbf{f}$, and weights $\mathbf{0}$
  - $\mathbf{P}^{(2)}$ for last $\sigma/2$ points, updated $\mathbf{f}$, and weights $\mathbf{0}$
  - $\mathbf{R}^{(2)} =$ Change weights of $\mathbf{P}^{(2)}$ to $\deg(\mathbf{P}^{(1)})$
  - return $\mathbf{P} = \mathbf{R}^{(2)}\mathbf{P}^{(1)}$

- Small weights: cost bound

$$\mathcal{O}(m^{\omega-1}\mathrm{M}(\sigma)\log(\sigma)^2)$$

- Arbitrary weights: choice between

$$\mathcal{O}(m^{\omega-1}\mathrm{M}(\sigma)\log(\sigma)^2 + m^{\omega-1}\mathrm{M}(|\mathbf{w}|)\log(|\mathbf{w}|))$$

and $\quad \mathcal{O}(m^{\omega}\mathrm{M}(\sigma)\log(\sigma)^2)$

Conclusion

This algorithm

- solves the general problem
- for small weights
- cost bound $\mathcal{O}^{\sim}(m^{\omega-1}\sigma)$
- outputs a minimal basis

Conclusion

This algorithm

- solves the general problem
- for small weights
- cost bound $\mathcal{O}^{\sim}(m^{\omega-1}\sigma)$
- outputs a minimal basis

Another algorithm (uses first algorithm as a building block)

- solves the general problem
- for arbitrary weights
- cost bound $\mathcal{O}^{\sim}(m^{\omega-1}\sigma)$
- output basis in Popov normal form

Conclusion

Algorithm for the general problem

- cost bound $\tilde{\mathcal{O}}(m^{\omega-1}\sigma)$
- outputs an interpolation basis
- supports arbitrary weights **w**
- basis in **w**-Popov form

Applications include

- simultaneous Hermite-Padé / M-Padé approximants
- list- and soft-decoding of Reed-Solomon codes: bivariate interpolation step
- list-decoding of folded R-S codes / Private Information Retrieval: multivariate interpolation step

Unbalanced Hermite-Padé ($m = 4, \sigma = 128$, $\mathbf{w} = \mathbf{0}$)

Parameters $\mathbb{K} = \mathbf{F}_{97}$, $m = 4, \sigma = 128$, $\mathbf{w} = \mathbf{0}$

Choose random polynomial R of degree < 128

$$\mathbf{f} = \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix} = \begin{bmatrix} R \\ R + XR \\ XR + X^2R \\ X^2R + X^3R \end{bmatrix}$$

Hermite-Padé

- interpolant $\mathbf{p}$ means $p_1 f_1 + \cdots + p_4 f_4 = 0 \bmod X^{128}$
- minimal basis has unbalanced row degrees $(1, 1, 1, 125)$
- will help to build an example with output size $\Theta(m^2\sigma)$

Unbalanced Hermite-Padé ($m = 4, \sigma = 128$, $\mathbf{w} = \mathbf{0}$)

Iterative algorithm:

i	1
$\mathbf{w}$	$(0, 0, 0, 0)$
f_1	R
f_2	$R + XR$
f_3	$XR + X^2R$
f_4	$X^2R + X^3R$

$\mathbf{P}$

Unbalanced Hermite-Padé ($m = 4, \sigma = 128$, $\mathbf{w} = \mathbf{0}$)

Iterative algorithm:

i	1	2
$\mathbf{w}$	$(0, 0, 0, 0)$	$(1, 0, 0, 0)$
f_1	R	XR
f_2	$R + XR$	XR
f_3	$XR + X^2R$	$XR + X^2R$
f_4	$X^2R + X^3R$	$X^2R + X^3R$
$\mathbf{P}$	$\begin{bmatrix} 1 & & & \\ 0 & 0 & & \\ & & 0 & \\ & & & 0 \end{bmatrix}$	

Unbalanced Hermite-Padé ($m = 4, \sigma = 128$, $\mathbf{w} = \mathbf{0}$)

Iterative algorithm:

i	1	2	3
$\mathbf{w}$	$(0, 0, 0, 0)$	$(1, 0, 0, 0)$	$(1, 1, 0, 0)$
f_1	R	XR	0
f_2	$R + XR$	XR	X^2R
f_3	$XR + X^2R$	$XR + X^2R$	X^2R
f_4	$X^2R + X^3R$	$X^2R + X^3R$	$X^2R + X^3R$
$\mathbf{P}$	$\begin{bmatrix} 1 & & & \\ 0 & 0 & & \\ & & 0 & \\ & & & 0 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & & \\ 1 & 1 & & \\ 0 & 0 & 0 & \\ & & & 0 \end{bmatrix}$	

Unbalanced Hermite-Padé ($m = 4, \sigma = 128$, $\mathbf{w} = \mathbf{0}$)

Iterative algorithm:

i	1	2	3	4
$\mathbf{w}$	$(0, 0, 0, 0)$	$(1, 0, 0, 0)$	$(1, 1, 0, 0)$	$(1, 1, 1, 0)$
f_1	R	XR	0	0
f_2	$R + XR$	XR	X^2R	0
f_3	$XR + X^2R$	$XR + X^2R$	X^2R	X^3R
f_4	$X^2R + X^3R$	$X^2R + X^3R$	$X^2R + X^3R$	X^3R
$\mathbf{P}$	$\begin{bmatrix} 1 & & & \\ 0 & 0 & & \\ & & 0 & \\ & & & 0 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & & \\ 1 & 1 & & \\ 0 & 0 & 0 & \\ & & & 0 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & & \\ 1 & 1 & 0 & \\ 1 & 1 & 1 & \\ 0 & 0 & 0 & 0 \end{bmatrix}$	

Unbalanced Hermite-Padé ($m = 4, \sigma = 128$, $\mathbf{w} = \mathbf{0}$)

Iterative algorithm:

i	1	2	3	4	$\cdots$
$\mathbf{w}$	$(0,0,0,0)$	$(1,0,0,0)$	$(1,1,0,0)$	$(1,1,1,0)$	$\cdots$
f_1	R	XR	0	0	0
f_2	$R + XR$	XR	X^2R	0	0
f_3	$XR + X^2R$	$XR + X^2R$	X^2R	X^3R	0
f_4	$X^2R + X^3R$	$X^2R + X^3R$	$X^2R + X^3R$	X^3R	X^4R
$\mathbf{P}$	$\begin{bmatrix} 1 & & & \\ 0 & 0 & & \\ & & 0 & \\ & & & 0 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & & \\ 1 & 1 & & \\ 0 & 0 & 0 & \\ & & & 0 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & & \\ 1 & 1 & 0 & \\ 1 & 1 & 1 & \\ 0 & 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & & \\ 1 & 1 & 0 & \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$	$\cdots$

Unbalanced Hermite-Padé ($m = 4, \sigma = 128$, $\mathbf{w} = \mathbf{0}$)

Iterative algorithm:

i	1	2	3	4	$\cdots$
$\mathbf{w}$	$(0,0,0,0)$	$(1,0,0,0)$	$(1,1,0,0)$	$(1,1,1,0)$	$\cdots$
f_1	R	XR	0	0	0
f_2	$R + XR$	XR	X^2R	0	0
f_3	$XR + X^2R$	$XR + X^2R$	X^2R	X^3R	0
f_4	$X^2R + X^3R$	$X^2R + X^3R$	$X^2R + X^3R$	X^3R	X^4R
$\mathbf{P}$	$\begin{bmatrix} 1 & & & \\ 0 & 0 & & \\ & & 0 & \\ & & & 0 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & & \\ 1 & 1 & & \\ 0 & 0 & 0 & \\ & & & 0 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & & \\ 1 & 1 & 0 & \\ 1 & 1 & 1 & \\ 0 & 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & & \\ 1 & 1 & 0 & \\ 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$	$\cdots$

Degrees and pivots in minimal basis: $\begin{bmatrix} 1 & 0 & & \\ 1 & 1 & 0 & \\ 1 & 1 & 1 & 0 \\ 125 & 125 & 125 & 125 \end{bmatrix}$

Minimal basis with size $\Theta(m^2\sigma)$

Parameters $m = 8, \sigma = 128$, $\mathbf{w} = (0, 0, 0, 0, \sigma, \sigma, \sigma, \sigma)$

Input $\mathbf{f}$: same f_1, f_2, f_3, f_4 / random f_5, f_6, f_7, f_8

$i = 4$ $\qquad\qquad$ $i = 128$

$$\begin{bmatrix}
1 & 0 & & & & & & \\
1 & 1 & 0 & & & & & \\
1 & 1 & 1 & 0 & & & & \\
1 & 1 & 1 & 1 & & & & \\
0 & 0 & 0 & 0 & 0 & & & \\
0 & 0 & 0 & 0 & & 0 & & \\
0 & 0 & 0 & 0 & & & 0 & \\
0 & 0 & 0 & 0 & & & & 0
\end{bmatrix}$$

Minimal basis with size $\Theta(m^2\sigma)$

Parameters $m = 8, \sigma = 128$, $\mathbf{w} = (0, 0, 0, 0, \sigma, \sigma, \sigma, \sigma)$

Input $\mathbf{f}$: same f_1, f_2, f_3, f_4 / random f_5, f_6, f_7, f_8

$i = 4$

$$\begin{bmatrix}
1 & 0 & & & & & & \\
1 & 1 & 0 & & & & & \\
1 & 1 & 1 & 0 & & & & \\
1 & 1 & 1 & 1 & & & & \\
0 & 0 & 0 & 0 & 0 & & & \\
0 & 0 & 0 & 0 & & 0 & & \\
0 & 0 & 0 & 0 & & & 0 & \\
0 & 0 & 0 & 0 & & & & 0
\end{bmatrix}$$

$i = 128$

$$\begin{bmatrix}
1 & 0 & & & & & & \\
1 & 1 & 0 & & & & & \\
1 & 1 & 1 & 0 & & & & \\
125 & 125 & 125 & 125 & & & & \\
124 & 124 & 124 & 124 & 0 & & & \\
124 & 124 & 124 & 124 & & 0 & & \\
124 & 124 & 124 & 124 & & & 0 & \\
124 & 124 & 124 & 124 & & & & 0
\end{bmatrix}$$

- $1/4$ of the entries have degree $\approx \sigma \rightsquigarrow$ size $\Theta(m^2\sigma)$
- cannot be computed in $\tilde{\mathcal{O}}(m^{\omega-1}\sigma)$

Popov form

Pivot of a row: rightmost entry with highest weighted degree

$$\begin{bmatrix}
1 & 0 & & & & & & \\
1 & 1 & 0 & & & & & \\
1 & 1 & 1 & 0 & & & & \\
125 & 125 & 125 & 125 & & & & \\
124 & 124 & 124 & 124 & 0 & & & \\
124 & 124 & 124 & 124 & & 0 & & \\
124 & 124 & 124 & 124 & & & 0 & \\
124 & 124 & 124 & 124 & & & & 0
\end{bmatrix}$$

Some **w**-minimal basis

Popov form

Pivot of a row: rightmost entry with highest weighted degree

$$\begin{bmatrix}
1 & 0 & & & & & & \\
1 & 1 & 0 & & & & & \\
1 & 1 & 1 & 0 & & & & \\
125 & 125 & 125 & 125 & & & & \\
124 & 124 & 124 & 124 & 0 & & & \\
124 & 124 & 124 & 124 & & 0 & & \\
124 & 124 & 124 & 124 & & & 0 & \\
124 & 124 & 124 & 124 & & & & 0
\end{bmatrix}$$

Some $\mathbf{w}$-minimal basis

$$\begin{bmatrix}
1 & 0 & & & & & & \\
0 & 1 & 0 & & & & & \\
0 & 0 & 1 & 0 & & & & \\
0 & 0 & 0 & 125 & & & & \\
0 & 0 & 0 & 124 & 0 & & & \\
0 & 0 & 0 & 124 & & 0 & & \\
0 & 0 & 0 & 124 & & & 0 & \\
0 & 0 & 0 & 124 & & & & 0
\end{bmatrix}$$

The basis in $\mathbf{w}$-Popov form

$\mathbf{w}$-Popov (normal) form

- pivot on the diagonal and highest degree in its column
- pivot degrees $\mathbf{d} = (d_1, \ldots, d_m)$
- sum of pivot degrees $\leqslant \sigma \quad \rightsquigarrow \quad$ size $\leqslant m\sigma$

Expected basis is almost Popov

Expected: pivots taken as expected from $\mathbf{w}$

$\mathbf{w} = (0, 0, 0, 0)$

$$\begin{bmatrix} 4 & 3 & 3 & 3 \\ 4 & 4 & 3 & 3 \\ 4 & 4 & 4 & 3 \\ 4 & 4 & 4 & 4 \end{bmatrix} \quad \begin{bmatrix} 7 & 6 & 1 & 6 \\ 0 & 1 & & 0 \\ & & 2 & \\ 6 & 6 & 1 & 6 \end{bmatrix}$$

$\mathbf{w} = (0, 2, 4, 6)$

$$\begin{bmatrix} 7^7 & 4^6 & 2^6 & 0^6 \\ 7^7 & 5^7 & 2^6 & 0^6 \\ 7^7 & 5^7 & 3^7 & 0^6 \\ 7^7 & 5^7 & 3^7 & 1^7 \end{bmatrix} \quad \begin{bmatrix} 8^8 & 5^7 & 1^5 & \\ 8^8 & 6^8 & 1^5 & \\ & & 2^6 & \\ 0^0 & 1^3 & & 0^6 \end{bmatrix}$$

Expected basis is almost Popov

Expected: pivots taken as expected from **w**

$\mathbf{w} = (0,0,0,0)$

$$\begin{bmatrix} 4 & 3 & 3 & 3 \\ 4 & 4 & 3 & 3 \\ 4 & 4 & 4 & 3 \\ 4 & 4 & 4 & 4 \end{bmatrix} \quad \begin{bmatrix} 7 & 6 & 1 & 6 \\ 0 & 1 & & 0 \\ & & 2 & \\ 6 & 6 & 1 & 6 \end{bmatrix}$$

$\mathbf{w} = (0,2,4,6)$

$$\begin{bmatrix} 7^7 & 4^6 & 2^6 & 0^6 \\ 7^7 & 5^7 & 2^6 & 0^6 \\ 7^7 & 5^7 & 3^7 & 0^6 \\ 7^7 & 5^7 & 3^7 & 1^7 \end{bmatrix} \quad \begin{bmatrix} 8^8 & 5^7 & 1^5 & \\ 8^8 & 6^8 & 1^5 & \\ & & 2^6 & \\ 0^0 & 1^3 & & 0^6 \end{bmatrix}$$

$\mathbf{w} = (0,0,0,0,128,128,128,128)$

$$\begin{bmatrix} 32 & 31 & 31 & 31 & & & & \\ 32 & 32 & 31 & 31 & & & & \\ 32 & 32 & 32 & 31 & & & & \\ 32 & 32 & 32 & 32 & & & & \\ 31 & 31 & 31 & 31 & 0 & & & \\ 31 & 31 & 31 & 31 & & 0 & & \\ 31 & 31 & 31 & 31 & & & 0 & \\ 31 & 31 & 31 & 31 & & & & 0 \end{bmatrix} \quad \begin{bmatrix} 1 & 0 & & & & & & \\ 1 & 1 & 0 & & & & & \\ 1 & 1 & 1 & 0 & & & & \\ 125 & 125 & 125 & 125 & & & & \\ 124 & 124 & 124 & 124 & 0 & & & \\ 124 & 124 & 124 & 124 & & 0 & & \\ 124 & 124 & 124 & 124 & & & 0 & \\ 124 & 124 & 124 & 124 & & & & 0 \end{bmatrix}$$

Follow a path with expected pivots (1/2)

Input: **f** and **w** (and the linear functionals)
and suppose we know $\mathbf{d} = (d_1, \ldots, d_m)$ the pivot degrees for **w**

Follow a path with expected pivots (1/2)

Input: **f** and **w** (and the linear functionals)
and suppose we know $\mathbf{d} = (d_1, \ldots, d_m)$ the pivot degrees for **w**

Compute minimal basis for weights with expected pivot degrees **d**

$$-\mathbf{d} = (-1, -1, -1, -125, 0, 0, 0, 0)$$

$$\begin{bmatrix}
1 & 0 & & & & & & \\
1 & 1 & 0 & & & & & \\
1 & 1 & 1 & 0 & & & & \\
1 & 1 & 1 & 125 & & & & \\
0 & 0 & 0 & 124 & 0 & & & \\
0 & 0 & 0 & 124 & & 0 & & \\
0 & 0 & 0 & 124 & & & 0 & \\
0 & 0 & 0 & 124 & & & & 0
\end{bmatrix}$$

- column degrees exactly **d**
- row degrees $(0, 0, 0, 0, 0, 0, 0, 0)$
- same pivot degrees **d** if any pivot

Follow a path with expected pivots (1/2)

Input: **f** and **w** (and the linear functionals)
and suppose we know $\mathbf{d} = (d_1, \ldots, d_m)$ the pivot degrees for **w**

Compute minimal basis for weights with expected pivot degrees **d**

$$-\mathbf{d} = (-1, -1, -1, -125, 0, 0, 0, 0)$$

$$\begin{bmatrix}
1 & 0 & & & & & & \\
1 & 1 & 0 & & & & & \\
1 & 1 & 1 & 0 & & & & \\
1 & 1 & 1 & 125 & & & & \\
0 & 0 & 0 & 124 & 0 & & & \\
0 & 0 & 0 & 124 & & 0 & & \\
0 & 0 & 0 & 124 & & & 0 & \\
0 & 0 & 0 & 124 & & & & 0
\end{bmatrix}
\qquad
\begin{bmatrix}
1 & 1 & 1 & 125 & 0 & 0 & 0 & 0 \\
1 & 1 & 1 & 125 & 0 & 0 & 0 & 0 \\
1 & 1 & 1 & 125 & 0 & 0 & 0 & 0 \\
1 & 1 & 1 & 125 & 0 & 0 & 0 & 0 \\
1 & 1 & 1 & 125 & 0 & 0 & 0 & 0 \\
1 & 1 & 1 & 125 & 0 & 0 & 0 & 0 \\
1 & 1 & 1 & 125 & 0 & 0 & 0 & 0 \\
1 & 1 & 1 & 125 & 0 & 0 & 0 & 0
\end{bmatrix}$$

- column degrees exactly **d**
- row degrees $(0, 0, 0, 0, 0, 0, 0, 0)$
- same pivot degrees **d** if any pivot

Follow a path with expected pivots (2/2)

Suppose we know $\mathbf{d} = (d_1, \ldots, d_m)$ the pivot degrees for $\mathbf{w}$

Compute $\mathbf{P}$ for weights $-\mathbf{d} = (-d_1, \ldots, -d_m)$

Correctness

- $\mathbf{P}$ $-\mathbf{d}$-minimal but likely not $\mathbf{w}$-minimal
- still, basis in $-\mathbf{d}$-Popov form $=$ basis in $\mathbf{w}$-Popov form

Follow a path with expected pivots (2/2)

Suppose we know $\mathbf{d} = (d_1, \ldots, d_m)$ the pivot degrees for $\mathbf{w}$

Compute $\mathbf{P}$ for weights $-\mathbf{d} = (-d_1, \ldots, -d_m)$

Correctness

- $\mathbf{P}$ $-\mathbf{d}$-minimal but likely not $\mathbf{w}$-minimal
- still, basis in $-\mathbf{d}$-Popov form $=$ basis in $\mathbf{w}$-Popov form

Efficiency

- $\mathbf{P}$ has $-\mathbf{d}$-row degrees $(0, \ldots, 0)$
  $\rightsquigarrow$ $-\mathbf{d}$-Popov form computed in $\mathcal{O}(m^{\omega-1}\sigma)$
- $\mathbf{P}$ has known column degrees and their sum $= \sigma$
  $\rightsquigarrow$ partial linearization [Storjohann's second transformation, 2006]
  $\rightsquigarrow$ reduces to small weights case, cost $\mathcal{O}^{\sim}(m^{\omega-1}\sigma)$

Find the pivot degrees

How to know the pivot degrees $\mathbf{d} = (d_1, \ldots, d_m)$?

Divide-and-conquer

- $\mathbf{P}^{(1)}$ in $\mathbf{w}$-Popov form with pivot degrees $\mathbf{d}^{(1)}$
- $\mathbf{P}^{(2)}$ in $\mathbf{w}'$-Popov form with pivot degrees $\mathbf{d}^{(2)}$

Product $\mathbf{P}^{(2)}\mathbf{P}^{(1)}$

- $\mathbf{w}$-minimal but usually not in $\mathbf{w}$-Popov form
- can have many large degree entries
  $\rightsquigarrow$ we cannot afford computing it

Find the pivot degrees

How to know the pivot degrees $\mathbf{d} = (d_1, \ldots, d_m)$?

Divide-and-conquer

- $\mathbf{P}^{(1)}$ in $\mathbf{w}$-Popov form with pivot degrees $\mathbf{d}^{(1)}$
- $\mathbf{P}^{(2)}$ in $\mathbf{w}'$-Popov form with pivot degrees $\mathbf{d}^{(2)}$

Product $\mathbf{P}^{(2)}\mathbf{P}^{(1)}$

- $\mathbf{w}$-minimal but usually not in $\mathbf{w}$-Popov form
- can have many large degree entries
  $\rightsquigarrow$ we cannot afford computing it
- pivots on the diagonal
- pivot degrees $\mathbf{d} = \mathbf{d}^{(1)} + \mathbf{d}^{(2)}$

Algorithm

Algorithm (Minimal interpolation basis, arbitrary $\mathbf{w}$)

- If $\sigma \leqslant m$:
  - compute $\mathbf{P}$ by linearization ($\rightsquigarrow$ returns Popov)
- Else:
  - $\mathbf{P}^{(1)}$ for first $\sigma/2$, $\mathbf{f}$, and weights $\mathbf{w}$
  - $\mathbf{P}^{(2)}$ for last $\sigma/2$, updated $\mathbf{f}$, and updated weights $\mathbf{w}'$
  - Deduce the pivot degrees $\mathbf{d} = (d_1, \ldots, d_m)$ of $\mathbf{P}^{(2)}\mathbf{P}^{(1)}$
  - $\mathbf{P}$ for $\sigma, \mathbf{f}$, and $-\mathbf{d}$ using partial linearization + small weights case
  - Return $\mathrm{lm}_{-\mathbf{d}}(\mathbf{P})^{-1}\mathbf{P}$

Arbitrary weights: cost bound

$$\mathcal{O}(m^{\omega-1}\mathrm{M}(\sigma)\log(\sigma)^3)$$

Conclusion

Algorithm for the general problem

- cost bound $\tilde{\mathcal{O}}(m^{\omega-1}\sigma)$
- outputs an interpolation basis
- supports arbitrary weights **w**
- basis in **w**-Popov form

Applications include

- simultaneous Hermite-Padé / M-Padé approximants
- list- and soft-decoding of Reed-Solomon codes: bivariate interpolation step
- list-decoding of folded R-S codes / Private Information Retrieval: multivariate interpolation step

Ingredient 2: maintaining an evaluation matrix

Divide-and-conquer algorithm:

- $\mathbf{P}^{(1)}$ for first $\sigma/2$ points and $\mathbf{w}$
- $\mathbf{P}^{(2)}$ for last $\sigma/2$ points and $\mathbf{w}'$, with $\mathbf{P}^{(1)}$-dependent evaluation
- return $\mathbf{P} = \mathbf{P}^{(2)}\mathbf{P}^{(1)}$

Input: evaluation matrix $\mathbf{E} = [E_1|\cdots|E_\sigma] \in \mathbb{K}^{m\times\sigma}$

Residual: $\mathbf{P}\cdot\mathbf{E} = [\mathbf{P}(x_1)E_1 \mid \cdots \mid \mathbf{P}(x_\sigma)E_\sigma]$
Assumption $\star$: $\mathbf{P}\cdot\mathbf{E}$ computed in $\mathcal{O}^{\sim}(m^{\omega-1}\sigma)$

Compromise between the number of distinct x_i and the degrees involved

- if few distinct x_i:
  - evaluate $\mathbf{P}$ at those points,
  - perform scalar matrices multiplications (e.g. $\mathbf{P}(x_1)E$)
- if many distinct x_i:
  - interpolate with some $\mathbf{F}(X)$ the values in $\mathbf{E}$ at distinct x_i
  - perform a polynomial matrix multiplication of the form $\mathbf{P}(X)\mathbf{F}(X)$
  - evaluate the result at distinct x_i

Ingredient 3: linearizing at small orders (1/3)

Base case $\sigma = m$, goal $\tilde{\mathcal{O}}(m^\omega)$

Input: points $x_1, \ldots, x_m$, evaluation matrix $\mathbf{E} \in \mathbb{K}^{m \times m}$

Output: $\mathbf{0}$-minimal interpolation basis $\mathbf{P}$

sum of row degrees $\leqslant m \Rightarrow$ average degree in the matrix $\leqslant 1$

Complete linearization over $\mathbb{K}$

$$\mathcal{K} = \begin{bmatrix} \mathbf{E} \\ \mathbf{ED} \\ \mathbf{ED^2} \\ \vdots \\ \mathbf{ED^M} \end{bmatrix} \qquad \text{where} \quad \mathbf{D} = \mathrm{Diag}(x_1, \ldots, x_m)$$

minimal interpolation basis $\longleftrightarrow$ minimal linear relations between rows of $\mathcal{K}$

Fast computation of those relations, in $\mathcal{O}(m^\omega \log m)$

Ingredient 3: linearizing at small orders (2/3)

$$\mathcal{K} = \left[\begin{array}{cccc}
E_{11} & E_{12} & E_{13} & E_{14} \\
E_{21} & E_{22} & E_{23} & E_{24} \\
E_{31} & E_{32} & E_{33} & E_{34} \\
E_{41} & E_{42} & E_{43} & E_{44} \\
\hline
x_1 E_{11} & x_2 E_{12} & x_3 E_{13} & x_4 E_{14} \\
x_1 E_{21} & x_2 E_{22} & x_3 E_{23} & x_4 E_{24} \\
x_1 E_{31} & x_2 E_{32} & x_3 E_{33} & x_4 E_{34} \\
x_1 E_{41} & x_2 E_{42} & x_3 E_{43} & x_4 E_{44} \\
\hline
x_1^2 E_{11} & x_2^2 E_{12} & x_3^2 E_{13} & x_4^2 E_{14} \\
x_1^2 E_{21} & x_2^2 E_{22} & x_3^2 E_{23} & x_4^2 E_{24} \\
x_1^2 E_{31} & x_2^2 E_{32} & x_3^2 E_{33} & x_4^2 E_{34} \\
x_1^2 E_{41} & x_2^2 E_{42} & x_3^2 E_{43} & x_4^2 E_{44} \\
\hline
x_1^3 E_{11} & x_2^3 E_{12} & x_3^3 E_{13} & x_4^3 E_{14} \\
x_1^3 E_{21} & x_2^3 E_{22} & x_3^3 E_{23} & x_4^3 E_{24} \\
x_1^3 E_{31} & x_2^3 E_{32} & x_3^3 E_{33} & x_4^3 E_{34} \\
x_1^3 E_{41} & x_2^3 E_{42} & x_3^3 E_{43} & x_4^3 E_{44} \\
\hline
x_1^4 E_{11} & x_2^4 E_{12} & x_3^4 E_{13} & x_4^4 E_{14} \\
x_1^4 E_{21} & x_2^4 E_{22} & x_3^4 E_{23} & x_4^4 E_{24} \\
x_1^4 E_{31} & x_2^4 E_{32} & x_3^4 E_{33} & x_4^4 E_{34} \\
x_1^4 E_{41} & x_2^4 E_{42} & x_3^4 E_{43} & x_4^4 E_{44}
\end{array}\right]$$

Linearize in degree $\leqslant m$

left nullspace $\longleftrightarrow$ interpolants

- Find the first $\operatorname{rank}(\mathcal{K})$ linearly independent rows
- Compute well-chosen linear relations between rows

$\rightsquigarrow$ minimal interpolation basis

Ingredient 3: linearizing at small orders (3/3)

Field $\mathbb{F}_{97}$, $m = \sigma = 4$, points $71, 66, 60, 72$

Indices

$$\left[\begin{array}{c} \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \end{array}\right]$$

$$\mathcal{K} = \left[\begin{array}{cccc} 91 & 32 & 77 & 32 \\ 7 & 0 & 25 & 35 \\ 1 & 32 & 5 & 67 \\ 25 & 18 & 81 & 59 \\ \hline 59 & 75 & 61 & 73 \\ 12 & 0 & 45 & 95 \\ 71 & 75 & 9 & 71 \\ 29 & 24 & 10 & 77 \\ \hline 18 & 3 & 71 & 18 \\ 76 & 0 & 81 & 50 \\ 94 & 3 & 55 & 68 \\ 22 & 32 & 18 & 15 \\ \hline 17 & 4 & 89 & 35 \\ 61 & 0 & 10 & 11 \\ 78 & 4 & 2 & 46 \\ 10 & 75 & 13 & 13 \end{array}\right]$$

Ingredient 3: linearizing at small orders (3/3)

Field $\mathbb{F}_{97}$, $m = \sigma = 4$, points $71, 66, 60, 72$

Indices

$$\left[\begin{array}{c} \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \end{array}\right]$$

$$\mathcal{K} = \left[\begin{array}{cccc} 91 & 32 & 77 & 32 \\ 7 & 0 & 25 & 35 \\ 1 & 32 & 5 & 67 \\ 25 & 18 & 81 & 59 \\ \hline 59 & 75 & 61 & 73 \\ 12 & 0 & 45 & 95 \\ 71 & 75 & 9 & 71 \\ 29 & 24 & 10 & 77 \\ \hline 18 & 3 & 71 & 18 \\ 76 & 0 & 81 & 50 \\ 94 & 3 & 55 & 68 \\ 22 & 32 & 18 & 15 \\ \hline 17 & 4 & 89 & 35 \\ 61 & 0 & 10 & 11 \\ 78 & 4 & 2 & 46 \\ 10 & 75 & 13 & 13 \end{array}\right]$$

Ingredient 3: linearizing at small orders (3/3)

Field $\mathbb{F}_{97}$, $m = \sigma = 4$, points $71, 66, 60, 72$

Indices

$$\begin{bmatrix} & & & \\ & & & \\ & & & \\ & & & \end{bmatrix}$$

$$\mathcal{K} = \left[\begin{array}{cccc} 91 & 32 & 77 & 32 \\ 0 & 5 & 34 & 40 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 55 & 81 \\ \hline 59 & 75 & 61 & 73 \\ 12 & 0 & 45 & 95 \\ 71 & 75 & 9 & 71 \\ 29 & 24 & 10 & 77 \\ \hline 18 & 3 & 71 & 18 \\ 76 & 0 & 81 & 50 \\ 94 & 3 & 55 & 68 \\ 22 & 32 & 18 & 15 \\ \hline 17 & 4 & 89 & 35 \\ 61 & 0 & 10 & 11 \\ 78 & 4 & 2 & 46 \\ 10 & 75 & 13 & 13 \end{array}\right]$$

Ingredient 3: linearizing at small orders (3/3)

Field $\mathbb{F}_{97}$, $m = \sigma = 4$, points $71, 66, 60, 72$

Indices 00 01 02

$$\begin{bmatrix} 96 & 96 & 1 \\ & & \\ & & \\ & & \end{bmatrix}$$

$$\mathcal{K} = \left[\begin{array}{cccc} 91 & 32 & 77 & 32 \\ 0 & 5 & 34 & 40 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 55 & 81 \\ \hline 59 & 75 & 61 & 73 \\ 12 & 0 & 45 & 95 \\ 71 & 75 & 9 & 71 \\ 29 & 24 & 10 & 77 \\ \hline 18 & 3 & 71 & 18 \\ 76 & 0 & 81 & 50 \\ 94 & 3 & 55 & 68 \\ 22 & 32 & 18 & 15 \\ \hline 17 & 4 & 89 & 35 \\ 61 & 0 & 10 & 11 \\ 78 & 4 & 2 & 46 \\ 10 & 75 & 13 & 13 \end{array}\right]$$

Ingredient 3: linearizing at small orders (3/3)

Field $\mathbb{F}_{97}$, $m = \sigma = 4$, points $71, 66, 60, 72$

Indices 00 01 02

$$\begin{bmatrix} 96 & 96 & 1 \\ & & \\ & & \\ & & \end{bmatrix}$$

$$\mathcal{K} = \left[\begin{array}{cccc} 91 & 32 & 77 & 32 \\ 0 & 5 & 34 & 40 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 55 & 81 \\ \hline 59 & 75 & 61 & 73 \\ 12 & 0 & 45 & 95 \\ 0 & 0 & 0 & 0 \\ 29 & 24 & 10 & 77 \\ \hline 18 & 3 & 71 & 18 \\ 76 & 0 & 81 & 50 \\ 0 & 0 & 0 & 0 \\ 22 & 32 & 18 & 15 \\ \hline 17 & 4 & 89 & 35 \\ 61 & 0 & 10 & 11 \\ 0 & 0 & 0 & 0 \\ 10 & 75 & 13 & 13 \end{array}\right]$$

Ingredient 3: linearizing at small orders (3/3)

Field $\mathbb{F}_{97}$, $m = \sigma = 4$, points $71, 66, 60, 72$

Indices 00 01 02

$$\begin{bmatrix} 96 & 96 & 1 \\ & & \\ & & \end{bmatrix}$$

$$\mathcal{K} = \left[\begin{array}{cccc} 91 & 32 & 77 & 32 \\ 0 & 5 & 34 & 40 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 55 & 81 \\ \hline 59 & 75 & 61 & 73 \\ 12 & 0 & 45 & 95 \\ 0 & 0 & 0 & 0 \\ 29 & 24 & 10 & 77 \\ \hline 18 & 3 & 71 & 18 \\ 76 & 0 & 81 & 50 \\ 0 & 0 & 0 & 0 \\ 22 & 32 & 18 & 15 \\ \hline 17 & 4 & 89 & 35 \\ 61 & 0 & 10 & 11 \\ 0 & 0 & 0 & 0 \\ 10 & 75 & 13 & 13 \end{array}\right]$$

Ingredient 3: linearizing at small orders (3/3)

Field $\mathbb{F}_{97}$, $m = \sigma = 4$, points $71, 66, 60, 72$

Indices 00 01 02

$$\begin{bmatrix} 96 & 96 & 1 \\ & & \\ & & \\ & & \end{bmatrix}$$

$$\mathcal{K} = \left[\begin{array}{cccc} 91 & 32 & 77 & 32 \\ 0 & 5 & 34 & 40 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 55 & 81 \\ \hline 0 & 0 & 0 & 10 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \hline 18 & 3 & 71 & 18 \\ 76 & 0 & 81 & 50 \\ 0 & 0 & 0 & 0 \\ 22 & 32 & 18 & 15 \\ \hline 17 & 4 & 89 & 35 \\ 61 & 0 & 10 & 11 \\ 0 & 0 & 0 & 0 \\ 10 & 75 & 13 & 13 \end{array}\right]$$

Ingredient 3: linearizing at small orders (3/3)

Field $\mathbb{F}_{97}$, $m = \sigma = 4$, points $71, 66, 60, 72$

Indices 00 01 02 03 10 11 13

$$\begin{bmatrix} 96 & 96 & 1 & & & & \\ 44 & 33 & 0 & 62 & 53 & 1 & \\ 31 & 35 & 0 & 68 & 93 & 0 & 1 \\ & & & & & & \end{bmatrix}$$

$$\mathcal{K} = \left[\begin{array}{cccc} 91 & 32 & 77 & 32 \\ 0 & 5 & 34 & 40 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 55 & 81 \\ \hline 0 & 0 & 0 & 10 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \hline 18 & 3 & 71 & 18 \\ 76 & 0 & 81 & 50 \\ 0 & 0 & 0 & 0 \\ 22 & 32 & 18 & 15 \\ \hline 17 & 4 & 89 & 35 \\ 61 & 0 & 10 & 11 \\ 0 & 0 & 0 & 0 \\ 10 & 75 & 13 & 13 \end{array}\right]$$

Ingredient 3: linearizing at small orders (3/3)

Field $\mathbb{F}_{97}$, $m = \sigma = 4$, points $71, 66, 60, 72$

Indices 00 01 02 03 10 11 13

$$\begin{bmatrix} 96 & 96 & 1 & & & & \\ 44 & 33 & 0 & 62 & 53 & 1 & \\ 31 & 35 & 0 & 68 & 93 & 0 & 1 \\ & & & & & & \end{bmatrix}$$

$$\mathcal{K} = \left[\begin{array}{cccc} 91 & 32 & 77 & 32 \\ 0 & 5 & 34 & 40 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 55 & 81 \\ \hline 0 & 0 & 0 & 10 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \hline 18 & 3 & 71 & 18 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \hline 17 & 4 & 89 & 35 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array}\right]$$

Ingredient 3: linearizing at small orders (3/3)

Field $\mathbb{F}_{97}$, $m = \sigma = 4$, points $71, 66, 60, 72$

Indices 00 01 02 03 10 11 13

$$\begin{bmatrix} 96 & 96 & 1 & & & & \\ 44 & 33 & 0 & 62 & 53 & 1 & \\ 31 & 35 & 0 & 68 & 93 & 0 & 1 \\ & & & & & & \end{bmatrix}$$

$$\mathcal{K} = \left[\begin{array}{cccc} 91 & 32 & 77 & 32 \\ 0 & 5 & 34 & 40 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 55 & 81 \\ \hline 0 & 0 & 0 & 10 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \hline 18 & 3 & 71 & 18 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \hline 17 & 4 & 89 & 35 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array}\right]$$

Ingredient 3: linearizing at small orders (3/3)

Field $\mathbb{F}_{97}$, $m = \sigma = 4$, points $71, 66, 60, 72$

Indices 00 01 02 03 10 11 13

$$\begin{bmatrix} 96 & 96 & 1 & & & & \\ 44 & 33 & 0 & 62 & 53 & 1 & \\ 31 & 35 & 0 & 68 & 93 & 0 & 1 \end{bmatrix}$$

$$\mathcal{K} = \left[\begin{array}{cccc} 91 & 32 & 77 & 32 \\ 0 & 5 & 34 & 40 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 55 & 81 \\ \hline 0 & 0 & 0 & 10 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array}\right]$$

Ingredient 3: linearizing at small orders (3/3)

Field $\mathbb{F}_{97}$, $m = \sigma = 4$, points $71, 66, 60, 72$

Indices 00 01 02 03 10 11 13 20

$$\begin{bmatrix} 96 & 96 & 1 & & & & & \\ 44 & 33 & 0 & 62 & 53 & 1 & & \\ 31 & 35 & 0 & 68 & 93 & 0 & 1 & \\ 19 & 44 & 0 & 15 & 18 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathcal{K} = \left[\begin{array}{cccc} 91 & 32 & 77 & 32 \\ 0 & 5 & 34 & 40 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 55 & 81 \\ \hline 0 & 0 & 0 & 10 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array}\right]$$

Ingredient 3: linearizing at small orders (3/3)

Field $\mathbb{F}_{97}$, $m = \sigma = 4$, points $71, 66, 60, 72$

Indices 00 01 02 03 10 11 13 20

$$\begin{bmatrix} 96 & 96 & 1 & & & & & \\ 44 & 33 & 0 & 62 & 53 & 1 & & \\ 31 & 35 & 0 & 68 & 93 & 0 & 1 & \\ 19 & 44 & 0 & 15 & 18 & 0 & 0 & 1 \end{bmatrix}$$

$$\mathcal{K} = \left[\begin{array}{cccc} 91 & 32 & 77 & 32 \\ 0 & 5 & 34 & 40 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 55 & 81 \\ \hline 0 & 0 & 0 & 10 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array}\right]$$

Ingredient 3: linearizing at small orders (3/3)

Field $\mathbb{F}_{97}$, $m = \sigma = 4$, points $71, 66, 60, 72$

Indices 00 01 02 03 10 11 13 20

$$\begin{bmatrix} 96 & 96 & 1 & & & & & \\ 44 & 33 & 0 & 62 & 53 & 1 & & \\ 31 & 35 & 0 & 68 & 93 & 0 & 1 & \\ 19 & 44 & 0 & 15 & 18 & 0 & 0 & 1 \end{bmatrix}$$

$$\downarrow$$

$$\begin{bmatrix} 96 & 96 & 1 & 0 \\ 53X + 44 & X + 33 & 0 & 62 \\ 93X + 31 & 35 & 0 & X + 68 \\ X^2 + 18X + 19 & 44 & 0 & 15 \end{bmatrix}$$

$$\mathcal{K} = \left[\begin{array}{cccc} 91 & 32 & 77 & 32 \\ 0 & 5 & 34 & 40 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 55 & 81 \\ \hline 0 & 0 & 0 & 10 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array}\right]$$

Via lattice reduction

```
sage: bF = GF(997) ; pR.<X> = bF[]
sage: m=4 ; sig=20 ; w=2
sage: points = [ ( bF.random_element() , bF.random_element() ) for i in range(sig) ]
sage:
sage: L = pR.lagrange_polynomial( points )
sage: M = prod( [ X-points[i][0] for i in range(sig) ] )
sage: P = Matrix( pR, 4, 4, [ [M,0,0,0], [ -L,1,0,0], [-L^2,0,1,0], [-L^3,0,0,1] ] )
sage:
```

Via lattice reduction

```
sage: bF = GF(997) ; pR.<X> = bF[]
sage: m=4 ; sig=20 ; w=2
sage: points = [ ( bF.random_element() , bF.random_element() ) for i in range(sig) ]
sage:
sage: L = pR.lagrange_polynomial( points )
sage: M = prod( [ X-points[i][0] for i in range(sig) ] )
sage: P = Matrix( pR, 4, 4, [ [M,0,0,0], [ -L,1,0,0], [-L^2,0,1,0], [-L^3,0,0,1] ] )
sage:
sage: Pmin = weak_popov_form( P, [0,w,2*w,3*w] )
sage: print is_minimal( P, [0,w,2*w,3*w] ); print degree_matrix( P )
False
[20 -1 -1 -1]
[19  0 -1 -1]
[38 -1  0 -1]
[57 -1 -1  0]
sage: print is_minimal( Pmin, [0,w,2*w,3*w] ); print degree_matrix( Pmin )
True
[8 5 3 1]
[8 6 3 1]
[8 6 4 2]
[8 6 4 0]
```

Via lattice reduction

```
sage: bF = GF(997) ; pR.<X> = bF[]
sage: m=4 ; sig=20 ; w=2
sage: points = [ ( bF.random_element() , bF.random_element() ) for i in range(sig) ]
sage:
sage: L = pR.lagrange_polynomial( points )
sage: M = prod( [ X-points[i][0] for i in range(sig) ] )
sage: P = Matrix( pR, 4, 4, [ [M,0,0,0], [ -L,1,0,0], [-L^2,0,1,0], [-L^3,0,0,1] ] )
sage:
sage: Pmin = weak_popov_form( P, [0,w,2*w,3*w] )
sage: print is_minimal( P, [0,w,2*w,3*w] ); print degree_matrix( P )
False
[20 -1 -1 -1]
[19  0 -1 -1]
[38 -1  0 -1]
[57 -1 -1  0]
sage: print is_minimal( Pmin, [0,w,2*w,3*w] ); print degree_matrix( Pmin )
True
[8 5 3 1]
[8 6 3 1]
[8 6 4 2]
[8 6 4 0]
```

- equations $\rightsquigarrow$ interpolation basis — not always "efficient"
- lattice reduction $\rightsquigarrow$ minimal interpolation basis

Cost bound $\tilde{\mathcal{O}}(m^\omega \sigma)$ — . . . or worse