Prony's problem and superresolution in several variables: structure and algorithms

Tomas Sauer

Lehrstuhl für Mathematik mit Schwerpunkt Digitale Bildverarbeitung
Fakultät für Informatik und Mathematik

FORWISS

Universität Passau

Multivariate Interpolation and Approximation with Applications, Luminy, September 19–23

The Problem

Exponentials

For an *exponential polynomial*

$$f(x) = \sum_{\omega\in\Omega} f_\omega \, e^{\omega^T x}, \qquad \Omega \subset (\mathbb{R} + i\mathbb{T})^s, \qquad f_\omega \in \mathbb{C},$$

"learn" Ω and f_ω from regular samples of f: $f(\Lambda)$, $\Lambda \subset \mathbb{Z}^s$.

Connections

1. Sparse polynomials: $f(x) = \sum_{\alpha\in A} f_\alpha \, x^\alpha$.
2. Multisource radar: MUSIC & ESPRIT (1D).
3. "Superresolution".
4. $\Re\omega < 0$: damped partials ...

The Problem

UNIVERSITÄT PASSAU

Exponentials

For a sparse *exponential polynomial*

$$f(x) = \sum_{\omega \in \Omega} f_\omega \, e^{\omega^T x}, \qquad \Omega \subset (\mathbb{R} + i\mathbb{T})^s, \quad 0 \neq f_\omega \in \mathbb{C},$$

"learn" Ω and f_ω from regular samples of f: $f(\Lambda)$, $\Lambda \subset \mathbb{Z}^s$.

Connections

1. Sparse polynomials: $f(x) = \sum_{\alpha \in A} f_\alpha \, x^\alpha$.
2. Multisource radar: MUSIC & ESPRIT (1D).
3. "Superresolution".
4. $\Re\omega < 0$: damped partials ...

The Problem

Exponentials

For a sparse *exponential polynomial*

$$f(x) = \sum_{\omega\in\Omega} f_\omega \, e^{\omega^T x}, \qquad \Omega \subset (\mathbb{R} + i\mathbb{T})^s, \quad 0 \neq f_\omega \in \mathbb{C},$$

"learn" Ω and f_ω from regular samples of f: $f(\Lambda)$, $\Lambda \subset \mathbb{Z}^s$.

Connections

1. Sparse polynomials: $f(x) = \sum_{\alpha\in A} f_\alpha \, x^\alpha$.
2. Multisource radar: MUSIC & ESPRIT (1D).
3. "Superresolution".
4. $\Re\omega < 0$: damped partials …

The Problem

UNIVERSITÄT PASSAU

Exponentials

For a sparse *exponential polynomial*

$$f(x) = \sum_{\omega\in\Omega} f_\omega \, e^{\omega^T x}, \qquad \Omega \subset (\mathbb{R} + i\mathbb{T})^s, \quad 0 \neq f_\omega \in \mathbb{C},$$

"learn" Ω and f_ω from regular samples of f: $f(\Lambda)$, $\Lambda \subset \mathbb{Z}^s$.

Connections

1. Sparse polynomials: $f(x) = \sum_{\alpha\in A} f_\alpha \, x^\alpha$.
2. Multisource radar: MUSIC & ESPRIT (1D).
3. "Superresolution".
4. $\Re\omega < 0$: damped partials …

The Problem

Exponentials

For a sparse *exponential polynomial*

$$f(x) = \sum_{\omega\in\Omega} f_\omega\, e^{\omega^T x}, \qquad \Omega \subset (\mathbb{R} + i\mathbb{T})^s, \quad 0 \neq f_\omega \in \mathbb{C},$$

"learn" Ω and f_ω from regular samples of f: $f(\Lambda)$, $\Lambda \subset \mathbb{Z}^s$.

Connections

1. Sparse polynomials: $f(x) = \sum_{\alpha\in A} f_\alpha\, x^\alpha$.
2. Multisource radar: MUSIC & ESPRIT (1D).
3. "Superresolution".
4. $\Re\omega < 0$: damped partials …

The Problem

Exponentials

For a sparse *exponential polynomial*

$$f(x) = \sum_{\omega\in\Omega} f_\omega\, e^{\omega^T x}, \qquad \Omega \subset (\mathbb{R} + i\mathbb{T})^s, \quad 0 \neq f_\omega \in \mathbb{C},$$

"learn" Ω and f_ω from regular samples of f: $f(\Lambda)$, $\Lambda \subset \mathbb{Z}^s$.

Connections

1. Sparse polynomials: $f(x) = \sum_{\alpha\in A} f_\alpha\, x^\alpha$.
2. Multisource radar: MUSIC & ESPRIT (1D).
3. "Superresolution".
4. $\Re\omega < 0$: damped partials …

The Problem

Exponentials

For a sparse *exponential polynomial*

$$f(x) = \sum_{\omega \in \Omega} f_\omega \, e^{\omega^T x}, \qquad \Omega \subset (\mathbb{R} + i\mathbb{T})^s, \quad 0 \neq f_\omega \in \mathbb{C},$$

"learn" Ω and f_ω from regular samples of f: $f(\Lambda)$, $\Lambda \subset \mathbb{Z}^s$.

Connections

1. Sparse polynomials: $f(x) = \sum_{\alpha \in A} f_\alpha \, x^\alpha$.
2. Multisource radar: MUSIC & ESPRIT (1D).
3. "Superresolution".
4. $\Re\omega < 0$: damped partials …

The Problem

Exponentials

For a sparse *exponential polynomial*

$$f(x) = \sum_{\omega \in \Omega} f_\omega \, e^{\omega^T x}, \qquad \Omega \subset (\mathbb{R} + i\mathbb{T})^s, \quad 0 \neq f_\omega \in \mathbb{C},$$

"learn" Ω and f_ω from regular samples of f: $f(\Lambda)$, $\Lambda \subset \mathbb{Z}^s$.

Connections

1. Sparse polynomials: $f(x) = \sum_{\alpha \in A} f_\alpha \, x^\alpha$.
2. Multisource radar: MUSIC & ESPRIT (1D).
3. "Superresolution".
4. $\Re\omega < 0$: damped partials, "piano learning" ...

Superresolution

The imaging model [Candes & Fernandez–Granda]

1. Few localized sources.
2. Image acquisition: *low pass*.
3. Point spread function.
4. Localization deteriorates.
5. Remedy: *deconvolution*.
6. For example by minimization.

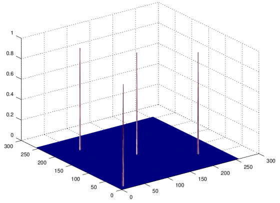
Mathematical model

$$x = \sum_{\omega \in \Omega} f_\omega \, \delta_\omega$$

Superresolution

The imaging model [Candes & Fernandez–Granda]

1. Few localized sources.
2. Image acquisition: *low pass*.
3. Point spread function.
4. Localization deteriorates.
5. Remedy: *deconvolution*.
6. For example by minimization.



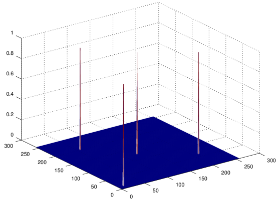
Mathematical model

$$x = \sum_{\omega \in \Omega} f_\omega \, \delta_\omega$$

Superresolution

The imaging model [Candes & Fernandez–Granda]

1. Few localized sources.
2. Image acquisition: *low pass*.
3. Point spread function.
4. Localization deteriorates.
5. Remedy: *deconvolution*.
6. For example by minimization.



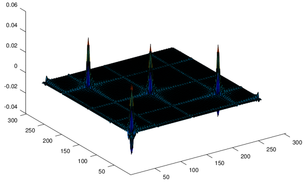
Mathematical model

$$x = \sum_{\omega \in \Omega} f_\omega \, \delta_\omega$$

Superresolution

The imaging model [Candes & Fernandez–Granda]

1. Few localized sources.
2. Image acquisition: *low pass*.
3. Point spread function.
4. Localization deteriorates.
5. Remedy: *deconvolution*.
6. For example by minimization.



Mathematical model

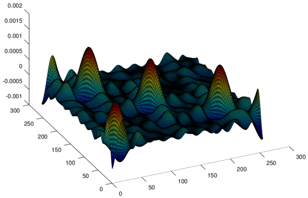
$$x = \sum_{\omega \in \Omega} f_\omega \, \delta_\omega$$

Superresolution

UNIVERSITÄT PASSAU

The imaging model [Candes & Fernandez–Granda]

1. Few localized sources.
2. Image acquisition: *low pass*.
3. Point spread function.
4. Localization deteriorates.
5. Remedy: *deconvolution*.
6. For example by minimization.



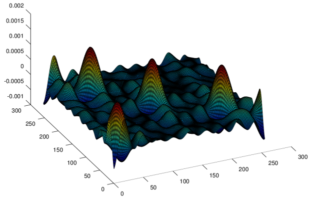
Mathematical model

$$x = \sum_{\omega \in \Omega} f_\omega \, \delta_\omega$$

Superresolution

The imaging model [Candes & Fernandez–Granda]

1. Few localized sources.
2. Image acquisition: *low pass*.
3. Point spread function.
4. Localization deteriorates.
5. Remedy: *deconvolution*.
6. For example by minimization.



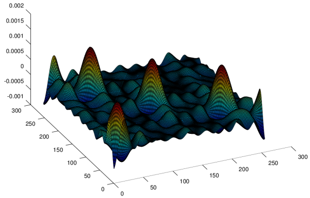
Mathematical model

$$x = \sum_{\omega \in \Omega} f_\omega \, \delta_\omega$$

Superresolution

The imaging model [Candes & Fernandez–Granda]

1. Few localized sources.
2. Image acquisition: *low pass*.
3. Point spread function.
4. Localization deteriorates.
5. Remedy: *deconvolution*.
6. For example by minimization.



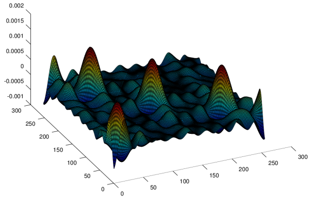
Mathematical model

$$x = \sum_{\omega \in \Omega} f_\omega \, \delta_\omega$$

Superresolution

The imaging model [Candes & Fernandez–Granda]

1. Few localized sources.
2. Image acquisition: *low pass*.
3. Point spread function.
4. Localization deteriorates.
5. Remedy: *deconvolution*.
6. For example by minimization.



Mathematical model

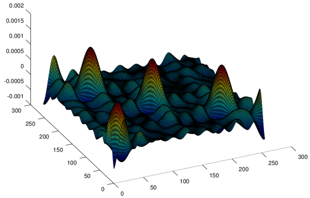
$$x = \sum_{\omega \in \Omega} f_\omega \, \delta_\omega$$

Superresolution

UNIVERSITÄT PASSAU

The imaging model [Candes & Fernandez–Granda]

1. Few localized sources.
2. Image acquisition: *low pass*.
3. Point spread function.
4. Localization deteriorates.
5. Remedy: *deconvolution*.
6. For example by minimization.



Mathematical model

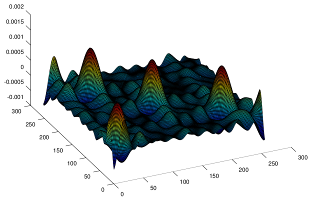
$$x = \sum_{\omega \in \Omega} f_\omega \, \delta_\omega \qquad \mapsto \qquad \hat{x}(\alpha) =$$

Superresolution

UNIVERSITÄT PASSAU

The imaging model [Candes & Fernandez–Granda]

1. Few localized sources.
2. Image acquisition: *low pass*.
3. Point spread function.
4. Localization deteriorates.
5. Remedy: *deconvolution*.
6. For example by minimization.



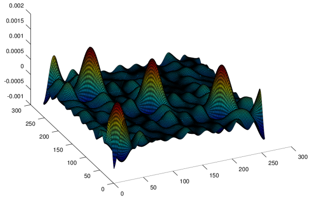
Mathematical model

$$x = \sum_{\omega \in \Omega} f_\omega \, \delta_\omega \qquad \mapsto \qquad \hat{x}(\alpha) = (2\pi)^{-s} \int_{\mathbb{T}^s} x(t) \, e^{-i\alpha^T t} \, dt$$

Superresolution

The imaging model [Candes & Fernandez–Granda]

1. Few localized sources.
2. Image acquisition: *low pass*.
3. Point spread function.
4. Localization deteriorates.
5. Remedy: *deconvolution*.
6. For example by minimization.



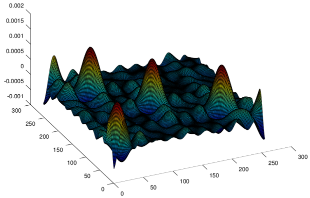
Mathematical model

$$x = \sum_{\omega\in\Omega} f_\omega \, \delta_\omega \qquad \mapsto \qquad \hat{x}(\alpha) = \sum_{\omega\in\Omega} f_\omega \, e^{-i\alpha^T\omega}$$

Superresolution

The imaging model [Candes & Fernandez–Granda]

1. Few localized sources.
2. Image acquisition: *low pass*.
3. Point spread function.
4. Localization deteriorates.
5. Remedy: *deconvolution*.
6. For example by minimization.



Mathematical model

$$x = \sum_{\omega \in \Omega} f_\omega \, \delta_\omega \qquad \mapsto \qquad \hat{x}(\alpha) = \sum_{\omega \in \Omega} f_\omega \, e^{-i\alpha^T \omega}, \qquad \alpha \in \Lambda.$$

Moment Problems

Points & moments

1. Point set $X_\Omega = e^\Omega = \{x_\omega := e^\omega : \omega \in \Omega\}$.
2. Discrete measure $\mu = \sum_\omega f_\omega \, \delta_{x_\omega}$.
3. Moments
$$\mu(\alpha) = \int x^\alpha \, d\mu(x)$$

Prony as moment problem

1. Square positive functionals.
2. Flat extensions $\to$ B. Mourrain.

Moment Problems

Points & moments

1. Point set $X_\Omega = e^\Omega = \{x_\omega := e^\omega : \omega \in \Omega\}$.
2. Discrete measure $\mu = \sum_\omega f_\omega \, \delta_{x_\omega}$.
3. Moments

$$\mu(\alpha) = \int x^\alpha \, d\mu(x)$$

Prony as moment problem

4. Square positive functionals.
5. Flat extensions $\rightarrow$ B. Mourrain.

Moment Problems

Points & moments

1. Point set $X_\Omega = e^\Omega = \{x_\omega := e^\omega : \omega \in \Omega\}$. $\qquad e^\omega = e^{\omega_1} \cdots e^{\omega_s}$.
2. Discrete measure $\mu = \sum_\omega f_\omega \, \delta_{x_\omega}$.
3. Moments
$$\mu(\alpha) = \int x^\alpha \, d\mu(x)$$

Prony as moment problem

4. Square positive functionals.
5. Flat extensions $\rightarrow$ B. Mourrain.

Moment Problems

UNIVERSITÄT PASSAU

Points & moments

1. Point set $X_\Omega = e^\Omega = \{x_\omega := e^\omega : \omega \in \Omega\}$. $\qquad e^\omega = e^{\omega_1} \cdots e^{\omega_s}$.
2. Discrete measure $\mu = \sum_\omega f_\omega \, \delta_{x_\omega}$.
3. Moments
$$\mu(\alpha) = \int x^\alpha \, d\mu(x)$$

Prony as moment problem

4. Square positive functionals.
5. Flat extensions $\rightarrow$ B. Mourrain.

Moment Problems

UNIVERSITÄT PASSAU

Points & moments

1. Point set $X_\Omega = e^\Omega = \{x_\omega := e^\omega : \omega \in \Omega\}$. $\qquad e^\omega = e^{\omega_1} \cdots e^{\omega_s}$.
2. Discrete measure $\mu = \sum_\omega f_\omega \, \delta_{x_\omega}$.
3. Moments
$$\mu(\alpha) = \int x^\alpha \, d\mu(x)$$

Prony as moment problem

4. Square positive functionals.
5. Flat extensions $\to$ B. Mourrain.

Moment Problems

UNIVERSITÄT PASSAU

Points & moments

1. Point set $X_\Omega = e^\Omega = \{x_\omega := e^\omega : \omega \in \Omega\}$. $\qquad e^\omega = e^{\omega_1} \cdots e^{\omega_s}$.
2. Discrete measure $\mu = \sum_\omega f_\omega \, \delta_{x_\omega}$.
3. Moments

$$\mu(\alpha) = \int x^\alpha \, d\mu(x) = \sum_{\omega \in \Omega} f_\omega \, x_\omega^\alpha$$

Prony as moment problem

4. Square positive functionals.
5. Flat extensions $\to$ B. Mourrain.

Moment Problems

UNIVERSITÄT PASSAU

Points & moments

1. Point set $X_\Omega = e^\Omega = \{x_\omega := e^\omega : \omega \in \Omega\}$. $\qquad e^\omega = e^{\omega_1} \cdots e^{\omega_s}$.
2. Discrete measure $\mu = \sum_\omega f_\omega \, \delta_{x_\omega}$.
3. Moments

$$\mu(\alpha) = \int x^\alpha \, d\mu(x) = \sum_{\omega \in \Omega} f_\omega \, e^{\omega^T \alpha}.$$

Prony as moment problem

4. Square positive functionals.
5. Flat extensions $\to$ B. Mourrain.

Moment Problems

Points & moments

1. Point set $X_\Omega = e^\Omega = \{x_\omega := e^\omega : \omega \in \Omega\}$. $\qquad e^\omega = e^{\omega_1} \cdots e^{\omega_s}$.
2. Discrete measure $\mu = \sum_\omega f_\omega \, \delta_{x_\omega}$.
3. Moments

$$\mu(\alpha) = \int x^\alpha \, d\mu(x) = \sum_{\omega \in \Omega} f_\omega \, e^{\omega^T \alpha}.$$

Prony as moment problem

1. Square positive functionals.
2. Flat extensions $\to$ B. Mourrain.

Moment Problems

UNIVERSITÄT PASSAU

Points & moments

1. Point set $X_\Omega = e^\Omega = \{x_\omega := e^\omega : \omega \in \Omega\}$. $\qquad e^\omega = e^{\omega_1} \cdots e^{\omega_s}$.
2. Discrete measure $\mu = \sum_\omega f_\omega \, \delta_{x_\omega}$.
3. Moments
$$\mu(\alpha) = \int x^\alpha \, d\mu(x) = \sum_{\omega \in \Omega} f_\omega \, e^{\omega^T \alpha}.$$

Prony as moment problem

1. Square positive functionals.
2. Flat extensions $\to$ B. Mourrain.

Moment Problems

Points & moments

1. Point set $X_\Omega = e^\Omega = \{x_\omega := e^\omega : \omega \in \Omega\}$. $\qquad e^\omega = e^{\omega_1} \cdots e^{\omega_s}$.
2. Discrete measure $\mu = \sum_\omega f_\omega \, \delta_{x_\omega}$.
3. Moments

$$\mu(\alpha) = \int x^\alpha \, d\mu(x) = \sum_{\omega \in \Omega} f_\omega \, e^{\omega^T \alpha}.$$

Prony as moment problem

1. Square positive functionals.
2. Flat extensions $\to$ B. Mourrain.

The History

Introduction

The History

Introduction

1795: *My name is R. Prony and I have a problem with alcohol.*

The History

Introduction

1795: *My name is R. Prony and I have a problem with alcohol.*

ESSAI EXPÉRIMENTAL

ET ANALYTIQUE

Sur les lois de la Dilatabilité des fluides élastiques et sur celles de la Force expansive de la vapeur de l'eau et de la vapeur de l'alkool, à différentes températures.

Par R. PRONY.

Remarks on Prony's Problem

Assumptions

1. Problem is *sparse*: $\#\Omega$ small.
2. A priori information: some number $N \geq \#\Omega$.

$$f(x) = \sum_{\omega \in \Omega} f_\omega \, e^{\omega^T x}.$$

Problem structure (Prony)

1. Frequencies: *nonlinear* problem.
2. Coefficients: *linear* problem.
3. Evaluation points: subgrid Λ of $\mathbb{Z}^s$
4. Shift of Λ irrelevant: $\Lambda \subset \mathbb{N}_0^s$.

Remarks on Prony's Problem

Assumptions

1. Problem is *sparse*: $\#\Omega$ small.
2. A priori information: some number $N \geq \#\Omega$.

$$f(x) = \sum_{\omega \in \Omega} f_\omega \, e^{\omega^T x}.$$

Problem structure (Prony)

1. Frequencies: *nonlinear* problem.
2. Coefficients: *linear* problem.
3. Evaluation points: subgrid Λ of $\mathbb{Z}^s$.
4. Shift of Λ irrelevant: $\Lambda \subset \mathbb{N}_0^s$.

Remarks on Prony's Problem

Assumptions

1. Problem is *sparse*: $\#\Omega$ small.
2. A priori information: some number $N \geq \#\Omega$.

$$f(x) = \sum_{\omega \in \Omega} f_\omega \, e^{\omega^T x}.$$

Problem structure (Prony)

1. Frequencies: *nonlinear* problem.
2. Coefficients: *linear* problem.
3. Evaluation points: subgrid Λ of $\mathbb{Z}^s$
4. Shift of Λ irrelevant: $\Lambda \subset \mathbb{N}_0^s$.

Remarks on Prony's Problem

UNIVERSITÄT PASSAU

Assumptions

1. Problem is *sparse*: $\#\Omega$ small.
2. A priori information: some number $N \geq \#\Omega$.

$$f(x) = \sum_{\omega \in \Omega} f_\omega \, e^{\omega^T x}.$$

Problem structure (Prony)

1. Frequencies: *nonlinear* problem.
2. Coefficients: *linear* problem.
3. Evaluation points: subgrid Λ of $\mathbb{Z}^s$
4. Shift of Λ irrelevant: $\Lambda \subset \mathbb{N}_0^s$.

Remarks on Prony's Problem

Assumptions

1. Problem is *sparse*: $\#\Omega$ small.
2. A priori information: some number $N \geq \#\Omega$.

$$f(x) = \sum_{\omega \in \Omega} f_\omega \, e^{\omega^T x}.$$

Problem structure (Prony)

1. Frequencies: *nonlinear* problem.
2. Coefficients: *linear* problem.
3. Evaluation points: subgrid Λ of $\mathbb{Z}^s$
4. Shift of Λ irrelevant: $\Lambda \subset \mathbb{N}_0^s$.

Remarks on Prony's Problem

Assumptions

1. Problem is *sparse*: $\#\Omega$ small.
2. A priori information: some number $N \geq \#\Omega$.

$$f(x) = \sum_{\omega \in \Omega} f_\omega \, e^{\omega^T x}.$$

Problem structure (Prony)

1. Frequencies: *nonlinear* problem.
2. Coefficients: *linear* problem.
3. Evaluation points: subgrid Λ of $\mathbb{Z}^s$
4. Shift of Λ irrelevant: $\Lambda \subset \mathbb{N}_0^s$.

Remarks on Prony's Problem

Assumptions

1. Problem is *sparse*: $\#\Omega$ small.
2. A priori information: some number $N \geq \#\Omega$.

$$f(x) = \sum_{\omega \in \Omega} f_\omega \, e^{\omega^T x}.$$

Problem structure (Prony)

1. Frequencies: *nonlinear* problem.
2. Coefficients: *linear* problem.
3. Evaluation points: subgrid Λ of $\mathbb{Z}^s$
4. Shift of Λ irrelevant: $\Lambda \subset \mathbb{N}_0^s$.

Remarks on Prony's Problem

Assumptions

1. Problem is *sparse*: $\#\Omega$ small.
2. A priori information: some number $N \geq \#\Omega$.

$$f(x) = \sum_{\omega \in \Omega} f_\omega \, e^{\omega^T x}.$$

Problem structure (Prony)

1. Frequencies: *nonlinear* problem.
2. Coefficients: *linear* problem.
3. Evaluation points: subgrid Λ of $\mathbb{Z}^s$ or $\Xi \mathbb{Z}^s$, $\Xi \in \mathbb{R}^{s \times s}$ nonsingular.
4. Shift of Λ irrelevant: $\Lambda \subset \mathbb{N}_0^s$.

Remarks on Prony's Problem

Assumptions

1. Problem is *sparse*: $\#\Omega$ small.
2. A priori information: some number $N \ge \#\Omega$.

$$f(x) = \sum_{\omega \in \Omega} f_\omega \, e^{\omega^T x}.$$

Problem structure (Prony)

1. Frequencies: *nonlinear* problem.
2. Coefficients: *linear* problem.
3. Evaluation points: subgrid Λ of $\mathbb{Z}^s$ or $\Xi \mathbb{Z}^s$, $\Xi \in \mathbb{R}^{s \times s}$ nonsingular.
4. Shift of Λ irrelevant: $\Lambda \subset \mathbb{N}_0^s$.

Notation

Polynomials

1. $\Pi = \mathbb{C}[x] = \mathbb{C}[x_1, \dots, x_s]$.
2. $\Pi_n = \left\{ p(x) = \sum_{|\alpha| \le n} p_\alpha x^\alpha : p_\alpha \in \mathbb{C} \right\}$ of *total degree* $\le n$.
3. *Total degree*: $\deg p = \max \{ |\alpha| : p_\alpha \neq 0 \}$.
4. $\Gamma_n := \{ \alpha \in \mathbb{N}_0^s : |\alpha| \le n \}$.
5. For $A \subset \mathbb{N}_0^s$ define $\Pi_A := \operatorname{span} \{ (\cdot)^\alpha : \alpha \in A \}$, i.e.,

$$\Pi_A \ni p = \sum_{\alpha \in A} p_\alpha (\cdot)^\alpha.$$

Coefficient vectors

$$p \simeq \boldsymbol{p} = [p_\alpha : \alpha \in \dots] \in \mathbb{C}^{\Gamma_n} \text{ or } \mathbb{C}^A.$$

Notation

Polynomials

1. $\Pi = \mathbb{C}[x] = \mathbb{C}[x_1, \dots, x_s]$.
2. $\Pi_n = \left\{ p(x) = \sum_{|\alpha| \le n} p_\alpha x^\alpha : p_\alpha \in \mathbb{C} \right\}$ of *total degree* $\le n$.
3. *Total degree*: $\deg p = \max\{|\alpha| : p_\alpha \ne 0\}$.
4. $\Gamma_n := \{\alpha \in \mathbb{N}_0^s : |\alpha| \le n\}$.
5. For $A \subset \mathbb{N}_0^s$ define $\Pi_A := \operatorname{span}\{(\cdot)^\alpha : \alpha \in A\}$, i.e.,

$$\Pi_A \ni p = \sum_{\alpha \in A} p_\alpha (\cdot)^\alpha.$$

Coefficient vectors

$$p \simeq \boldsymbol{p} = [p_\alpha : \alpha \in \dots] \in \mathbb{C}^{\Gamma_n} \text{ or } \mathbb{C}^A.$$

Notation

Polynomials

1. $\Pi = \mathbb{C}[x] = \mathbb{C}[x_1, \dots, x_s]$.
2. $\Pi_n = \left\{ p(x) = \sum_{|\alpha| \le n} p_\alpha x^\alpha : p_\alpha \in \mathbb{C} \right\}$ of *total degree* $\le n$.
3. *Total degree*: $\deg p = \max\{|\alpha| : p_\alpha \neq 0\}$.
4. $\Gamma_n := \{\alpha \in \mathbb{N}_0^s : |\alpha| \le n\}$.
5. For $A \subset \mathbb{N}_0^s$ define $\Pi_A := \operatorname{span}\{(\cdot)^\alpha : \alpha \in A\}$, i.e.,

$$\Pi_A \ni p = \sum_{\alpha \in A} p_\alpha (\cdot)^\alpha.$$

Coefficient vectors

$$p \simeq \boldsymbol{p} = [p_\alpha : \alpha \in \dots] \in \mathbb{C}^{\Gamma_n} \text{ or } \mathbb{C}^A.$$

Notation

Polynomials

1. $\Pi = \mathbb{C}[x] = \mathbb{C}[x_1, \dots, x_s]$.
2. $\Pi_n = \left\{ p(x) = \sum_{|\alpha| \le n} p_\alpha x^\alpha : p_\alpha \in \mathbb{C} \right\}$ of *total degree* $\le n$.
3. *Total degree*: $\deg p = \max\{|\alpha| : p_\alpha \neq 0\}$.
4. $\Gamma_n := \{\alpha \in \mathbb{N}_0^s : |\alpha| \le n\}$.
5. For $A \subset \mathbb{N}_0^s$ define $\Pi_A := \operatorname{span}\{(\cdot)^\alpha : \alpha \in A\}$, i.e.,
$$\Pi_A \ni p = \sum_{\alpha \in A} p_\alpha (\cdot)^\alpha.$$

Coefficient vectors

$$p \simeq \boldsymbol{p} = [p_\alpha : \alpha \in \dots] \in \mathbb{C}^{\Gamma_n} \text{ or } \mathbb{C}^A.$$

Notation

UNIVERSITÄT PASSAU

Polynomials

1. $\Pi = \mathbb{C}[x] = \mathbb{C}[x_1, \dots, x_s]$.
2. $\Pi_n = \left\{ p(x) = \sum_{|\alpha| \le n} p_\alpha x^\alpha : p_\alpha \in \mathbb{C} \right\}$ of *total degree* $\le n$.
3. *Total degree*: $\deg p = \max\{|\alpha| : p_\alpha \ne 0\}$.
4. $\Gamma_n := \{\alpha \in \mathbb{N}_0^s : |\alpha| \le n\}$.
5. For $A \subset \mathbb{N}_0^s$ define $\Pi_A := \operatorname{span}\{(\cdot)^\alpha : \alpha \in A\}$, i.e.,

$$\Pi_A \ni p = \sum_{\alpha \in A} p_\alpha (\cdot)^\alpha.$$

Coefficient vectors

$$p \simeq \mathbf{p} = [p_\alpha : \alpha \in \dots] \in \mathbb{C}^{\Gamma_n} \text{ or } \mathbb{C}^A.$$

Notation

UNIVERSITÄT PASSAU

Polynomials

1. $\Pi = \mathbb{C}[x] = \mathbb{C}[x_1, \dots, x_s]$.
2. $\Pi_n = \left\{ p(x) = \sum_{|\alpha| \le n} p_\alpha x^\alpha : p_\alpha \in \mathbb{C} \right\}$ of *total degree* $\le n$.
3. *Total degree*: $\deg p = \max \{ |\alpha| : p_\alpha \neq 0 \}$.
4. $\Gamma_n := \{ \alpha \in \mathbb{N}_0^s : |\alpha| \le n \}$.
5. For $A \subset \mathbb{N}_0^s$ define $\Pi_A := \operatorname{span} \{ (\cdot)^\alpha : \alpha \in A \}$, i.e.,

$$\Pi_A \ni p = \sum_{\alpha \in A} p_\alpha (\cdot)^\alpha.$$

Coefficient vectors

$$p \simeq \boldsymbol{p} = [p_\alpha : \alpha \in \dots] \in \mathbb{C}^{\Gamma_n} \text{ or } \mathbb{C}^A.$$

Notation

UNIVERSITÄT PASSAU

Polynomials

1. $\Pi = \mathbb{C}[x] = \mathbb{C}[x_1, \dots, x_s]$.
2. $\Pi_n = \left\{ p(x) = \sum_{|\alpha| \le n} p_\alpha \, x^\alpha : p_\alpha \in \mathbb{C} \right\}$ of *total degree* $\le n$.
3. *Total degree*: $\deg p = \max\{|\alpha| : p_\alpha \neq 0\}$.
4. $\Gamma_n := \{\alpha \in \mathbb{N}_0^s : |\alpha| \le n\}$.
5. For $A \subset \mathbb{N}_0^s$ define $\Pi_A := \operatorname{span}\{(\cdot)^\alpha : \alpha \in A\}$, i.e.,

$$\Pi_A \ni p = \sum_{\alpha \in A} p_\alpha \, (\cdot)^\alpha.$$

Coefficient vectors

$$p \simeq \boldsymbol{p} = [p_\alpha : \alpha \in \dots] \in \mathbb{C}^{\Gamma_n} \text{ or } \mathbb{C}^A.$$

The Prony Trick

A Hankel matrix

$$F_n := \left[f(\alpha+\beta) : \begin{array}{c} \alpha \in A \\ \beta \in B \end{array} \right] \in \mathbb{R}^{A \times B}$$

A computation …

For $p \in \Pi_B$ and $\alpha \in A$:

Consequence

1. $p(X_\Omega) = 0$ implies $F_n p = 0$.
2. Converse

The Prony Trick

A Hankel matrix

$$F_n := \left[f(\alpha+\beta) : \begin{array}{c} \alpha \in A \\ \beta \in B \end{array} \right] \in \mathbb{R}^{A \times B}$$

A computation …

For $p \in \Pi_B$ and $\alpha \in A$:

Consequence

1. $p(X_\Omega) = 0$ implies $F_n p = 0$.
2. Converse

The Prony Trick

A Hankel matrix

$$\boldsymbol{F}_n := \left[f(\alpha+\beta) : \begin{array}{c} \alpha \in A \\ \beta \in B \end{array} \right] \in \mathbb{R}^{A \times B}$$

A computation …

For $p \in \Pi_B$ and $\alpha \in A$:

$$(\boldsymbol{F}_n \boldsymbol{p})_\alpha =$$

Consequence

1. $p(X_\Omega) = 0$ implies $F_n p = 0$.
2. Converse

The Prony Trick

A Hankel matrix

$$F_n := \left[f(\alpha+\beta) : \begin{array}{c} \alpha \in A \\ \beta \in B \end{array} \right] \in \mathbb{R}^{A \times B}$$

A computation …

For $p \in \Pi_B$ and $\alpha \in A$:

$$(F_n p)_\alpha = \sum_{\beta \in B} f(\alpha+\beta)\, p_\beta$$

Consequence

1. $p(X_\Omega) = 0$ implies $F_n p = 0$.
2. Converse

The Prony Trick

A Hankel matrix

$$\boldsymbol{F}_n := \left[f(\alpha+\beta) : \begin{array}{c} \alpha \in A \\ \beta \in B \end{array} \right] \in \mathbb{R}^{A \times B}$$

A computation …

For $p \in \Pi_B$ and $\alpha \in A$:

$$(\boldsymbol{F}_n \boldsymbol{p})_\alpha = \sum_{\beta \in B} \sum_{\omega \in \Omega} f_\omega \, e^{\omega^T(\alpha+\beta)} \, p_\beta$$

Consequence

1. $p(X_\Omega) = 0$ implies $\boldsymbol{F}_n \boldsymbol{p} = 0$.
2. Converse

The Prony Trick

A Hankel matrix

$$\boldsymbol{F}_n := \left[f(\alpha+\beta) : \begin{array}{c} \alpha \in A \\ \beta \in B \end{array} \right] \in \mathbb{R}^{A \times B}$$

A computation …

For $p \in \Pi_B$ and $\alpha \in A$:

$$(\boldsymbol{F}_n \boldsymbol{p})_\alpha = \sum_{\beta \in B} \sum_{\omega \in \Omega} f_\omega \, e^{\omega^T \alpha} \, e^{\omega^T \beta} \, p_\beta$$

Consequence

1. $p(X_\Omega) = 0$ implies $F_n p = 0$.
2. Converse

The Prony Trick

A Hankel matrix

$$\boldsymbol{F}_n := \left[f(\alpha+\beta) : \begin{array}{l} \alpha \in A \\ \beta \in B \end{array} \right] \in \mathbb{R}^{A \times B}$$

A computation …

For $p \in \Pi_B$ and $\alpha \in A$:

$$(\boldsymbol{F}_n \boldsymbol{p})_\alpha = \sum_{\omega \in \Omega} f_\omega \, e^{\omega^T \alpha} \sum_{\beta \in B} e^{\omega^T \beta} \, p_\beta$$

Consequence

1. $p(X_\Omega) = 0$ implies $F_n p = 0$.
2. Converse

The Prony Trick

UNIVERSITÄT PASSAU

A Hankel matrix

$$F_n := \left[f(\alpha+\beta) : \begin{matrix} \alpha \in A \\ \beta \in B \end{matrix} \right] \in \mathbb{R}^{A \times B}$$

A computation …

For $p \in \Pi_B$ and $\alpha \in A$:

$$(F_n p)_\alpha = \sum_{\omega \in \Omega} f_\omega \, e^{\omega^T \alpha} \, p(e^\omega),$$

Consequence

1. $p(X_\Omega) = 0$ implies $F_n p = 0$.
2. Converse

The Prony Trick

A Hankel matrix

$$\boldsymbol{F}_n := \left[f(\alpha+\beta) : \begin{array}{c} \alpha \in A \\ \beta \in B \end{array} \right] \in \mathbb{R}^{A \times B}$$

A computation ...

For $p \in \Pi_B$ and $\alpha \in A$:

$$(\boldsymbol{F}_n \boldsymbol{p})_\alpha = \sum_{\omega \in \Omega} f_\omega \, e^{\omega^T \alpha} p(x_\omega).$$

Consequence

1. $p(X_\Omega) = 0$ implies $F_n p = 0$.
2. Converse

The Prony Trick

A Hankel matrix

$$F_n := \left[f(\alpha + \beta) : \begin{array}{c} \alpha \in A \\ \beta \in B \end{array} \right] \in \mathbb{R}^{A \times B}$$

A computation …

For $p \in \Pi_B$ and $\alpha \in A$:

$$(\boldsymbol{F}_n \boldsymbol{p})_\alpha = \sum_{\omega \in \Omega} f_\omega \, e^{\omega^T \alpha} p(x_\omega).$$

Consequence

1. $p(X_\Omega) = 0$ implies $\boldsymbol{F}_n \boldsymbol{p} = 0$.
2. Converse

The Prony Trick

UNIVERSITÄT PASSAU

A Hankel matrix

$$F_n := \left[f(\alpha+\beta) : \begin{array}{c} \alpha \in A \\ \beta \in B \end{array} \right] \in \mathbb{R}^{A \times B}$$

A computation …

For $p \in \Pi_B$ and $\alpha \in A$:

$$(\boldsymbol{F}_n \boldsymbol{p})_\alpha = \sum_{\omega \in \Omega} f_\omega \, e^{\omega^T \alpha} p(x_\omega).$$

Consequence

1. $p(X_\Omega) = 0$ implies $\boldsymbol{F}_n \boldsymbol{p} = 0$.
2. Converse

The Prony Trick

A Hankel matrix

$$F_n := \left[f(\alpha+\beta) : \begin{array}{c} \alpha \in A \\ \beta \in B \end{array} \right] \in \mathbb{R}^{A \times B}$$

A computation ...

For $p \in \Pi_B$ and $\alpha \in A$:

$$(F_n p)_\alpha = \sum_{\omega \in \Omega} f_\omega \, e^{\omega^T \alpha} p(x_\omega).$$

Consequence

1. $p(X_\Omega) = 0$ implies $F_n p = 0$.
2. Converse?

The Prony Trick

A Hankel matrix

$$F_n := \left[f(\alpha+\beta) : \begin{array}{c} \alpha \in A \\ \beta \in B \end{array} \right] \in \mathbb{R}^{A\times B}$$

A computation ...

For $p \in \Pi_B$ and $\alpha \in A$:

$$(F_n p)_\alpha = \sum_{\omega\in\Omega} f_\omega \, e^{\omega^T \alpha} \, p(x_\omega).$$

Consequence

1. $p(X_\Omega) = 0$ implies $F_n p = 0$.
2. Converse for $s > 1$?

The Prony Trick

A Hankel matrix

$$\boldsymbol{F}_n := \left[f(\alpha+\beta) : \begin{array}{c} \alpha \in A \\ \beta \in B \end{array} \right] \in \mathbb{R}^{A \times B}$$

A computation …

For $p \in \Pi_B$ and $\alpha \in A$:

$$(\boldsymbol{F}_n \boldsymbol{p})_\alpha = \sum_{\omega \in \Omega} f_\omega \, e^{\omega^T \alpha} \, p(x_\omega).$$

Consequence

1. $p(X_\Omega) = 0$ implies $\boldsymbol{F}_n \boldsymbol{p} = 0$.
2. Converse for $s > 1$? No!

The Prony Trick

UNIVERSITÄT PASSAU

A Hankel matrix

$$F_n := \left[f(\alpha+\beta) : \begin{array}{c} \alpha \in A \\ \beta \in B \end{array} \right] \in \mathbb{R}^{A \times B}$$

A computation …

For $p \in \Pi_B$ and $\alpha \in A$:

$$(\boldsymbol{F}_n \boldsymbol{p})_\alpha = \sum_{\omega \in \Omega} f_\omega \, e^{\omega^T \alpha} p(x_\omega).$$

Consequence

1. $p(X_\Omega) = 0$ implies $\boldsymbol{F}_n \boldsymbol{p} = 0$.
2. Converse for $s > 1$? Depends! Let's see …

Vandermonde Factorizations

Definition

Vandermonde matrix for $X \subset \mathbb{C}^s$ and $A \subset \Gamma$:

$$\boldsymbol{V}(X,A) := \left[x^\alpha : \begin{array}{c} x \in X \\ \alpha \in A \end{array} \right] \in \mathbb{C}^{X \times A}.$$

Factorizations (known from ESPRIT)

Sampling matrix (Hankel) for $A, B \subset \Gamma$:

$$\boldsymbol{F}_{A,B} := \left[f(\alpha+\beta) : \begin{array}{c} \alpha \in A \\ \beta \in B \end{array} \right]$$

1. $\boldsymbol{F}_{A,B} = \boldsymbol{V}(X_\Omega, A)^T \boldsymbol{F}_\Omega \boldsymbol{V}(X_\Omega, B)$.
2. $\boldsymbol{F}_{A,B} p = \boldsymbol{V}(X_\Omega, A)^T \boldsymbol{F}_\Omega \, p(X_\Omega), \qquad p \in \Pi_B$.

Vandermonde Factorizations

Definition

Vandermonde matrix for $X \subset \mathbb{C}^s$ and $A \subset \Gamma$:

$$\boldsymbol{V}(X,A) := \left[x^\alpha : \begin{array}{c} x \in X \\ \alpha \in A \end{array} \right] \in \mathbb{C}^{X \times A}.$$

Factorizations (known from ESPRIT)

Sampling matrix (Hankel) for $A, B \subset \Gamma$:

$$\boldsymbol{F}_{A,B} := \left[f(\alpha+\beta) : \begin{array}{c} \alpha \in A \\ \beta \in B \end{array} \right]$$

1. $\boldsymbol{F}_{A,B} = \boldsymbol{V}(X_\Omega, A)^T \boldsymbol{F}_\Omega \boldsymbol{V}(X_\Omega, B)$.
2. $\boldsymbol{F}_{A,B}\boldsymbol{p} = \boldsymbol{V}(X_\Omega, A)^T \boldsymbol{F}_\Omega \, p(X_\Omega), \qquad p \in \Pi_B$.

Vandermonde Factorizations

UNIVERSITÄT PASSAU

Definition

Vandermonde matrix for $X \subset \mathbb{C}^s$ and $A \subset \Gamma$:

$$\boldsymbol{V}(X,A) := \left[x^\alpha : \begin{array}{l} x \in X \\ \alpha \in A \end{array} \right] \in \mathbb{C}^{X \times A}.$$

Factorizations (known from ESPRIT)

Sampling matrix (Hankel) for $A, B \subset \Gamma$:

$$\boldsymbol{F}_{A,B} := \left[f(\alpha+\beta) : \begin{array}{l} \alpha \in A \\ \beta \in B \end{array} \right]$$

1. $\boldsymbol{F}_{A,B} = \boldsymbol{V}(X_\Omega, A)^T \boldsymbol{F}_\Omega \boldsymbol{V}(X_\Omega, B).$
2. $\boldsymbol{F}_{A,B} \boldsymbol{p} = \boldsymbol{V}(X_\Omega, A)^T \boldsymbol{F}_\Omega \, p(X_\Omega), \qquad p \in \Pi_B.$

Vandermonde Factorizations

Definition

Vandermonde matrix for $X \subset \mathbb{C}^s$ and $A \subset \Gamma$:

$$\boldsymbol{V}(X,A) := \left[x^\alpha : \begin{array}{c} x \in X \\ \alpha \in A \end{array} \right] \in \mathbb{C}^{X \times A}.$$

Factorizations (known from ESPRIT)

Sampling matrix (Hankel) for $A, B \subset \Gamma$:

$$\boldsymbol{F}_{A,B} := \left[f(\alpha+\beta) : \begin{array}{c} \alpha \in A \\ \beta \in B \end{array} \right], \qquad \boldsymbol{F}_\Omega = \operatorname{diag}\left[f_\omega : \omega \in \Omega \right],$$

1. $\boldsymbol{F}_{A,B} = \boldsymbol{V}(X_\Omega, A)^T \boldsymbol{F}_\Omega \boldsymbol{V}(X_\Omega, B)$.
2. $\boldsymbol{F}_{A,B}\boldsymbol{p} = \boldsymbol{V}(X_\Omega, A)^T \boldsymbol{F}_\Omega\, p(X_\Omega), \qquad p \in \Pi_B$.

Vandermonde Factorizations

Definition

Vandermonde matrix for $X \subset \mathbb{C}^s$ and $A \subset \Gamma$:

$$\boldsymbol{V}(X,A) := \left[x^\alpha : \begin{array}{c} x \in X \\ \alpha \in A \end{array} \right] \in \mathbb{C}^{X \times A}.$$

Factorizations (known from ESPRIT)

Sampling matrix (Hankel) for $A, B \subset \Gamma$:

$$\boldsymbol{F}_{A,B} := \left[f(\alpha+\beta) : \begin{array}{c} \alpha \in A \\ \beta \in B \end{array} \right], \qquad \boldsymbol{F}_\Omega = \operatorname{diag} \left[f_\omega : \omega \in \Omega \right],$$

1. $\boldsymbol{F}_{A,B} = \boldsymbol{V}(X_\Omega, A)^T \boldsymbol{F}_\Omega \boldsymbol{V}(X_\Omega, B)$.
2. $\boldsymbol{F}_{A,B}\boldsymbol{p} = \boldsymbol{V}(X_\Omega, A)^T \boldsymbol{F}_\Omega \, p(X_\Omega), \qquad p \in \Pi_B$.

Interpolation Spaces

UNIVERSITÄT PASSAU

Definition

1. $\mathscr{P} \subset \Pi$ *interpolation space* for $X \subset \mathbb{C}^s$:
   for any $q \in \Pi$ there is $p \in \mathscr{P}$ such that $p(X) = q(X)$.
2. *Degree reducing* interpolation space: $\deg p \le \deg q$.
3. $A \subset \mathbb{N}_0^s$ *interpolation set*: Π_A is interpolation space.
4. *Degree reducing* interpolation set:

Interpolation folklore

1. A is interpolation set for X iff $\operatorname{rank} V(X, A) = \# X$.
2. $\# X = \# A$: interpolation iff $V(X, A)$ nonsingular.
3. For fixed A, generic case: nonsingular $V(\cdot, A)$ open & dense.
4. Degree reducing: ideal interpolation, Gröber-/H–bases …

Interpolation Spaces

Definition

1. $\mathscr{P} \subset \Pi$ *interpolation space* for $X \subset \mathbb{C}^s$:
   for any $q \in \Pi$ there is $p \in \mathscr{P}$ such that $p(X) = q(X)$.
2. *Degree reducing* interpolation space: $\deg p \le \deg q$.
3. $A \subset \mathbb{N}_0^s$ *interpolation set*: Π_A is interpolation space.
4. *Degree reducing* interpolation set:

Interpolation folklore

1. A is interpolation set for X iff rank $V(X, A) = \#X$.
2. $\#X = \#A$: interpolation iff $V(X, A)$ nonsingular.
3. For fixed A, generic case: nonsingular $V(\cdot, A)$ open & dense.
4. Degree reducing: ideal interpolation, Gröber-/H–bases . . .

Interpolation Spaces

Definition

1. $\mathscr{P} \subset \Pi$ *interpolation space* for $X \subset \mathbb{C}^s$:
   for any $q \in \Pi$ there is $p \in \mathscr{P}$ such that $p(X) = q(X)$.
2. *Degree reducing* interpolation space: $\deg p \leq \deg q$.
3. $A \subset \mathbb{N}_0^s$ *interpolation set*: Π_A is interpolation space.
4. *Degree reducing* interpolation set:

Interpolation folklore

1. A is interpolation set for X iff $\operatorname{rank} V(X, A) = \#X$.
2. $\#X = \#A$: interpolation iff $V(X, A)$ nonsingular.
3. For fixed A, generic case: nonsingular $V(\cdot, A)$ open & dense.
4. Degree reducing: ideal interpolation, Gröber-/H–bases ...

Interpolation Spaces

UNIVERSITÄT PASSAU

Definition

1. $\mathscr{P} \subset \Pi$ *interpolation space* for $X \subset \mathbb{C}^s$:
   for any $q \in \Pi$ there is $p \in \mathscr{P}$ such that $p(X) = q(X)$.
2. *Degree reducing* interpolation space: $\deg p \le \deg q$.
3. $A \subset \mathbb{N}_0^s$ *interpolation set*: Π_A is interpolation space.
4. *Degree reducing* interpolation set:

Interpolation folklore

1. A is interpolation set for X iff $\operatorname{rank} V(X,A) = \#X$.
2. $\#X = \#A$: interpolation iff $V(X,A)$ nonsingular.
3. For fixed A, generic case: nonsingular $V(\cdot,A)$ open & dense.
4. Degree reducing: ideal interpolation, Gröber-/H–bases ...

Interpolation Spaces

UNIVERSITÄT PASSAU

Definition

1. $\mathscr{P} \subset \Pi$ *interpolation space* for $X \subset \mathbb{C}^s$:
   for any $q \in \Pi$ there is $p \in \mathscr{P}$ such that $p(X) = q(X)$.
2. *Degree reducing* interpolation space: $\deg p \leq \deg q$.
3. $A \subset \mathbb{N}_0^s$ *interpolation set*: Π_A is interpolation space.
4. *Degree reducing* interpolation set:

Interpolation folklore

1. A is interpolation set for X iff $\operatorname{rank} V(X, A) = \#X$.
2. $\#X = \#A$: interpolation iff $V(X, A)$ nonsingular.
3. For fixed A, generic case: nonsingular $V(\cdot, A)$ open & dense.
4. Degree reducing: ideal interpolation, Gröber-/H–bases . . .

Interpolation Spaces

Definition

1. $\mathscr{P} \subset \Pi$ *interpolation space* for $X \subset \mathbb{C}^s$:
   for any $q \in \Pi$ there is $p \in \mathscr{P}$ such that $p(X) = q(X)$.
2. *Degree reducing* interpolation space: $\deg p \leq \deg q$.
3. $A \subset \mathbb{N}_0^s$ *interpolation set*: Π_A is interpolation space.
4. *Degree reducing* interpolation set: Exercise

Interpolation folklore

1. A is interpolation set for X iff $\operatorname{rank} V(X, A) = \#X$.
2. $\#X = \#A$: interpolation iff $V(X, A)$ nonsingular.
3. For fixed A, generic case: nonsingular $V(\cdot, A)$ open & dense.
4. Degree reducing: ideal interpolation, Gröber-/H–bases ...

Interpolation Spaces

UNIVERSITÄT PASSAU

Definition

1. $\mathscr{P} \subset \Pi$ *interpolation space* for $X \subset \mathbb{C}^s$:
   for any $q \in \Pi$ there is $p \in \mathscr{P}$ such that $p(X) = q(X)$.
2. *Degree reducing* interpolation space: $\deg p \le \deg q$.
3. $A \subset \mathbb{N}_0^s$ *interpolation set*: Π_A is interpolation space.
4. *Degree reducing* interpolation set: Exercise

Interpolation folklore

1. A is interpolation set for X iff $\operatorname{rank} V(X, A) = \#X$.
2. $\#X = \#A$: interpolation iff $V(X, A)$ nonsingular.
3. For fixed A, generic case: nonsingular $V(\cdot, A)$ open & dense.
4. Degree reducing: ideal interpolation, Gröber-/H–bases …

Interpolation Spaces

Definition

1. $\mathscr{P} \subset \Pi$ *interpolation space* for $X \subset \mathbb{C}^s$:
   for any $q \in \Pi$ there is $p \in \mathscr{P}$ such that $p(X) = q(X)$.
2. *Degree reducing* interpolation space: $\deg p \le \deg q$.
3. $A \subset \mathbb{N}_0^s$ *interpolation set*: Π_A is interpolation space.
4. *Degree reducing* interpolation set: Exercise

Interpolation folklore

1. A is interpolation set for X iff $\operatorname{rank} \boldsymbol{V}(X, A) = \#X$.
2. $\#X = \#A$: interpolation iff $V(X, A)$ nonsingular.
3. For fixed A, generic case: nonsingular $V(\cdot, A)$ open & dense.
4. Degree reducing: ideal interpolation, Gröber-/H–bases …

Interpolation Spaces

UNIVERSITÄT PASSAU

Definition

1. $\mathscr{P} \subset \Pi$ *interpolation space* for $X \subset \mathbb{C}^s$:
   for any $q \in \Pi$ there is $p \in \mathscr{P}$ such that $p(X) = q(X)$.
2. *Degree reducing* interpolation space: $\deg p \le \deg q$.
3. $A \subset \mathbb{N}_0^s$ *interpolation set*: Π_A is interpolation space.
4. *Degree reducing* interpolation set: Exercise

Interpolation folklore

1. A is interpolation set for X iff rank $\boldsymbol{V}(X, A) = \#X$.
2. $\#X = \#A$: interpolation iff $\boldsymbol{V}(X, A)$ nonsingular.
3. For fixed A, generic case: nonsingular $V(\cdot, A)$ open & dense.
4. Degree reducing: ideal interpolation, Gröber-/H–bases ...

Interpolation Spaces

UNIVERSITÄT PASSAU

Definition

1. $\mathscr{P} \subset \Pi$ *interpolation space* for $X \subset \mathbb{C}^s$:
   for any $q \in \Pi$ there is $p \in \mathscr{P}$ such that $p(X) = q(X)$.
2. *Degree reducing* interpolation space: $\deg p \le \deg q$.
3. $A \subset \mathbb{N}_0^s$ *interpolation set*: Π_A is interpolation space.
4. *Degree reducing* interpolation set: Exercise

Interpolation folklore

1. A is interpolation set for X iff $\operatorname{rank} \boldsymbol{V}(X, A) = \#X$.
2. $\#X = \#A$: *unique* interpolation iff $\boldsymbol{V}(X, A)$ nonsingular.
3. For fixed A, generic case: nonsingular $V(\cdot, A)$ open & dense.
4. Degree reducing: ideal interpolation, Gröber-/H–bases …

Interpolation Spaces

UNIVERSITÄT PASSAU

Definition

1. $\mathscr{P} \subset \Pi$ *interpolation space* for $X \subset \mathbb{C}^s$:
   for any $q \in \Pi$ there is $p \in \mathscr{P}$ such that $p(X) = q(X)$.
2. *Degree reducing* interpolation space: $\deg p \le \deg q$.
3. $A \subset \mathbb{N}_0^s$ *interpolation set*: Π_A is interpolation space.
4. *Degree reducing* interpolation set: Exercise

Interpolation folklore

1. A is interpolation set for X iff $\operatorname{rank} \boldsymbol{V}(X, A) = \#X$.
2. $\#X = \#A$: *unique* interpolation iff $\boldsymbol{V}(X, A)$ nonsingular.
3. For fixed A, generic case: nonsingular $\boldsymbol{V}(\cdot, A)$ open & dense.
4. Degree reducing: ideal interpolation, Gröber-/H–bases …

Interpolation Spaces

Definition

1. $\mathscr{P} \subset \Pi$ *interpolation space* for $X \subset \mathbb{C}^s$:
   for any $q \in \Pi$ there is $p \in \mathscr{P}$ such that $p(X) = q(X)$.
2. *Degree reducing* interpolation space: $\deg p \le \deg q$.
3. $A \subset \mathbb{N}_0^s$ *interpolation set*: Π_A is interpolation space.
4. *Degree reducing* interpolation set: Exercise

Not so much **Interpolation folklore**

1. A is interpolation set for X iff $\operatorname{rank} \boldsymbol{V}(X, A) = \#X$.
2. $\#X = \#A$: *unique* interpolation iff $\boldsymbol{V}(X, A)$ nonsingular.
3. For fixed A, generic case: nonsingular $\boldsymbol{V}(\cdot, A)$ open & dense.
4. Degree reducing: ideal interpolation, Gröber-/H–bases …

Vandermonde Consequences

The Prony ideal

1. $I_\Omega := \{p \in \Pi : p(X_\Omega) = 0\}$.
2. Replacement of univariate *Prony polynomial*.

$$
\begin{aligned}
\boldsymbol{F}_{A,B} &= \boldsymbol{V}(X_\Omega, A)^T \boldsymbol{F}_\Omega \boldsymbol{V}(X_\Omega, B), \\
\boldsymbol{F}_{A,B}\boldsymbol{p} &= \boldsymbol{V}(X_\Omega, A)^T \boldsymbol{F}_\Omega \, p(X_\Omega), \qquad p \in \Pi_B.
\end{aligned}
$$

Theorem

1. Reconstruct F_Ω from $F_{A,B}$ iff A, B are interpolation sets for X_Ω.
2. If A is interpolation set for X_Ω then
   - $\ker F_{A,B} \simeq I_\Omega \cap \Pi_B$.
   - $n \mapsto \operatorname{rank} F_{A,\Gamma_n}$ is the *affine Hilbert function* of I_Ω.

Vandermonde Consequences

The Prony ideal

1. $I_\Omega := \{p \in \Pi : p(X_\Omega) = 0\}$.
2. Replacement of univariate *Prony polynomial*.

$$
\begin{aligned}
\boldsymbol{F}_{A,B} &= \boldsymbol{V}(X_\Omega, A)^T \boldsymbol{F}_\Omega \boldsymbol{V}(X_\Omega, B), \\
\boldsymbol{F}_{A,B}\boldsymbol{p} &= \boldsymbol{V}(X_\Omega, A)^T \boldsymbol{F}_\Omega \, p(X_\Omega), \qquad p \in \Pi_B.
\end{aligned}
$$

Theorem

1. Reconstruct F_Ω from $F_{A,B}$ iff A, B are interpolation sets for X_Ω.
2. If A is interpolation set for X_Ω then
   - $\ker F_{A,B} \simeq I_\Omega \cap \Pi_B$.
   - $n \mapsto \operatorname{rank} F_{A,\Gamma_n}$ is the *affine Hilbert function* of I_Ω.

Vandermonde Consequences

The Prony ideal

1. $I_\Omega := \{p \in \Pi : p(X_\Omega) = 0\}$.
2. Replacement of univariate *Prony polynomial*.

$$
\begin{aligned}
F_{A,B} &= V(X_\Omega, A)^T F_\Omega V(X_\Omega, B), \\
F_{A,B} p &= V(X_\Omega, A)^T F_\Omega p(X_\Omega), \qquad p \in \Pi_B.
\end{aligned}
$$

Theorem

1. Reconstruct F_Ω from $F_{A,B}$ iff A, B are interpolation sets for X_Ω.
2. If A is interpolation set for X_Ω then
   - $\ker F_{A,B} \simeq I_\Omega \cap \Pi_B$.
   - $n \mapsto \operatorname{rank} F_{A,\Gamma_n}$ is the *affine Hilbert function* of I_Ω.

Vandermonde Consequences

The Prony ideal

1. $I_\Omega := \{p \in \Pi : p(X_\Omega) = 0\}$. Zero dimensional & radical.
2. Replacement of univariate *Prony polynomial*.

$$\begin{aligned} \boldsymbol{F}_{A,B} &= \boldsymbol{V}(X_\Omega, A)^T \boldsymbol{F}_\Omega \, \boldsymbol{V}(X_\Omega, B), \\ \boldsymbol{F}_{A,B}\boldsymbol{p} &= \boldsymbol{V}(X_\Omega, A)^T \boldsymbol{F}_\Omega \, p(X_\Omega), \qquad p \in \Pi_B. \end{aligned}$$

Theorem

1. Reconstruct F_Ω from $F_{A,B}$ iff A, B are interpolation sets for X_Ω.
2. If A is interpolation set for X_Ω then
   - $\ker F_{A,B} \simeq I_\Omega \cap \Pi_B$.
   - $n \mapsto \operatorname{rank} F_{A,\Gamma_n}$ is the *affine Hilbert function* of I_Ω.

Vandermonde Consequences

The Prony ideal

1. $I_\Omega := \{p \in \Pi : p(X_\Omega) = 0\}$. Zero dimensional & radical.
2. Replacement of univariate *Prony polynomial*.

$$
\begin{aligned}
\boldsymbol{F}_{A,B} &= \boldsymbol{V}(X_\Omega, A)^T \boldsymbol{F}_\Omega \boldsymbol{V}(X_\Omega, B), \\
\boldsymbol{F}_{A,B}\boldsymbol{p} &= \boldsymbol{V}(X_\Omega, A)^T \boldsymbol{F}_\Omega \, p(X_\Omega), \qquad p \in \Pi_B.
\end{aligned}
$$

Theorem

1. Reconstruct F_Ω from $F_{A,B}$ iff A, B are interpolation sets for X_Ω.
2. If A is interpolation set for X_Ω then

Vandermonde Consequences

The Prony ideal

1. $I_\Omega := \{p \in \Pi : p(X_\Omega) = 0\}$. Zero dimensional & radical.
2. Replacement of univariate *Prony polynomial*.

$$\begin{aligned} \boldsymbol{F}_{A,B} &= \boldsymbol{V}(X_\Omega, A)^T \boldsymbol{F}_\Omega \boldsymbol{V}(X_\Omega, B), \\ \boldsymbol{F}_{A,B}\boldsymbol{p} &= \boldsymbol{V}(X_\Omega, A)^T \boldsymbol{F}_\Omega \, p(X_\Omega), \qquad p \in \Pi_B. \end{aligned}$$

Theorem

1. Reconstruct F_Ω from $F_{A,B}$ iff A, B are interpolation sets for X_Ω.
2. If A is interpolation set for X_Ω then
   - $\ker F_{A,B} \simeq I_\Omega \cap \Pi_B$.
   - $n \mapsto \operatorname{rank} F_{A,\Gamma_n}$ is the *affine Hilbert function* of I_Ω.

Vandermonde Consequences

The Prony ideal

1. $I_\Omega := \{p \in \Pi : p(X_\Omega) = 0\}$. Zero dimensional & radical.
2. Replacement of univariate *Prony polynomial*.

$$
\begin{aligned}
\boldsymbol{F}_{A,B} &= \boldsymbol{V}(X_\Omega, A)^T \boldsymbol{F}_\Omega \boldsymbol{V}(X_\Omega, B), \\
\boldsymbol{F}_{A,B}\boldsymbol{p} &= \boldsymbol{V}(X_\Omega, A)^T \boldsymbol{F}_\Omega \, p(X_\Omega), \qquad p \in \Pi_B.
\end{aligned}
$$

Theorem

1. Reconstruct $\boldsymbol{F}_\Omega$ from $\boldsymbol{F}_{A,B}$ iff A, B are interpolation sets for X_Ω.
2. If A is interpolation set for X_Ω then
   - $\ker \boldsymbol{F}_{A,B} \simeq I_\Omega \cap \Pi_B$.
   - $n \mapsto \operatorname{rank} \boldsymbol{F}_{A,\Gamma_n}$ is the *affine Hilbert function* of I_Ω.

Vandermonde Consequences

The Prony ideal

1. $I_\Omega := \{p \in \Pi : p(X_\Omega) = 0\}$. Zero dimensional & radical.
2. Replacement of univariate *Prony polynomial*.

$$\begin{aligned} \boldsymbol{F}_{A,B} &= \boldsymbol{V}(X_\Omega, A)^T \boldsymbol{F}_\Omega \boldsymbol{V}(X_\Omega, B), \\ \boldsymbol{F}_{A,B}\boldsymbol{p} &= \boldsymbol{V}(X_\Omega, A)^T \boldsymbol{F}_\Omega \, p(X_\Omega), \qquad p \in \Pi_B. \end{aligned}$$

Theorem

1. Reconstruct $\boldsymbol{F}_\Omega$ from $\boldsymbol{F}_{A,B}$ iff A, B are interpolation sets for X_Ω.
2. If A is interpolation set for X_Ω then
   - $\ker \boldsymbol{F}_{A,B} \simeq I_\Omega \cap \Pi_B$.
   - $n \mapsto \operatorname{rank} \boldsymbol{F}_{A,\Gamma_n}$ is the *affine Hilbert function* of I_Ω.

Vandermonde Consequences

UNIVERSITÄT PASSAU

The Prony ideal

1. $I_\Omega := \{p \in \Pi : p(X_\Omega) = 0\}$. Zero dimensional & radical.
2. Replacement of univariate *Prony polynomial*.

$$\begin{aligned} \boldsymbol{F}_{A,B} &= \boldsymbol{V}(X_\Omega, A)^T \boldsymbol{F}_\Omega \boldsymbol{V}(X_\Omega, B), \\ \boldsymbol{F}_{A,B}\boldsymbol{p} &= \boldsymbol{V}(X_\Omega, A)^T \boldsymbol{F}_\Omega \, p(X_\Omega), \qquad p \in \Pi_B. \end{aligned}$$

Theorem

1. Reconstruct $\boldsymbol{F}_\Omega$ from $\boldsymbol{F}_{A,B}$ iff A, B are interpolation sets for X_Ω.
2. If A is interpolation set for X_Ω then
   - $\ker \boldsymbol{F}_{A,B} \simeq I_\Omega \cap \Pi_B$.
   - $n \mapsto \operatorname{rank} \boldsymbol{F}_{A,\Gamma_n}$ is the *affine Hilbert function* of I_Ω.

Prony & Interpolation

Needed for solution

1. Find interpolation set A for X_Ω.
2. **But:** X_Ω is unknown.

The generic case

1. Generic interpolation space

$$\Pi_n \qquad \text{where} \qquad \binom{n-1+s}{s} < \#\Omega \leq \binom{n+s}{s}.$$

2. $\ker F_{\Gamma_n,\Gamma_{n+1}}$: nonlinear homogeneous equations.
3. $\leq 2^s \#\Omega$ samples, 2^s best constant.
4. **Linear** in $\#\Omega$, not like $(\#\Omega)^s$.

Prony & Interpolation

Needed for solution

1. Find interpolation set A for X_Ω.
2. **But:** X_Ω is unknown.

The generic case

1. Generic interpolation space

$$\Pi_n \qquad \text{where} \qquad \binom{n-1+s}{s} < \#\Omega \le \binom{n+s}{s}.$$

2. $\ker F_{\Gamma_n,\Gamma_{n+1}}$: nonlinear homogeneous equations.
3. $\le 2^s \#\Omega$ samples, 2^s best constant.
4. **Linear** in $\#\Omega$, not like $(\#\Omega)^s$.

Prony & Interpolation

Needed for solution

1. Find interpolation set A for X_Ω.
2. **But:** X_Ω is unknown.

The generic case

1. Generic interpolation space

$$\Pi_n \qquad \text{where} \qquad \binom{n-1+s}{s} < \#\Omega \le \binom{n+s}{s}.$$

2. $\ker F_{\Gamma_n,\Gamma_{n+1}}$: nonlinear homogeneous equations.
3. $\le 2^s \#\Omega$ samples, 2^s best constant.
4. **Linear** in $\#\Omega$, not like $(\#\Omega)^s$.

Prony & Interpolation

Needed for solution

1. Find interpolation set A for X_Ω.
2. **But:** X_Ω is unknown.

The generic case

1. Generic interpolation space

$$\Pi_n \qquad \text{where} \qquad \binom{n-1+s}{s} < \#\Omega \le \binom{n+s}{s}.$$

2. $\ker F_{\Gamma_n,\Gamma_{n+1}}$: nonlinear homogeneous equations.
3. $\le 2^s \#\Omega$ samples, 2^s best constant.
4. **Linear** in $\#\Omega$, not like $(\#\Omega)^s$.

Prony & Interpolation

Needed for solution

1. Find interpolation set A for X_Ω.
2. **But:** X_Ω is unknown.

The generic case

1. Generic interpolation space

$$\Pi_n \qquad \text{where} \qquad \binom{n-1+s}{s} < \#\Omega \le \binom{n+s}{s}.$$

2. $\ker F_{\Gamma_n,\Gamma_{n+1}}$: nonlinear homogeneous equations.
3. $\le 2^s \#\Omega$ samples, 2^s best constant.
4. **Linear** in $\#\Omega$, not like $(\#\Omega)^s$.

Prony & Interpolation

Needed for solution

1. Find interpolation set A for X_Ω.
2. **But:** X_Ω is unknown.

The generic case

1. Generic interpolation space

$$\Pi_n \qquad \text{where} \qquad \binom{n-1+s}{s} < \#\Omega \le \binom{n+s}{s}.$$

2. $\ker \boldsymbol{F}_{\Gamma_n,\Gamma_{n+1}}$: nonlinear homogeneous equations.
3. $\le 2^s \#\Omega$ samples, 2^s best constant.
4. **Linear** in $\#\Omega$, not like $(\#\Omega)^s$.

Prony & Interpolation

Needed for solution

1. Find interpolation set A for X_Ω.
2. **But:** X_Ω is unknown.

The generic case

1. Generic interpolation space

$$\Pi_n \qquad \text{where} \qquad \binom{n-1+s}{s} < \#\Omega \le \binom{n+s}{s}.$$

2. $\ker \boldsymbol{F}_{\Gamma_n,\Gamma_{n+1}}$: nonlinear homogeneous equations.
3. $\le 2^s \#\Omega$ samples, 2^s best constant.
4. **Linear** in $\#\Omega$, not like $(\#\Omega)^s$.

Prony & Interpolation

Needed for solution

1. Find interpolation set A for X_Ω.
2. **But:** X_Ω is unknown.

The generic case

1. Generic interpolation space

$$\Pi_n \qquad \text{where} \qquad \binom{n-1+s}{s} < \#\Omega \le \binom{n+s}{s}.$$

2. $\ker \boldsymbol{F}_{\Gamma_n,\Gamma_{n+1}}$: nonlinear homogeneous equations.
3. $\le 2^s \#\Omega$ samples, 2^s best constant.
4. **Linear** in $\#\Omega$, not like $(\#\Omega)^s$.

Prony & Interpolation

Needed for solution

1. Find interpolation set A for X_Ω.
2. **But:** X_Ω is unknown and life is not generic.

The generic case is of little value

1. Generic interpolation space
$$\Pi_n \qquad \text{where} \qquad \binom{n-1+s}{s} < \#\Omega \le \binom{n+s}{s}.$$
2. $\ker \boldsymbol{F}_{\Gamma_n,\Gamma_{n+1}}$: nonlinear homogeneous equations.
3. $\le 2^s \#\Omega$ samples, 2^s best constant.
4. **Linear** in $\#\Omega$, not like $(\#\Omega)^s$.

Minimal Sampling

Two things needed

1. Interpolation space for *all* sets X with $\#X \leq N$.
2. Efficient polynomial solver.

Choices for the set A

1. Interpolation guaranteed for $A = \Gamma_{N-1}$.
2. Minimal for $s = 1$.
3. $\dim \Pi_{N-1} = \binom{N-1+s}{s} \simeq N^s / s!$.
4. F_{Γ_N,Γ_N} too large: e.g. $s = 13$, $N = 200$ yields $10^{20} \times 10^{20}$.

Notabene: naive algorithm

Build $F_n := F_{\Gamma_n,\Gamma_n}$, $n = 0, 1, 2, \dots,$ until $\operatorname{rank} F_n = \operatorname{rank} F_{n+1}$.

Minimal Sampling

Two things needed

1. Interpolation space for *all* sets X with $\#X \le N$.
2. Efficient polynomial solver.

Choices for the set A

1. Interpolation guaranteed for $A = \Gamma_{N-1}$.
2. Minimal for $s = 1$.
3. $\dim \Pi_{N-1} = \binom{N-1+s}{s} \simeq N^s/s!$.
4. F_{Γ_N,Γ_N} too large: e.g. $s = 13$, $N = 200$ yields $10^{20} \times 10^{20}$.

Notabene: naive algorithm

Build $F_n := F_{\Gamma_n,\Gamma_n}$, $n = 0, 1, 2, \ldots,$ until $\operatorname{rank} F_n = \operatorname{rank} F_{n+1}$.

Minimal Sampling

Two things needed

1. Interpolation space for *all* sets X with $\#X \le N$.
2. Efficient polynomial solver.

Choices for the set A

1. Interpolation guaranteed for $A = \Gamma_{N-1}$.
2. Minimal for $s = 1$.
3. $\dim \Pi_{N-1} = \binom{N-1+s}{s} \simeq N^s/s!$.
4. F_{Γ_N,Γ_N} too large: e.g. $s = 13$, $N = 200$ yields $10^{20} \times 10^{20}$.

Notabene: naive algorithm

Build $F_n := F_{\Gamma_n,\Gamma_n}$, $n = 0, 1, 2, \ldots,$ until $\operatorname{rank} F_n = \operatorname{rank} F_{n+1}$.

Minimal Sampling

Two things needed

1. Interpolation space for *all* sets X with $\#X \leq N$.
2. Efficient polynomial solver.

Choices for the set A

1. Interpolation guaranteed for $A = \Gamma_{N-1}$.
2. Minimal for $s = 1$.
3. $\dim \Pi_{N-1} = \binom{N-1+s}{s} \simeq N^s / s!$.
4. F_{Γ_N, Γ_N} too large: e.g. $s = 13$, $N = 200$ yields $10^{20} \times 10^{20}$.

Notabene: naive algorithm

Build $F_n := F_{\Gamma_n, \Gamma_n}$, $n = 0, 1, 2, \ldots,$ until $\operatorname{rank} F_n = \operatorname{rank} F_{n+1}$.

Minimal Sampling

Two things needed

1. Interpolation space for *all* sets X with $\#X \le N$.
2. Efficient polynomial solver.

Choices for the set A

1. Interpolation guaranteed for $A = \Gamma_{N-1}$.
2. Minimal for $s = 1$.
3. $\dim \Pi_{N-1} = \binom{N-1+s}{s} \simeq N^s/s!$.
4. F_{Γ_N,Γ_N} too large: e.g. $s = 13$, $N = 200$ yields $10^{20} \times 10^{20}$.

Notabene: naive algorithm

Build $F_n := F_{\Gamma_n,\Gamma_n}$, $n = 0, 1, 2, \dots,$ until $\operatorname{rank} F_n = \operatorname{rank} F_{n+1}$.

Minimal Sampling

Two things needed

1. Interpolation space for *all* sets X with $\#X \le N$.
2. Efficient polynomial solver.

Choices for the set A

1. Interpolation guaranteed for $A = \Gamma_{N-1}$.
2. Minimal for $s = 1$.
3. $\dim \Pi_{N-1} = \binom{N-1+s}{s} \simeq N^s/s!$.
4. F_{Γ_N,Γ_N} too large: e.g. $s = 13$, $N = 200$ yields $10^{20} \times 10^{20}$.

Notabene: naive algorithm

Build $F_n := F_{\Gamma_n,\Gamma_n}$, $n = 0, 1, 2, \ldots,$ until $\operatorname{rank} F_n = \operatorname{rank} F_{n+1}$.

Minimal Sampling

Two things needed

1. Interpolation space for *all* sets X with $\#X \leq N$.
2. Efficient polynomial solver.

Choices for the set A

1. Interpolation guaranteed for $A = \Gamma_{N-1}$.
2. Minimal for $s = 1$.
3. $\dim \Pi_{N-1} = \binom{N-1+s}{s} \simeq N^s/s!$.
4. F_{Γ_N,Γ_N} too large: e.g. $s = 13$, $N = 200$ yields $10^{20} \times 10^{20}$.

Notabene: naive algorithm

Build $F_n := F_{\Gamma_n,\Gamma_n}$, $n = 0, 1, 2, \ldots,$ until $\operatorname{rank} F_n = \operatorname{rank} F_{n+1}$.

Minimal Sampling

UNIVERSITÄT PASSAU

Two things needed

1. Interpolation space for *all* sets X with $\#X \le N$.
2. Efficient polynomial solver.

Choices for the set A

1. Interpolation guaranteed for $A = \Gamma_{N-1}$.
2. Minimal for $s = 1$.
3. $\dim \Pi_{N-1} = \binom{N-1+s}{s} \simeq N^s/s!$.
4. $\boldsymbol{F}_{\Gamma_N,\Gamma_N}$ too large: e.g. $s = 13$, $N = 200$ yields $10^{20} \times 10^{20}$.

Notabene: naive algorithm

Build $F_n := F_{\Gamma_n,\Gamma_n}$, $n = 0,1,2,\dots,$ until $\operatorname{rank} F_n = \operatorname{rank} F_{n+1}$.

Minimal Sampling

Two things needed

1. Interpolation space for *all* sets X with $\#X \leq N$.
2. Efficient polynomial solver.

Choices for the set A

1. Interpolation guaranteed for $A = \Gamma_{N-1}$.
2. Minimal for $s = 1$.
3. $\dim \Pi_{N-1} = \binom{N-1+s}{s} \simeq N^s/s!$.
4. $\boldsymbol{F}_{\Gamma_N,\Gamma_N}$ too large: e.g. $s = 13$, $N = 200$ yields $10^{20} \times 10^{20}$.

Notabene: naive algorithm

Build $\boldsymbol{F}_n := \boldsymbol{F}_{\Gamma_n,\Gamma_n}$, $n = 0, 1, 2, \ldots,$ until $\operatorname{rank} \boldsymbol{F}_n = \operatorname{rank} \boldsymbol{F}_{n+1}$.

Minimal Sampling

Two things needed

1. Interpolation space for *all* sets X with $\#X \le N$.
2. Efficient polynomial solver.

Choices for the set A

1. Interpolation guaranteed for $A = \Gamma_{N-1}$.
2. Minimal for $s = 1$.
3. $\dim \Pi_{N-1} = \binom{N-1+s}{s} \simeq N^s/s!$.
4. $\boldsymbol{F}_{\Gamma_N,\Gamma_N}$ too large: e.g. $s = 13$, $N = 200$ yields $10^{20} \times 10^{20}$.

Notabene: naive algorithm does not work

Build $\boldsymbol{F}_n := \boldsymbol{F}_{\Gamma_n,\Gamma_n}$, $n = 0, 1, 2, \ldots$, until $\operatorname{rank} \boldsymbol{F}_n = \operatorname{rank} \boldsymbol{F}_{n+1}$.
Example: $\Omega = \{\omega, \omega'\}, f_\omega = -f_{\omega'}$

Minimal Sampling

Two things needed

1. Interpolation space for *all* sets X with $\#X \le N$.
2. Efficient polynomial solver.

Choices for the set A

1. Interpolation guaranteed for $A = \Gamma_{N-1}$.
2. Minimal for $s = 1$.
3. $\dim \Pi_{N-1} = \binom{N-1+s}{s} \simeq N^s/s!$.
4. $\boldsymbol{F}_{\Gamma_N,\Gamma_N}$ too large: e.g. $s = 13$, $N = 200$ yields $10^{20} \times 10^{20}$.

Notabene: naive algorithm does not work

Build $\boldsymbol{F}_n := \boldsymbol{F}_{\Gamma_n,\Gamma_n}$, $n = 0, 1, 2, \dots$, until $\operatorname{rank} \boldsymbol{F}_n = \operatorname{rank} \boldsymbol{F}_{n+1}$.
Example: $\Omega = \{\omega, \omega'\}, f_\omega = -f_{\omega'} \Rightarrow \boldsymbol{F}_0 = f_\omega e^{\omega^T 0} + f_{\omega'} e^{\omega'^T 0}$

Minimal Sampling

Two things needed

1. Interpolation space for *all* sets X with $\#X \le N$.
2. Efficient polynomial solver.

Choices for the set A

1. Interpolation guaranteed for $A = \Gamma_{N-1}$.
2. Minimal for $s = 1$.
3. $\dim \Pi_{N-1} = \binom{N-1+s}{s} \simeq N^s/s!$.
4. $\boldsymbol{F}_{\Gamma_N,\Gamma_N}$ too large: e.g. $s = 13$, $N = 200$ yields $10^{20} \times 10^{20}$.

Notabene: naive algorithm does not work

Build $\boldsymbol{F}_n := \boldsymbol{F}_{\Gamma_n,\Gamma_n}$, $n = 0, 1, 2, \dots,$ until $\operatorname{rank} \boldsymbol{F}_n = \operatorname{rank} \boldsymbol{F}_{n+1}$.
Example: $\Omega = \{\omega, \omega'\}, f_\omega = -f_{\omega'} \Rightarrow \boldsymbol{F}_0 = f_\omega + f_{\omega'}$

Minimal Sampling

Two things needed

1. Interpolation space for *all* sets X with $\# X \le N$.
2. Efficient polynomial solver.

Choices for the set A

1. Interpolation guaranteed for $A = \Gamma_{N-1}$.
2. Minimal for $s = 1$.
3. $\dim \Pi_{N-1} = \binom{N-1+s}{s} \simeq N^s / s!$.
4. $\boldsymbol{F}_{\Gamma_N, \Gamma_N}$ too large: e.g. $s = 13$, $N = 200$ yields $10^{20} \times 10^{20}$.

Notabene: naive algorithm does not work

Build $\boldsymbol{F}_n := \boldsymbol{F}_{\Gamma_n, \Gamma_n}$, $n = 0, 1, 2, \dots$, until $\operatorname{rank} \boldsymbol{F}_n = \operatorname{rank} \boldsymbol{F}_{n+1}$.
Example: $\Omega = \{\omega, \omega'\}, f_\omega = -f_{\omega'} \Rightarrow \boldsymbol{F}_0 = 0$.

Universal Interpolation

Definition

$\mathscr{P} \subset \Pi$ *universal* of order N if $\mathscr{P}$ is interpolation space space for any $X \subset \mathbb{C}^s$ with $\# \leq N$.

Classical problem: minimal universal space

Given N, what is the least dimensional subspace of Π that allows for interpolation at any $X \subset \mathbb{C}^s$ with $\#X \leq N$?

Prony version, monomial

Given N, what is the smallest set $\Upsilon_N \subset \Gamma$ with:

- for any $X \subset \mathbb{C}^s$, $\#X \leq N$,
- there exists $A \subset \Upsilon_N$, $\#A = \#X$,
- such that A is a *degree reducing interpolation set* for X.

Universal Interpolation

Definition (poor man's Haar space)

$\mathscr{P} \subset \Pi$ *universal* of order N if $\mathscr{P}$ is interpolation space space for any $X \subset \mathbb{C}^s$ with $\# \le N$.

Classical problem: minimal universal space

Given N, what is the least dimensional subspace of Π that allows for interpolation at any $X \subset \mathbb{C}^s$ with $\#X \le N$?

Prony version, monomial

Given N, what is the smallest set $\Upsilon_N \subset \Gamma$ with:

- for any $X \subset \mathbb{C}^s$, $\#X \le N$,
- there exists $A \subset \Upsilon_N$, $\#A = \#X$,
- such that A is a *degree reducing interpolation set* for X.

Universal Interpolation

Definition (poor man's Haar space)

$\mathscr{P} \subset \Pi$ *universal* of order N if $\mathscr{P}$ is interpolation space space for any $X \subset \mathbb{C}^s$ with $\# \le N$.

Classical problem: minimal universal space

Given N, what is the least dimensional subspace of Π that allows for interpolation at any $X \subset \mathbb{C}^s$ with $\#X \le N$?

Prony version, monomial

Given N, what is the smallest set $\Upsilon_N \subset \Gamma$ with:

- for any $X \subset \mathbb{C}^s$, $\#X \le N$,
- there exists $A \subset \Upsilon_N$, $\#A = \#X$,
- such that A is a *degree reducing interpolation set* for X.

Universal Interpolation

UNIVERSITÄT PASSAU

Definition (poor man's Haar space)

$\mathscr{P} \subset \Pi$ *universal* of order N if $\mathscr{P}$ is interpolation space space for any $X \subset \mathbb{C}^s$ with $\# \le N$.

Classical problem: minimal universal space

Given N, what is the least dimensional subspace of Π that allows for interpolation at any $X \subset \mathbb{C}^s$ with $\#X \le N$?

Prony version, monomial

Given N, what is the smallest set $\Upsilon_N \subset \Gamma$ with:

- for any $X \subset \mathbb{C}^s$, $\#X \le N$,
- there exists $A \subset \Upsilon_N$, $\#A = \#X$,
- such that A is a *degree reducing interpolation set* for X.

Universal Interpolation

Definition (poor man's Haar space)

$\mathscr{P} \subset \Pi$ *universal* of order N if $\mathscr{P}$ is interpolation space space for any $X \subset \mathbb{C}^s$ with $\# \le N$.

Classical problem: minimal universal space

Given N, what is the least dimensional subspace of Π that allows for interpolation at any $X \subset \mathbb{C}^s$ with $\#X \le N$?

Prony version, monomial

Given N, what is the smallest set $\Upsilon_N \subset \Gamma$ with:

- for any $X \subset \mathbb{C}^s$, $\#X \le N$,
- there exists $A \subset \Upsilon_N$, $\#A = \#X$,
- such that A is a *degree reducing interpolation set* for X.

Universal Interpolation

Definition (poor man's Haar space)

$\mathscr{P} \subset \Pi$ *universal* of order N if $\mathscr{P}$ is interpolation space space for any $X \subset \mathbb{C}^s$ with $\# \le N$.

Classical problem: minimal universal space

Given N, what is the least dimensional subspace of Π that allows for interpolation at any $X \subset \mathbb{C}^s$ with $\#X \le N$?

Prony version, monomial

Given N, what is the smallest set $\Upsilon_N \subset \Gamma$ with:

- for any $X \subset \mathbb{C}^s$, $\#X \le N$,
- there exists $A \subset \Upsilon_N$, $\#A = \#X$,
- such that A is a *degree reducing interpolation set* for X.

Universal Interpolation

Definition (poor man's Haar space)

$\mathscr{P} \subset \Pi$ *universal* of order N if $\mathscr{P}$ is interpolation space space for any $X \subset \mathbb{C}^s$ with $\# \le N$.

Classical problem: minimal universal space

Given N, what is the least dimensional subspace of Π that allows for interpolation at any $X \subset \mathbb{C}^s$ with $\#X \le N$?

Prony version, monomial

Given N, what is the smallest set $\Upsilon_N \subset \Gamma$ with:

- for any $X \subset \mathbb{C}^s$, $\#X \le N$,
- there exists $A \subset \Upsilon_N$, $\#A = \#X$,
- such that A is a *degree reducing interpolation set* for X.

Universal Interpolation and Lower Sets

Definition

1. $A \subset \Gamma$ is called *lower set* if $\alpha \in A \quad \Rightarrow \quad \{\beta : \beta \le \alpha\} \subseteq A$.
2. $\mathscr{L}_j :=$ all lower sets of cardinality j.

Theorem

1. If Π_Θ is degree reducing monomially universal then
$$\Theta \supseteq \bigcup_{j=1}^{N} \bigcup_{A \in \mathscr{L}_j} B. \tag{1}$$
2. The set on the right hand side of (1) is universal …
3. $\alpha \in \Upsilon$ iff $\prod_{j=1}^{s} (1+\alpha_j) \le N$.

Universal Interpolation and Lower Sets

Definition

1. $A \subset \Gamma$ is called *lower set* if $\alpha \in A \quad \Rightarrow \quad \{\beta : \beta \le \alpha\} \subseteq A$.
2. $\mathscr{L}_j :=$ all lower sets of cardinality j.

Theorem

1. If Π_Θ is degree reducing monomially universal then
$$\Theta \supseteq \bigcup_{j=1}^{N} \bigcup_{A \in \mathscr{L}_j} B. \tag{1}$$
2. The set on the right hand side of (1) is universal …
3. $\alpha \in \Upsilon$ iff $\prod_{j=1}^{s} (1 + \alpha_j) \le N$.

Universal Interpolation and Lower Sets

Definition

1. $A \subset \Gamma$ is called *lower set* if $\alpha \in A \quad \Rightarrow \quad \{\beta : \beta \leq \alpha\} \subseteq A$.
2. $\mathscr{L}_j :=$ all lower sets of cardinality j.

Theorem

1. If Π_Θ is degree reducing monomially universal then
$$\Theta \supseteq \bigcup_{j=1}^{N} \bigcup_{A \in \mathscr{L}_j} B. \tag{1}$$
2. The set on the right hand side of (1) is universal …
3. $\alpha \in \Upsilon$ iff $\prod_{j=1}^{s} (1 + \alpha_j) \leq N$.

Universal Interpolation and Lower Sets

Definition

1. $A \subset \Gamma$ is called *lower set* if $\alpha \in A \quad \Rightarrow \quad \{\beta : \beta \leq \alpha\} \subseteq A$.
2. $\mathscr{L}_j :=$ all lower sets of cardinality j.

Theorem

1. If Π_Θ is degree reducing monomially universal then
$$\Theta \supseteq \bigcup_{j=1}^{N} \bigcup_{A \in \mathscr{L}_j} B. \tag{1}$$
2. The set on the right hand side of (1) is universal …
3. $\alpha \in \Upsilon$ iff $\prod_{j=1}^{s} (1 + \alpha_j) \leq N$.

Universal Interpolation and Lower Sets

UNIVERSITÄT PASSAU

Definition

1. $A \subset \Gamma$ is called *lower set* if $\alpha \in A \quad \Rightarrow \quad \{\beta : \beta \le \alpha\} \subseteq A$.
2. $\mathscr{L}_j :=$ all lower sets of cardinality j.

Theorem (Tools: H–bases and ideals of tensor product grids)

1. If Π_Θ is degree reducing monomially universal then

$$\Theta \supseteq \bigcup_{j=1}^{N} \bigcup_{A \in \mathscr{L}_j} B. \tag{1}$$

2. The set on the right hand side of (1) is universal ...
3. $\alpha \in \Upsilon$ iff $\prod_{j=1}^{s} (1+\alpha_j) \le N$.

Universal Interpolation and Lower Sets

Definition

1. $A \subset \Gamma$ is called *lower set* if $\alpha \in A \quad \Rightarrow \quad \{\beta : \beta \leq \alpha\} \subseteq A$.
2. $\mathscr{L}_j :=$ all lower sets of cardinality j.

Theorem (Tools: H–bases and ideals of tensor product grids)

1. If Π_Θ is degree reducing monomially universal then

$$\Theta \supseteq \bigcup_{j=1}^{N} \bigcup_{A \in \mathscr{L}_j} B. \tag{1}$$

2. The set on the right hand side of (1) is universal …
3. $\alpha \in \Upsilon$ iff $\prod_{j=1}^{s} (1 + \alpha_j) \leq N$.

Universal Interpolation and Lower Sets

Definition

1. $A \subset \Gamma$ is called *lower set* if $\alpha \in A \quad \Rightarrow \quad \{\beta : \beta \le \alpha\} \subseteq A$.
2. $\mathscr{L}_j :=$ all lower sets of cardinality j.

Theorem (Tools: H–bases and ideals of tensor product grids)

1. If Π_Θ is degree reducing monomially universal then
$$\Theta \supseteq \bigcup_{j=1}^{N} \bigcup_{A \in \mathscr{L}_j} B =: \Upsilon. \tag{1}$$
2. The set on the right hand side of (1) is universal …
3. $\alpha \in \Upsilon$ iff $\prod_{j=1}^{s} (1 + \alpha_j) \le N$.

Universal Interpolation and Lower Sets

Definition

1. $A \subset \Gamma$ is called *lower set* if $\alpha \in A \quad \Rightarrow \quad \{\beta : \beta \le \alpha\} \subseteq A$.
2. $\mathscr{L}_j :=$ all lower sets of cardinality j.

Theorem (Tools: H–bases and ideals of tensor product grids)

1. If Π_Θ is degree reducing monomially universal then

$$\Theta \supseteq \bigcup_{j=1}^{N} \bigcup_{A \in \mathscr{L}_j} B =: \Upsilon. \tag{1}$$

2. The set Υ is universal ...
3. $\alpha \in \Upsilon$ iff $\prod_{j=1}^{s} (1+\alpha_j) \le N$.

Universal Interpolation and Lower Sets

Definition

1. $A \subset \Gamma$ is called *lower set* if $\alpha \in A \quad \Rightarrow \quad \{\beta : \beta \le \alpha\} \subseteq A$.
2. $\mathscr{L}_j :=$ all lower sets of cardinality j.

Theorem (Tools: H–bases and ideals of tensor product grids)

1. If Π_Θ is degree reducing monomially universal then
$$\Theta \supseteq \bigcup_{j=1}^{N} \bigcup_{A \in \mathscr{L}_j} B =: \Upsilon. \qquad (1)$$
2. The set Υ is minimally universal …
3. $\alpha \in \Upsilon$ iff $\prod_{j=1}^{s} (1+\alpha_j) \le N$.

Universal Interpolation and Lower Sets

Definition

1. $A \subset \Gamma$ is called *lower set* if $\alpha \in A \quad \Rightarrow \quad \{\beta : \beta \le \alpha\} \subseteq A$.
2. $\mathscr{L}_j :=$ all lower sets of cardinality j.

Theorem (Tools: H–bases and ideals of tensor product grids)

1. If Π_Θ is degree reducing monomially universal then
$$\Theta \supseteq \bigcup_{j=1}^{N} \bigcup_{A \in \mathscr{L}_j} B =: \Upsilon. \tag{1}$$
2. The set Υ is minimally universal …
3. $\alpha \in \Upsilon$ iff $\prod_{j=1}^{s} (1 + \alpha_j) \le N$.

Universal Interpolation and Lower Sets

Definition

1. $A \subset \Gamma$ is called *lower set* if $\alpha \in A \quad \Rightarrow \quad \{\beta : \beta \leq \alpha\} \subseteq A$.
2. $\mathscr{L}_j :=$ all lower sets of cardinality j.

Theorem (Tools: H–bases and ideals of tensor product grids)

1. If Π_Θ is degree reducing monomially universal then
$$\Theta \supseteq \bigcup_{j=1}^{N} \bigcup_{A \in \mathscr{L}_j} B =: \Upsilon. \tag{1}$$
2. The set Υ is minimally universal ...
3. $\alpha \in \Upsilon$ iff $\prod_{j=1}^{s} (1 + \alpha_j) \leq N$. Positive part of *hyperbolic cross*.

Summary & Algorithm

Theorem

The hyperbolic cross $\Upsilon_N \subset \mathbb{N}_0^s$ of order N

1. is the unique minimal universal monomial degree reducing interpolation space.
2. has cardinality $\le N \log^{s-1} N$.
3. satisfies $\Upsilon_N \subset \Gamma_N \subset \{\|\alpha\|_\infty \le N\}$.

Algorithm (Prony ideal & interpolation space)

For increasing sets $\{0\} = A_0 \subset A_1 \subset \cdots \subset \mathbb{N}_0^s$

1. build $F_j = F_{\Upsilon_N, A_j} = \left[F_{\Upsilon_N, A_{j-1}} \mid * \right]$,
2. from $\ker F_j$
3. from $(\ker F_j)^c$

Summary & Algorithm

UNIVERSITÄT PASSAU

Theorem

The hyperbolic cross $\Upsilon_N \subset \mathbb{N}_0^s$ of order N

1. is the unique minimal universal monomial degree reducing interpolation space.
2. has cardinality $\le N \log^{s-1} N$.
3. satisfies $\Upsilon_N \subset \Gamma_N \subset \{\|\alpha\|_\infty \le N\}$.

Algorithm (Prony ideal & interpolation space)

For increasing sets $\{0\} = A_0 \subset A_1 \subset \cdots \subset \mathbb{N}_0^s$

4. build $F_j = F_{\Upsilon_N, A_j} = \left[F_{\Upsilon_N, A_{j-1}} \,|\, * \right]$,
5. from $\ker F_j$
6. from $(\ker F_j)^c$

Summary & Algorithm

Theorem

The hyperbolic cross $\Upsilon_N \subset \mathbb{N}_0^s$ of order N

1. is the unique minimal universal monomial degree reducing interpolation space.
2. has cardinality $\leq N \log^{s-1} N$.
3. satisfies $\Upsilon_N \subset \Gamma_N \subset \{\|\alpha\|_\infty \leq N\}$.

Algorithm (Prony ideal & interpolation space)

For increasing sets $\{0\} = A_0 \subset A_1 \subset \cdots \subset \mathbb{N}_0^s$

1. build $F_j = F_{\Upsilon_N, A_j} = \left[F_{\Upsilon_N, A_{j-1}} \mid * \right]$,
2. from $\ker F_j$
3. from $(\ker F_j)^c$

Summary & Algorithm

Theorem

The hyperbolic cross $\Upsilon_N \subset \mathbb{N}_0^s$ of order N

1. is the unique minimal universal monomial degree reducing interpolation space.
2. has cardinality $\le N \log^{s-1} N$.
3. satisfies $\Upsilon_N \subset \Gamma_N \subset \{\|\alpha\|_\infty \le N\}$.

Algorithm (Prony ideal & interpolation space)

For increasing sets $\{0\} = A_0 \subset A_1 \subset \cdots \subset \mathbb{N}_0^s$

1. build $F_j = F_{\Upsilon_N, A_j} = \left[F_{\Upsilon_N, A_{j-1}} \,|\, * \right]$,
2. from $\ker F_j$
3. from $(\ker F_j)^c$

Summary & Algorithm

UNIVERSITÄT PASSAU

Theorem

The hyperbolic cross $\Upsilon_N \subset \mathbb{N}_0^s$ of order N

1. is the unique minimal universal monomial degree reducing interpolation space.
2. has cardinality $\leq N \log^{s-1} N$.
3. satisfies $\Upsilon_N \subset \Gamma_N \subset \{\|\alpha\|_\infty \leq N\}$.

- S. Kunis, Th. Peter, T. Römer, and U. von der Ohe, *A multivariate generalization of Prony's method*, Linear Algebra Appl. **490** (2016), 31–47.
- T. Sauer, *Prony's method in several variables*, Numer. Math., to appear

Algorithm (Prony ideal & interpolation space)

For increasing sets $\{0\} = A_0 \subset A_1 \subset \cdots \subset \mathbb{N}_0^s$

1. build $F_j = F_{\Upsilon_N, A_j} = \left[F_{\Upsilon_N, A_{j-1}} \mid * \right]$,

Summary & Algorithm

Theorem

The hyperbolic cross $\Upsilon_N \subset \mathbb{N}_0^s$ of order N

1. is the unique minimal universal monomial degree reducing interpolation space.
2. has cardinality $\le N \log^{s-1} N$.
3. satisfies $\Upsilon_N \subset \Gamma_N \subset \{\|\alpha\|_\infty \le N\}$.

Algorithm (Prony ideal & interpolation space)

For increasing sets $\{0\} = A_0 \subset A_1 \subset \cdots \subset \mathbb{N}_0^s$

1. build $F_j = F_{\Upsilon_N, A_j} = \left[F_{\Upsilon_N, A_{j-1}} \,|\, * \right]$,
2. from $\ker F_j$
3. from $(\ker F_j)^c$

Summary & Algorithm

Theorem

The hyperbolic cross $\Upsilon_N \subset \mathbb{N}_0^s$ of order N

1. is the unique minimal universal monomial degree reducing interpolation space.
2. has cardinality $\leq N \log^{s-1} N$.
3. satisfies $\Upsilon_N \subset \Gamma_N \subset \{\|\alpha\|_\infty \leq N\}$.

Algorithm (Prony ideal & interpolation space)

For increasing sets $\{0\} = A_0 \subset A_1 \subset \cdots \subset \mathbb{N}_0^s$

1. build $\boldsymbol{F}_j = \boldsymbol{F}_{\Upsilon_N, A_j} = \left[\boldsymbol{F}_{\Upsilon_N, A_{j-1}} \,|\, * \right]$,
2. from $\ker \boldsymbol{F}_j$
3. from $(\ker \boldsymbol{F}_j)^c$

Summary & Algorithm

UNIVERSITÄT PASSAU

Theorem

The hyperbolic cross $\Upsilon_N \subset \mathbb{N}_0^s$ of order N

1. is the unique minimal universal monomial degree reducing interpolation space.
2. has cardinality $\le N \log^{s-1} N$.
3. satisfies $\Upsilon_N \subset \Gamma_N \subset \{\|\alpha\|_\infty \le N\}$.

Algorithm (Prony ideal & interpolation space)

For increasing sets $\{0\} = A_0 \subset A_1 \subset \cdots \subset \mathbb{N}_0^s$

1. build $\boldsymbol{F}_j = \boldsymbol{F}_{\Upsilon_N, A_j} = \left[\boldsymbol{F}_{\Upsilon_N, A_{j-1}} \mid * \right]$,
2. from $\ker \boldsymbol{F}_j$
3. from $(\ker \boldsymbol{F}_j)^c$

Summary & Algorithm

UNIVERSITÄT PASSAU

Theorem

The hyperbolic cross $\Upsilon_N \subset \mathbb{N}_0^s$ of order N

1. is the unique minimal universal monomial degree reducing interpolation space.
2. has cardinality $\le N \log^{s-1} N$.
3. satisfies $\Upsilon_N \subset \Gamma_N \subset \{\|\alpha\|_\infty \le N\}$.

Algorithm (Prony ideal & interpolation space)

For increasing sets $\{0\} = A_0 \subset A_1 \subset \cdots \subset \mathbb{N}_0^s$

1. build $\boldsymbol{F}_j = \boldsymbol{F}_{\Upsilon_N, A_j} = \left[\boldsymbol{F}_{\Upsilon_N, A_{j-1}} \,|\, * \right]$,
2. from $\ker \boldsymbol{F}_j$ extend ideal basis to $I_\Omega \cap \Pi_{A_j}$,
3. from $(\ker \boldsymbol{F}_j)^c$

Summary & Algorithm

UNIVERSITÄT PASSAU

Theorem

The hyperbolic cross $\Upsilon_N \subset \mathbb{N}_0^s$ of order N

1. is the unique minimal universal monomial degree reducing interpolation space.
2. has cardinality $\le N \log^{s-1} N$.
3. satisfies $\Upsilon_N \subset \Gamma_N \subset \{\|\alpha\|_\infty \le N\}$.

Algorithm (Prony ideal & interpolation space)

For increasing sets $\{0\} = A_0 \subset A_1 \subset \cdots \subset \mathbb{N}_0^s$

1. build $\boldsymbol{F}_j = \boldsymbol{F}_{\Upsilon_N, A_j} = \left[\boldsymbol{F}_{\Upsilon_N, A_{j-1}} \mid * \right]$,
2. from $\ker \boldsymbol{F}_j$ extend ideal basis to $I_\Omega \cap \Pi_{A_j}$,
3. from $(\ker \boldsymbol{F}_j)^c$ extend interpolation space to Π_{A_j},

Summary & Algorithm

UNIVERSITÄT PASSAU

Theorem

The hyperbolic cross $\Upsilon_N \subset \mathbb{N}_0^s$ of order N

1. is the unique minimal universal monomial degree reducing interpolation space.
2. has cardinality $\le N \log^{s-1} N$.
3. satisfies $\Upsilon_N \subset \Gamma_N \subset \{\|\alpha\|_\infty \le N\}$.

Algorithm (Prony ideal & interpolation space)

For increasing sets $\{0\} = A_0 \subset A_1 \subset \cdots \subset \mathbb{N}_0^s$

1. build $\boldsymbol{F}_j = \boldsymbol{F}_{\Upsilon_N, A_j} = \left[\boldsymbol{F}_{\Upsilon_N, A_{j-1}} \mid * \right]$,
2. from $\ker \boldsymbol{F}_j$ extend ideal basis to $I_\Omega \cap \Pi_{A_j}$,
3. from $(\ker \boldsymbol{F}_j)^c$ extend interpolation space to Π_{A_j},

until $\operatorname{rank} \boldsymbol{F}_j = \operatorname{rank} \boldsymbol{F}_{j-1}$.

Summary & Algorithm

Theorem

The hyperbolic cross $\Upsilon_N \subset \mathbb{N}_0^s$ of order N

1. is the unique minimal universal monomial degree reducing interpolation space.
2. has cardinality $\le N \log^{s-1} N$.
3. satisfies $\Upsilon_N \subset \Gamma_N \subset \{\|\alpha\|_\infty \le N\}$.

Algorithm (Prony ideal & interpolation space)

For increasing sets $\{0\} = A_0 \subset A_1 \subset \cdots \subset \mathbb{N}_0^s$ filling $\Gamma_0, \Gamma_1, \ldots$

1. build $\boldsymbol{F}_j = \boldsymbol{F}_{\Upsilon_N, A_j} = \left[\boldsymbol{F}_{\Upsilon_N, A_{j-1}} \,|\, * \right]$,
2. from $\ker \boldsymbol{F}_j$ extend ideal basis to $I_\Omega \cap \Pi_{A_j}$,
3. from $(\ker \boldsymbol{F}_j)^c$ extend interpolation space to Π_{A_j},

until $\operatorname{rank} \boldsymbol{F}_j = \operatorname{rank} \boldsymbol{F}_{j-1}$.

Numerically or Symbolically?

1. Symbolic:

   Sparse Monomial Interpolation with Least Elements .

2. Symbolic/Numeric:

   Sparse Homogeneous Ideal Techniques

Theorem (Small sample sets)

SMILE computes Gröbner basis and interpolation space from at most $s\,N^2 \log^{s-1} N$ samples of f on Γ.

Remarks

1. N^2 cannot be improved.
2. Nonlinear equations are generated in "good" form:

Numerically or Symbolically?

UNIVERSITÄT PASSAU

1. Symbolic:

   Sparse Monomial Interpolation with Least Elements .

2. Symbolic/Numeric:

   Sparse Homogeneous Ideal Techniques

Theorem (Small sample sets)

SMILE computes Gröbner basis and interpolation space from at most $s\,N^2 \log^{s-1} N$ samples of f on Γ.

Remarks

1. N^2 cannot be improved.
2. Nonlinear equations are generated in "good" form:

Numerically or Symbolically?

1. Symbolic:

   Sparse Monomial Interpolation with Least Elements .

2. Symbolic/Numeric:

   Sparse Homogeneous Ideal Techniques

Theorem (Small sample sets)

SMILE computes Gröbner basis and interpolation space from at most $s\,N^2 \log^{s-1} N$ samples of f on Γ.

Remarks

1. N^2 cannot be improved.
2. Nonlinear equations are generated in "good" form:

Numerically or Symbolically?

1. Symbolic:

   Sparse Monomial Interpolation with Least Elements (*SMILE*).

2. Symbolic/Numeric:

   Sparse Homogeneous Ideal Techniques

Theorem (Small sample sets)

SMILE computes Gröbner basis and interpolation space from at most $s\,N^2 \log^{s-1} N$ samples of f on Γ.

Remarks

1. N^2 cannot be improved.
2. Nonlinear equations are generated in "good" form:

Numerically or Symbolically?

1. Symbolic:

   Sparse Monomial Interpolation with Least Elements (SMILE).

2. Symbolic/Numeric:

   Sparse Homogeneous Ideal Techniques

Theorem (Small sample sets)

SMILE computes Gröbner basis and interpolation space from at most $s\,N^2 \log^{s-1} N$ samples of f on Γ.

Remarks

1. N^2 cannot be improved.
2. Nonlinear equations are generated in "good" form:

Numerically or Symbolically?

UNIVERSITÄT PASSAU

1. Symbolic:

   Sparse Monomial Interpolation with Least Elements (*SMILE*).

2. Symbolic/Numeric:

   Sparse Homogeneous Ideal Techniques

Theorem (Small sample sets)

SMILE computes Gröbner basis and interpolation space from at most $s\,N^2 \log^{s-1} N$ samples of f on Γ.

Remarks

1. N^2 cannot be improved.
2. Nonlinear equations are generated in “good” form:

Numerically or Symbolically?

1. Symbolic:

   Sparse Monomial Interpolation with Least Elements (*SMILE*).

2. Symbolic/Numeric:

   Sparse Homogeneous Ideal Techniques

Theorem (Small sample sets)

SMILE computes Gröbner basis and interpolation space from at most $s\,N^2 \log^{s-1} N$ samples of f on Γ.

Remarks

1. N^2 cannot be improved.
2. Nonlinear equations are generated in "good" form:

Numerically or Symbolically?

1. Symbolic:

   Sparse Monomial Interpolation with Least Elements (SMILE).

2. Symbolic/Numeric:

   Sparse Homogeneous Ideal Techniques

Theorem (Small sample sets)

SMILE computes Gröbner basis and interpolation space from at most $s\,N^2 \log^{s-1} N$ samples of f on Γ.

Remarks

1. N^2 cannot be improved.
2. Nonlinear equations are generated in "good" form:

Numerically or Symbolically?

1. Symbolic:

   Sparse Monomial Interpolation with Least Elements (*SMILE*).

2. Symbolic/Numeric:

   Sparse Homogeneous Ideal Techniques

Theorem (Small sample sets)

SMILE computes Gröbner basis and interpolation space from at most $s\,N^2 \log^{s-1} N$ samples of f on Γ.

Remarks

1. N^2 cannot be improved.
2. Nonlinear equations are generated in "good" form:

   Gröbner/H–basis + graded interpolation basis.

Augmenting Matrices

Rank detection

1. Compute SVD of *block augmented* $\boldsymbol{F}_j = [\boldsymbol{F}_{j-1} \,|\, \boldsymbol{A}_j]$.
2. $\tau \geq \sigma_{k+1} \geq \cdots \geq \sigma_n$ yields

$$\|\boldsymbol{F}_j \boldsymbol{x}_j\|_2 \leq \tau \|\boldsymbol{x}\|_2, \qquad \boldsymbol{x} \in \operatorname{span}\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{n-k}\}.$$

3. $\boldsymbol{F}_j = \boldsymbol{U}\Sigma\boldsymbol{V}$: $\boldsymbol{V}$ gives bases for $\ker \boldsymbol{F}_j$ and $(\ker \boldsymbol{F}_j)^c$.

SVD update [with J. M. Peña]

1. Use SVD of $\boldsymbol{F}_{j-1}$ for SVD of $\boldsymbol{F}_j$.
2. Strategy for given threshold τ.
3. Uses QR and SVD of smaller matrices.
4. Guarantees $\operatorname{rank} \boldsymbol{F}_j \geq \operatorname{rank} \boldsymbol{F}_{j-1}$.

Augmenting Matrices

Rank detection

1. Compute SVD of *block augmented* $\boldsymbol{F}_j = [\boldsymbol{F}_{j-1} \,|\, \boldsymbol{A}_j]$.
2. $\tau \geq \sigma_{k+1} \geq \cdots \geq \sigma_n$ yields

$$\|\boldsymbol{F}_j \boldsymbol{x}_j\|_2 \leq \tau \|\boldsymbol{x}\|_2, \qquad \boldsymbol{x} \in \operatorname{span}\{v_1, \dots, v_{n-k}\}.$$

3. $\boldsymbol{F}_j = \boldsymbol{U}\Sigma\boldsymbol{V}$: $\boldsymbol{V}$ gives bases for $\ker \boldsymbol{F}_j$ and $(\ker \boldsymbol{F}_j)^c$.

SVD update [with J. M. Peña]

1. Use SVD of F_{j-1} for SVD of F_j.
2. Strategy for given threshold τ.
3. Uses QR and SVD of smaller matrices.
4. Guarantees rank $F_j \geq$ rank F_{j-1}.

Augmenting Matrices

UNIVERSITÄT PASSAU

Rank detection

1. Compute SVD of *block augmented* $\boldsymbol{F}_j = [\boldsymbol{F}_{j-1} \,|\, \boldsymbol{A}_j]$.
2. $\tau \geq \sigma_{k+1} \geq \cdots \geq \sigma_n$ yields

$$\|\boldsymbol{F}_j \boldsymbol{x}_j\|_2 \leq \tau \|\boldsymbol{x}\|_2, \qquad \boldsymbol{x} \in \operatorname{span}\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{n-k}\}.$$

3. $\boldsymbol{F}_j = \boldsymbol{U}\Sigma\boldsymbol{V}$: $\boldsymbol{V}$ gives bases for $\ker \boldsymbol{F}_j$ and $(\ker \boldsymbol{F}_j)^c$.

SVD update [with J. M. Peña]

1. Use SVD of $\boldsymbol{F}_{j-1}$ for SVD of $\boldsymbol{F}_j$.
2. Strategy for given threshold τ.
3. Uses QR and SVD of smaller matrices.
4. Guarantees $\operatorname{rank} \boldsymbol{F}_j \geq \operatorname{rank} \boldsymbol{F}_{j-1}$.

Augmenting Matrices

UNIVERSITÄT PASSAU

Rank detection

1. Compute SVD of *block augmented* $\boldsymbol{F}_j = [\boldsymbol{F}_{j-1} \,|\, \boldsymbol{A}_j]$.
2. $\tau \geq \sigma_{k+1} \geq \cdots \geq \sigma_n$ yields

$$\|\boldsymbol{F}_j \boldsymbol{x}_j\|_2 \leq \tau \|\boldsymbol{x}\|_2, \qquad \boldsymbol{x} \in \operatorname{span}\{v_1, \dots, v_{n-k}\}.$$

3. $\boldsymbol{F}_j = \boldsymbol{U}\Sigma\boldsymbol{V}$: $\boldsymbol{V}$ gives bases for $\ker \boldsymbol{F}_j$ and $(\ker \boldsymbol{F}_j)^c$.

SVD update [with J. M. Peña]

1. Use SVD of F_{j-1} for SVD of F_j.
2. Strategy for given threshold τ.
3. Uses QR and SVD of smaller matrices.
4. Guarantees $\operatorname{rank} F_j \geq \operatorname{rank} F_{j-1}$.

Augmenting Matrices

Rank detection

1. Compute SVD of *block augmented* $\boldsymbol{F}_j = [\boldsymbol{F}_{j-1} \,|\, \boldsymbol{A}_j]$.
2. $\tau \ge \sigma_{k+1} \ge \cdots \ge \sigma_n$ yields
$$\|\boldsymbol{F}_j \boldsymbol{x}_j\|_2 \le \tau \|\boldsymbol{x}\|_2, \qquad \boldsymbol{x} \in \operatorname{span}\{v_1, \dots, v_{n-k}\}.$$
3. $\boldsymbol{F}_j = \boldsymbol{U}\Sigma\boldsymbol{V}$: $\boldsymbol{V}$ gives bases for $\ker \boldsymbol{F}_j$ and $(\ker \boldsymbol{F}_j)^c$.

SVD update [with J. M. Peña]

1. Use SVD of $\boldsymbol{F}_{j-1}$ for SVD of $\boldsymbol{F}_j$.
2. Strategy for given threshold τ.
3. Uses QR and SVD of smaller matrices.
4. Guarantees $\operatorname{rank} \boldsymbol{F}_j \ge \operatorname{rank} \boldsymbol{F}_{j-1}$.

Augmenting Matrices

Rank detection

1. Compute SVD of *block augmented* $\boldsymbol{F}_j = [\boldsymbol{F}_{j-1} \,|\, \boldsymbol{A}_j]$.
2. $\tau \ge \sigma_{k+1} \ge \cdots \ge \sigma_n$ yields

$$\|\boldsymbol{F}_j \boldsymbol{x}_j\|_2 \le \tau \|\boldsymbol{x}\|_2, \qquad \boldsymbol{x} \in \operatorname{span}\{\boldsymbol{v}_1, \dots, \boldsymbol{v}_{n-k}\}.$$

3. $\boldsymbol{F}_j = \boldsymbol{U}\Sigma\boldsymbol{V}$: $\boldsymbol{V}$ gives bases for $\ker \boldsymbol{F}_j$ and $(\ker \boldsymbol{F}_j)^c$.

SVD update [with J. M. Peña]

1. Use SVD of $\boldsymbol{F}_{j-1}$ for SVD of $\boldsymbol{F}_j$.
2. Strategy for given threshold τ.
3. Uses QR and SVD of smaller matrices.
4. Guarantees $\operatorname{rank} \boldsymbol{F}_j \ge \operatorname{rank} \boldsymbol{F}_{j-1}$.

Augmenting Matrices

Rank detection

1. Compute SVD of *block augmented* $\boldsymbol{F}_j = [\boldsymbol{F}_{j-1} \,|\, \boldsymbol{A}_j]$.
2. $\tau \ge \sigma_{k+1} \ge \cdots \ge \sigma_n$ yields

$$\|\boldsymbol{F}_j \boldsymbol{x}_j\|_2 \le \tau \|\boldsymbol{x}\|_2, \qquad \boldsymbol{x} \in \operatorname{span}\{\boldsymbol{v}_1, \dots, \boldsymbol{v}_{n-k}\}.$$

3. $\boldsymbol{F}_j = \boldsymbol{U}\Sigma\boldsymbol{V}$: $\boldsymbol{V}$ gives bases for $\ker \boldsymbol{F}_j$ and $(\ker \boldsymbol{F}_j)^c$.

SVD update [with J. M. Peña]

1. Use SVD of $\boldsymbol{F}_{j-1}$ for SVD of $\boldsymbol{F}_j$.
2. Strategy for given threshold τ.
3. Uses QR and SVD of smaller matrices.
4. Guarantees $\operatorname{rank} \boldsymbol{F}_j \ge \operatorname{rank} \boldsymbol{F}_{j-1}$.

Augmenting Matrices

Rank detection

1. Compute SVD of *block augmented* $\boldsymbol{F}_j = [\boldsymbol{F}_{j-1} \,|\, \boldsymbol{A}_j]$.
2. $\tau \geq \sigma_{k+1} \geq \cdots \geq \sigma_n$ yields

$$\|\boldsymbol{F}_j \boldsymbol{x}_j\|_2 \leq \tau \|\boldsymbol{x}\|_2, \qquad \boldsymbol{x} \in \operatorname{span}\{v_1, \ldots, v_{n-k}\}.$$

3. $\boldsymbol{F}_j = \boldsymbol{U}\Sigma\boldsymbol{V}$: $\boldsymbol{V}$ gives bases for $\ker \boldsymbol{F}_j$ and $(\ker \boldsymbol{F}_j)^c$.

SVD update [with J. M. Peña]

1. Use SVD of $\boldsymbol{F}_{j-1}$ for SVD of $\boldsymbol{F}_j$.
2. Strategy for given threshold τ.
3. Uses QR and SVD of smaller matrices.
4. Guarantees $\operatorname{rank} \boldsymbol{F}_j \geq \operatorname{rank} \boldsymbol{F}_{j-1}$.

Augmenting Matrices

Rank detection

1. Compute SVD of *block augmented* $\boldsymbol{F}_j = [\boldsymbol{F}_{j-1} \,|\, \boldsymbol{A}_j]$.
2. $\tau \ge \sigma_{k+1} \ge \cdots \ge \sigma_n$ yields

$$\|\boldsymbol{F}_j \boldsymbol{x}_j\|_2 \le \tau \|\boldsymbol{x}\|_2, \qquad \boldsymbol{x} \in \operatorname{span}\{\boldsymbol{v}_1, \dots, \boldsymbol{v}_{n-k}\}.$$

3. $\boldsymbol{F}_j = \boldsymbol{U}\Sigma\boldsymbol{V}$: $\boldsymbol{V}$ gives bases for $\ker \boldsymbol{F}_j$ and $(\ker \boldsymbol{F}_j)^c$.

SVD update [with J. M. Peña]

1. Use SVD of $\boldsymbol{F}_{j-1}$ for SVD of $\boldsymbol{F}_j$.
2. Strategy for given threshold τ.
3. Uses QR and SVD of smaller matrices.
4. Guarantees $\operatorname{rank} \boldsymbol{F}_j \ge \operatorname{rank} \boldsymbol{F}_{j-1}$.

Augmenting Matrices

UNIVERSITÄT PASSAU

Rank detection

1. Compute SVD of *block augmented* $\boldsymbol{F}_j = [\boldsymbol{F}_{j-1} \,|\, \boldsymbol{A}_j]$.
2. $\tau \geq \sigma_{k+1} \geq \cdots \geq \sigma_n$ yields

$$\|\boldsymbol{F}_j \boldsymbol{x}_j\|_2 \leq \tau \|\boldsymbol{x}\|_2, \qquad \boldsymbol{x} \in \operatorname{span}\{\boldsymbol{v}_1, \dots, \boldsymbol{v}_{n-k}\}.$$

3. $\boldsymbol{F}_j = \boldsymbol{U}\Sigma\boldsymbol{V}$: $\boldsymbol{V}$ gives bases for $\ker \boldsymbol{F}_j$ and $(\ker \boldsymbol{F}_j)^c$.

SVD update [with J. M. Peña]

1. Use SVD of $\boldsymbol{F}_{j-1}$ for SVD of $\boldsymbol{F}_j$.
2. Strategy for given threshold τ.
3. Uses QR and SVD of smaller matrices.
4. Guarantees $\operatorname{rank} \boldsymbol{F}_j \geq \operatorname{rank} \boldsymbol{F}_{j-1}$ after thresholding.

Completing the Job: Companion Matrices

Multiplication tables for ideal projectors

1. Multiplication: $\mathscr{P} \ni p$
2. Linear operation on $\mathscr{P} \to$ matrix M_j for a basis P of $\mathscr{P}$.
3. Implicitly computable: *reduction*.
4. Monomial basis: *Frobenius companion matrix*.

Theorem (Stetter, Sticklberger, ...)

The *eigenvalues* of the M_j are $(x_\omega)_j$ and the *eigenvectors* ℓ_ω, $\omega \in \Omega$.

Completing the Job: Companion Matrices

Multiplication tables for ideal projectors

1. Multiplication: $\mathscr{P} \ni p$
2. Linear operation on $\mathscr{P} \to$ matrix M_j for a basis P of $\mathscr{P}$.
3. Implicitly computable: *reduction*.
4. Monomial basis: *Frobenius companion matrix*.

Theorem (Stetter, Sticklberger, …)

The *eigenvalues* of the M_j are $(x_\omega)_j$ and the *eigenvectors* ℓ_ω, $\omega \in \Omega$.

Completing the Job: Companion Matrices

UNIVERSITÄT PASSAU

Multiplication tables for ideal projectors

1. Multiplication: $\mathscr{P} \ni p \mapsto \quad (\cdot)_j p \quad \in \Pi$
2. Linear operation on $\mathscr{P} \to$ matrix M_j for a basis P of $\mathscr{P}$.
3. Implicitly computable: *reduction*.
4. Monomial basis: *Frobenius companion matrix*.

Theorem (Stetter, Sticklberger, ...)

The *eigenvalues* of the M_j are $(x_\omega)_j$ and the *eigenvectors* ℓ_ω, $\omega \in \Omega$.

Completing the Job: Companion Matrices

Multiplication tables for ideal projectors

1. Multiplication modulo ideal: $\mathscr{P} \ni p \mapsto L_{\mathscr{P}}\left((\cdot)_j p\right) \in \mathscr{P}$.
2. Linear operation on $\mathscr{P} \to$ matrix M_j for a basis P of $\mathscr{P}$.
3. Implicitly computable: *reduction*.
4. Monomial basis: *Frobenius companion matrix*.

Theorem (Stetter, Sticklberger, ...)

The *eigenvalues* of the M_j are $(x_\omega)_j$ and the *eigenvectors* ℓ_ω, $\omega \in \Omega$.

Completing the Job: Companion Matrices

Multiplication tables for ideal projectors

1. Multiplication modulo ideal: $\mathscr{P} \ni p \mapsto L_{\mathscr{P}}\left((\cdot)_j p\right) \in \mathscr{P}$.
2. Linear operation on $\mathscr{P} \to$ matrix $\boldsymbol{M}_j$ for a basis P of $\mathscr{P}$.
3. Implicitly computable: *reduction*.
4. Monomial basis: *Frobenius companion matrix*.

Theorem (Stetter, Sticklberger, ...)

The *eigenvalues* of the $\boldsymbol{M}_j$ are $(x_\omega)_j$ and the *eigenvectors* ℓ_ω, $\omega \in \Omega$.

Completing the Job: Companion Matrices

UNIVERSITÄT PASSAU

Multiplication tables for ideal projectors

1. Multiplication modulo ideal: $\mathscr{P} \ni p \mapsto L_{\mathscr{P}} \left((\cdot)_j p \right) \in \mathscr{P}$.
2. Linear operation on $\mathscr{P} \to$ matrix $\boldsymbol{M}_j$ for a basis P of $\mathscr{P}$.
3. Implicitly computable: *reduction*.
4. Monomial basis: *Frobenius companion matrix*.

Theorem (Stetter, Sticklberger, ...)

The *eigenvalues* of the $\boldsymbol{M}_j$ are $(x_\omega)_j$ and the *eigenvectors* ℓ_ω, $\omega \in \Omega$.

Completing the Job: Companion Matrices

UNIVERSITÄT PASSAU

Multiplication tables for ideal projectors

1. Multiplication modulo ideal: $\mathscr{P} \ni p \mapsto L_{\mathscr{P}}\left((\cdot)_j p \right) \in \mathscr{P}$.
2. Linear operation on $\mathscr{P} \to$ matrix $\boldsymbol{M}_j$ for a basis P of $\mathscr{P}$.
3. Implicitly computable: *reduction*. Relies on good decomposition.
4. Monomial basis: *Frobenius companion matrix*.

Theorem (Stetter, Sticklberger, ...)

The *eigenvalues* of the $\boldsymbol{M}_j$ are $(x_\omega)_j$ and the *eigenvectors* ℓ_ω, $\omega \in \Omega$.

Completing the Job: Companion Matrices

Multiplication tables for ideal projectors

1. Multiplication modulo ideal: $\mathscr{P} \ni p \mapsto L_{\mathscr{P}}\left((\cdot)_j p \right) \in \mathscr{P}$.
2. Linear operation on $\mathscr{P} \to$ matrix $\boldsymbol{M}_j$ for a basis P of $\mathscr{P}$.
3. Implicitly computable: *reduction*. Relies on good decomposition.
4. Monomial basis: *Frobenius companion matrix*.

Theorem (Stetter, Sticklberger, ...)

The *eigenvalues* of the $\boldsymbol{M}_j$ are $(x_\omega)_j$ and the *eigenvectors* ℓ_ω, $\omega \in \Omega$.

Completing the Job: Companion Matrices

UNIVERSITÄT PASSAU

Multiplication tables for ideal projectors

1. Multiplication modulo ideal: $\mathscr{P} \ni p \mapsto L_{\mathscr{P}}\left((\cdot)_j p\right) \in \mathscr{P}$.
2. Linear operation on $\mathscr{P} \to$ matrix $\boldsymbol{M}_j$ for a basis P of $\mathscr{P}$.
3. Implicitly computable: *reduction*. Relies on good decomposition.
4. Monomial basis: *Frobenius companion matrix*.

$$L_{\mathscr{P}} p = \sum_{\omega \in \Omega} p(x_\omega)\, \ell_\omega, \qquad \ell_\omega(\omega') = \delta_{\omega,\omega'}$$

Theorem (Stetter, Sticklberger, …)

The *eigenvalues* of the $\boldsymbol{M}_j$ are $(x_\omega)_j$ and the *eigenvectors* ℓ_ω, $\omega \in \Omega$.

Completing the Job: Companion Matrices

Multiplication tables for ideal projectors

1. Multiplication modulo ideal: $\mathscr{P} \ni p \mapsto L_{\mathscr{P}}\left((\cdot)_j p\right) \in \mathscr{P}$.
2. Linear operation on $\mathscr{P} \to$ matrix $\boldsymbol{M}_j$ for a basis P of $\mathscr{P}$.
3. Implicitly computable: *reduction*. Relies on good decomposition.
4. Monomial basis: *Frobenius companion matrix*.

$$L_{\mathscr{P}}\left((\cdot)_j \ell_\omega\right)$$

Theorem (Stetter, Sticklberger, ...)

The *eigenvalues* of the $\boldsymbol{M}_j$ are $(x_\omega)_j$ and the *eigenvectors* ℓ_ω, $\omega \in \Omega$.

Completing the Job: Companion Matrices

Multiplication tables for ideal projectors

1. Multiplication modulo ideal: $\mathscr{P} \ni p \mapsto L_{\mathscr{P}}\left((\cdot)_j p\right) \in \mathscr{P}$.
2. Linear operation on $\mathscr{P}$ → matrix $\boldsymbol{M}_j$ for a basis P of $\mathscr{P}$.
3. Implicitly computable: *reduction*. Relies on good decomposition.
4. Monomial basis: *Frobenius companion matrix*.

$$L_{\mathscr{P}}\left((\cdot)_j \ell_\omega\right) = \sum_{\omega' \in \Omega} \left((x_{\omega'})_j \, \ell_\omega(x_{\omega'})\right) \, \ell_{\omega'}$$

Theorem (Stetter, Sticklberger, ...)

The *eigenvalues* of the $\boldsymbol{M}_j$ are $(x_\omega)_j$ and the *eigenvectors* ℓ_ω, $\omega \in \Omega$.

Completing the Job: Companion Matrices

UNIVERSITÄT PASSAU

Multiplication tables for ideal projectors

1. Multiplication modulo ideal: $\mathscr{P} \ni p \mapsto L_{\mathscr{P}}\left((\cdot)_j p \right) \in \mathscr{P}$.
2. Linear operation on $\mathscr{P} \to$ matrix $\boldsymbol{M}_j$ for a basis P of $\mathscr{P}$.
3. Implicitly computable: *reduction*. Relies on good decomposition.
4. Monomial basis: *Frobenius companion matrix*.

$$L_{\mathscr{P}}\left((\cdot)_j \ell_\omega \right) = \sum_{\omega' \in \Omega} (x_{\omega'})_j \, \delta_{\omega,\omega'} \, \ell_{\omega'}$$

Theorem (Stetter, Sticklberger, ...)

The *eigenvalues* of the $\boldsymbol{M}_j$ are $(x_\omega)_j$ and the *eigenvectors* ℓ_ω, $\omega \in \Omega$.

Completing the Job: Companion Matrices

Multiplication tables for ideal projectors

1. Multiplication modulo ideal: $\mathscr{P} \ni p \mapsto L_{\mathscr{P}}\left((\cdot)_j p\right) \in \mathscr{P}$.
2. Linear operation on $\mathscr{P} \to$ matrix $\boldsymbol{M}_j$ for a basis P of $\mathscr{P}$.
3. Implicitly computable: *reduction*. Relies on good decomposition.
4. Monomial basis: *Frobenius companion matrix*.

$$L_{\mathscr{P}}\left((\cdot)_j \ell_\omega\right) = (x_\omega)_j \, \ell_\omega$$

Theorem (Stetter, Sticklberger, ...)

The *eigenvalues* of the $\boldsymbol{M}_j$ are $(x_\omega)_j$ and the *eigenvectors* ℓ_ω, $\omega \in \Omega$.

Completing the Job: Companion Matrices

Multiplication tables for ideal projectors

1. Multiplication modulo ideal: $\mathscr{P} \ni p \mapsto L_{\mathscr{P}}\left((\cdot)_j p \right) \in \mathscr{P}$.
2. Linear operation on $\mathscr{P} \to$ matrix $\boldsymbol{M}_j$ for a basis P of $\mathscr{P}$.
3. Implicitly computable: *reduction*. Relies on good decomposition.
4. Monomial basis: *Frobenius companion matrix*.

$$L_{\mathscr{P}}\left((\cdot)_j \ell_\omega \right) = (x_\omega)_j \, \ell_\omega$$

Theorem (Stetter, Sticklberger, ...)

The *eigenvalues* of the $\boldsymbol{M}_j$ are $(x_\omega)_j$ and the *eigenvectors* $\boldsymbol{\ell}_\omega$, $\omega \in \Omega$.

Some Results

Random frequencies & coefficients, real, 100 tests

parameters			average error		max error	
s	# freq.	n	coeff	freq	coeff	freq
2	5	3	1.3688e-11	1.8332e-09	3.5131e-09	2.4165e-07
2	10	5	4.9366e-08	2.6388e-06	7.3010e-05	5.3330e-04
2	15	8	7.0614e-07	2.9725e-04	1.4659e-04	4.4493e-02
2	20	9	Inf	Inf	NaN	NaN
3	20	6	1.5874e-08	1.4165e-06	4.7337e-05	8.9382e-04
4	20	5	8.4712e-12	4.6565e-11	9.0309e-09	3.7456e-09
5	20	5	1.6879e-12	5.9416e-11	1.9510e-09	1.3243e-08
5	50	5	1.1079e-10	6.6070e-10	3.1709e-07	6.6913e-08
5	100	6	2.9307e-09	1.9431e-08	1.0034e-05	1.3912e-06
5	150	8	1.3142e-08	8.4199e-08	5.7281e-06	4.3975e-06

Some Results

Random frequencies, purely imaginary, 100 tests

parameters			average error		max error	
s	# freq.	n	coeff	freq	coeff	freq
2	10	5	1.3476e-14	3.4744e-13	6.0290e-12	1.3724e-10
2	20	7	2.5148e-14	1.2420e-12	3.2103e-11	7.8847e-10
2	50	11	5.9357e-14	3.9721e-12	1.1845e-10	5.5214e-09
2	100	15	9.0480e-13	5.7684e-11	8.8308e-09	2.0468e-07
5	100	6	2.3796e-15	4.3794e-15	3.1431e-11	3.2918e-14
5	150	8	2.3954e-15	4.7773e-15	1.1702e-11	6.9726e-14

Observations

1. Performs very well:

2. be combined with hyperbolic cross.

Some Results

Random frequencies, purely imaginary, 100 tests

parameters			average error		max error	
s	# freq.	n	coeff	freq	coeff	freq
2	10	5	1.3476e-14	3.4744e-13	6.0290e-12	1.3724e-10
2	20	7	2.5148e-14	1.2420e-12	3.2103e-11	7.8847e-10
2	50	11	5.9357e-14	3.9721e-12	1.1845e-10	5.5214e-09
2	100	15	9.0480e-13	5.7684e-11	8.8308e-09	2.0468e-07
5	100	6	2.3796e-15	4.3794e-15	3.1431e-11	3.2918e-14
5	150	8	2.3954e-15	4.7773e-15	1.1702e-11	6.9726e-14

Observations

1. Performs very well:

2. be combined with hyperbolic cross.

Some Results

UNIVERSITÄT PASSAU

Random frequencies, purely imaginary, 100 tests

parameters			average error		max error	
s	# freq.	n	coeff	freq	coeff	freq
2	10	5	1.3476e-14	3.4744e-13	6.0290e-12	1.3724e-10
2	20	7	2.5148e-14	1.2420e-12	3.2103e-11	7.8847e-10
2	50	11	5.9357e-14	3.9721e-12	1.1845e-10	5.5214e-09
2	100	15	9.0480e-13	5.7684e-11	8.8308e-09	2.0468e-07
5	100	6	2.3796e-15	4.3794e-15	3.1431e-11	3.2918e-14
5	150	8	2.3954e-15	4.7773e-15	1.1702e-11	6.9726e-14

Observations

1. Performs very well:
2. be combined with hyperbolic cross.

Some Results

Random frequencies, purely imaginary, 100 tests

parameters			average error		max error	
s	# freq.	n	coeff	freq	coeff	freq
2	10	5	1.3476e-14	3.4744e-13	6.0290e-12	1.3724e-10
2	20	7	2.5148e-14	1.2420e-12	3.2103e-11	7.8847e-10
2	50	11	5.9357e-14	3.9721e-12	1.1845e-10	5.5214e-09
2	100	15	9.0480e-13	5.7684e-11	8.8308e-09	2.0468e-07
5	100	6	2.3796e-15	4.3794e-15	3.1431e-11	3.2918e-14
5	150	8	2.3954e-15	4.7773e-15	1.1702e-11	6.9726e-14

Observations

1. Performs very well:
   $\#\Omega = 200$, $s = 13$: 2180 quartic equations in 154.26s, accuracy $\sim 10^{-14}$
2. be combined with hyperbolic cross.

Some Results

Random frequencies, purely imaginary, 100 tests

parameters			average error		max error	
s	# freq.	n	coeff	freq	coeff	freq
2	10	5	1.3476e-14	3.4744e-13	6.0290e-12	1.3724e-10
2	20	7	2.5148e-14	1.2420e-12	3.2103e-11	7.8847e-10
2	50	11	5.9357e-14	3.9721e-12	1.1845e-10	5.5214e-09
2	100	15	9.0480e-13	5.7684e-11	8.8308e-09	2.0468e-07
5	100	6	2.3796e-15	4.3794e-15	3.1431e-11	3.2918e-14
5	150	8	2.3954e-15	4.7773e-15	1.1702e-11	6.9726e-14

Observations

1. Performs very well:
   $\#\Omega = 200$, $s = 13$: 2180 quartic equations in 154.26s, accuracy $\sim 10^{-14}$
2. Can be combined with hyperbolic cross.

Some Results

UNIVERSITÄT PASSAU

Random frequencies, purely imaginary, 100 tests

parameters			average error		max error	
s	# freq.	n	coeff	freq	coeff	freq
2	10	5	1.3476e-14	3.4744e-13	6.0290e-12	1.3724e-10
2	20	7	2.5148e-14	1.2420e-12	3.2103e-11	7.8847e-10
2	50	11	5.9357e-14	3.9721e-12	1.1845e-10	5.5214e-09
2	100	15	9.0480e-13	5.7684e-11	8.8308e-09	2.0468e-07
5	100	6	2.3796e-15	4.3794e-15	3.1431e-11	3.2918e-14
5	150	8	2.3954e-15	4.7773e-15	1.1702e-11	6.9726e-14

Observations

1. Performs very well:
   $\#\Omega = 200$, $s = 13$: 2180 quartic equations in 154.26s, accuracy $\sim 10^{-14}$
2. **Must** be combined with hyperbolic cross.

Some Results

Perturbation of imaginary input data, 100 tests, 100 frequencies

parameters			average error		max error	
s	ε	fail	coeff	freq	coeff	freq
5	10^{-5}	0	3.7885e-08	1.1462e-06	2.0235e-06	1.3860e-05
5	10^{-7}	0	3.7916e-10	1.1133e-08	2.1059e-08	7.8396e-08
5	10^{-10}	0	3.7221e-13	1.1200e-11	1.5896e-11	1.6209e-10
3	10^{-4}	27	0.0023822	0.0791873	5.3002	148.6645
4	10^{-4}	1	1.2563e-04	5.9020e+01	1.1638e+00	2.9213e+05
5	10^{-4}	2	3.7969e-07	1.1228e-05	6.8484e-06	7.1493e-05
10	10^{-4}	0	1.2672e-07	3.7955e-06	7.7848e-07	1.4004e-05

Explanation

1. SVD threshold tolerance adapted to error.
2. Compensates errors smaller than “conditioning”.

Some Results

UNIVERSITÄT PASSAU

Perturbation of imaginary input data, 100 tests, 100 frequencies

parameters			average error		max error	
s	ε	fail	coeff	freq	coeff	freq
5	10^{-5}	0	3.7885e-08	1.1462e-06	2.0235e-06	1.3860e-05
5	10^{-7}	0	3.7916e-10	1.1133e-08	2.1059e-08	7.8396e-08
5	10^{-10}	0	3.7221e-13	1.1200e-11	1.5896e-11	1.6209e-10
3	10^{-4}	27	0.0023822	0.0791873	5.3002	148.6645
4	10^{-4}	1	1.2563e-04	5.9020e+01	1.1638e+00	2.9213e+05
5	10^{-4}	2	3.7969e-07	1.1228e-05	6.8484e-06	7.1493e-05
10	10^{-4}	0	1.2672e-07	3.7955e-06	7.7848e-07	1.4004e-05

Explanation

1. SVD threshold tolerance adapted to error.
2. Compensates errors smaller than "conditioning".

Some Results

UNIVERSITÄT PASSAU

Perturbation of imaginary input data, 100 tests, 100 frequencies

parameters			average error		max error	
s	ε	fail	coeff	freq	coeff	freq
5	10^{-5}	0	3.7885e-08	1.1462e-06	2.0235e-06	1.3860e-05
5	10^{-7}	0	3.7916e-10	1.1133e-08	2.1059e-08	7.8396e-08
5	10^{-10}	0	3.7221e-13	1.1200e-11	1.5896e-11	1.6209e-10
3	10^{-4}	27	0.0023822	0.0791873	5.3002	148.6645
4	10^{-4}	1	1.2563e-04	5.9020e+01	1.1638e+00	2.9213e+05
5	10^{-4}	2	3.7969e-07	1.1228e-05	6.8484e-06	7.1493e-05
10	10^{-4}	0	1.2672e-07	3.7955e-06	7.7848e-07	1.4004e-05

Explanation

1. SVD threshold tolerance adapted to error.
2. Compensates errors smaller than “conditioning”.

Some Results

Perturbation of imaginary input data, 100 tests, 100 frequencies

parameters			average error		max error	
s	ε	fail	coeff	freq	coeff	freq
5	10^{-5}	0	3.7885e-08	1.1462e-06	2.0235e-06	1.3860e-05
5	10^{-7}	0	3.7916e-10	1.1133e-08	2.1059e-08	7.8396e-08
5	10^{-10}	0	3.7221e-13	1.1200e-11	1.5896e-11	1.6209e-10
3	10^{-4}	27	0.0023822	0.0791873	5.3002	148.6645
4	10^{-4}	1	1.2563e-04	5.9020e+01	1.1638e+00	2.9213e+05
5	10^{-4}	2	3.7969e-07	1.1228e-05	6.8484e-06	7.1493e-05
10	10^{-4}	0	1.2672e-07	3.7955e-06	7.7848e-07	1.4004e-05

Explanation

1. SVD threshold tolerance adapted to error.
2. Compensates errors smaller than “conditioning”.

Summary

UNIVERSITÄT PASSAU

Prony's problem in several variables ...

1. ... is interpolation & ideal theory.
2. ... motivation for universal interpolation.
3. ... efficiently solvable.

To do

1. Quantitative analysis, error estimates.
2. Good implementation

It's on the `arXiv`!

- T. Sauer, *Prony's method in several variables*. `arXiv:1602.02352`
- T. Sauer, *Prony's method in several variables: symbolic solutions by universal interpolation*. `arXiv:1603.03944`

Summary

UNIVERSITÄT PASSAU

Prony's problem in several variables ...

1. ... is interpolation & ideal theory.
2. ... motivation for universal interpolation.
3. ... efficiently solvable.

To do

1. Quantitative analysis, error estimates.
2. Good implementation

It's on the `arXiv`!

- T. Sauer, *Prony's method in several variables*. `arXiv:1602.02352`
- T. Sauer, *Prony's method in several variables: symbolic solutions by universal interpolation*. `arXiv:1603.03944`

Summary

Prony's problem in several variables . . .

1. . . . is interpolation & ideal theory.
2. . . . motivation for universal interpolation.
3. . . . efficiently solvable.

To do

1. Quantitative analysis, error estimates.
2. Good implementation

It's on the `arXiv`!

- T. Sauer, *Prony's method in several variables*. `arXiv:1602.02352`
- T. Sauer, *Prony's method in several variables: symbolic solutions by universal interpolation*. `arXiv:1603.03944`

Summary

Prony's problem in several variables ...

1. ... is interpolation & ideal theory.
2. ... motivation for universal interpolation.
3. ... efficiently solvable.

To do

1. Quantitative analysis, error estimates.
2. Good implementation

It's on the `arXiv`!

- T. Sauer, *Prony's method in several variables.* `arXiv:1602.02352`
- T. Sauer, *Prony's method in several variables: symbolic solutions by universal interpolation.* `arXiv:1603.03944`

Summary

Prony's problem in several variables ...

1. ... is interpolation & ideal theory.
2. ... motivation for universal interpolation.
3. ... efficiently solvable.

To do

1. Quantitative analysis, error estimates.
2. Good implementation

It's on the `arXiv`!

- T. Sauer, *Prony's method in several variables*. `arXiv:1602.02352`
- T. Sauer, *Prony's method in several variables: symbolic solutions by universal interpolation*. `arXiv:1603.03944`

Summary

Prony's problem in several variables . . .

1. . . . is interpolation & ideal theory.
2. . . . motivation for universal interpolation.
3. . . . efficiently solvable.

To do

1. Quantitative analysis, error estimates.
2. Good implementation

It's on the `arXiv`!

- T. Sauer, *Prony's method in several variables.* `arXiv:1602.02352`
- T. Sauer, *Prony's method in several variables: symbolic solutions by universal interpolation.* `arXiv:1603.03944`

Summary

Prony's problem in several variables . . .

1. . . . is interpolation & ideal theory.
2. . . . motivation for universal interpolation.
3. . . . efficiently solvable.

To do

1. Quantitative analysis, error estimates.
2. Good implementation

It's on the `arXiv`!

- T. Sauer, *Prony's method in several variables*. `arXiv:1602.02352`
- T. Sauer, *Prony's method in several variables: symbolic solutions by universal interpolation*. `arXiv:1603.03944`

Summary

Prony's problem in several variables …

1. …is interpolation & ideal theory.
2. …motivation for universal interpolation.
3. …efficiently solvable.

To do

1. Quantitative analysis, error estimates.
2. Good implementation & application?

It's on the `arXiv`!

- T. Sauer, *Prony's method in several variables*. `arXiv:1602.02352`
- T. Sauer, *Prony's method in several variables: symbolic solutions by universal interpolation*. `arXiv:1603.03944`

Summary

Prony's problem in several variables ...

1. ...is interpolation & ideal theory.
2. ...motivation for universal interpolation.
3. ...efficiently solvable.

To do

1. Quantitative analysis, error estimates.
2. Good implementation & application?

It's on the `arXiv`!

- T. Sauer, *Prony's method in several variables*. `arXiv:1602.02352`
- T. Sauer, *Prony's method in several variables: symbolic solutions by universal interpolation*. `arXiv:1603.03944`

Summary

UNIVERSITÄT PASSAU

Prony's problem in several variables ...

1. ... is interpolation & ideal theory.
2. ... motivation for universal interpolation.
3. ... efficiently solvable.

To do

1. Quantitative analysis, error estimates.
2. Good implementation & application?

It's on the `arXiv`!

- T. Sauer, *Prony's method in several variables*. `arXiv:1602.02352`
- T. Sauer, *Prony's method in several variables: symbolic solutions by universal interpolation*. `arXiv:1603.03944`

Thank you for your attention!

Passau

Where three rivers meet . . .

Passau

...lies the "Bavarian Venice" ...

Passau

. . . lies the "Bavarian Venice" . . .

Passau

. . . lies the "Bavarian Venice" . . .

Passau

. . . with a "University at the beach" . . .

Passau

…with a "University at the beach" …

Passau

. . . with a "University at the beach" . . .

Passau

UNIVERSITÄT PASSAU

. . . and great students

Passau

…and great students

Start