Sparse Interpolation

Annie Cuyt and Wen-shin Lee

Canazei Workshop 7–12/9/2014

Universiteit Antwerpen
Department of Mathematics
and Computer Science

Motivation

Motivation

interpolate

$$f(x) = \alpha_1 + \alpha_2 x^{100}$$

- Newton/Lagrange interpolation: 101 samples
- only 4 unknowns: α_1, α_2, x^0, x^{100}!
- how to solve it from 4 samples?

Motivation

- exponential analysis
- generalized eigenvalue problems
- computer algebra
- orthogonal polynomials
- signal processing
- moment problems
- nonlinear approximation theory
- many applications ...

Motivation

$$x_s = s\Delta, \quad s = 0, 1, 2, \ldots$$

$$\sum_{i=1}^{n} \alpha_i x_s^{k_i} = f_s, \quad n \ll \max(k_i), \quad k_i \in \mathbb{N}$$

$$\sum_{i=1}^{n_1} \alpha_{i,1} \cos(\phi_{i,1} x_s) + \sum_{i=1}^{n_2} \alpha_{i,2} \sin(\phi_{i,2} x_s) = f_s$$

$$\sum_{i=1}^{n} \alpha_i \exp(\phi_i x_s) = f_s, \quad \phi_i \in \mathbb{C}$$

Motivation

1. Univariate exponential sparse interpolation
   (Exercise)
2. Multivariate polynomial sparse interpolation
   (Exercise)
3. Connection with rational approximation theory
   (Exercise)
4. Applications unlimited

Basics: exponential

Basics: exponential

Figure: Gaspard Riche de Prony [1795]

Basics: exponential

interpolation problem:

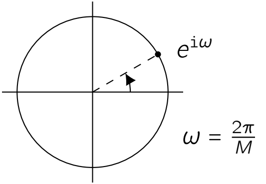
$$\sum_{i=1}^{n} \alpha_i \exp(\phi_i x_s) = f_s, \quad s = 0, \ldots, 2n-1$$

$$x_s = s\frac{2\pi}{M}, \quad \omega = 2\pi/M$$

$$|\Im(\phi_i)| < M/2, \quad \Omega_i = \exp(\phi_i \omega),$$

$$f_s = \sum_{i=1}^{n} \alpha_i \Omega_i^s, \quad s = 0, \ldots, 2n-1$$

$$\begin{cases} \alpha_1 + \cdots + \alpha_n = f_0 \\ \alpha_1 \Omega_1 + \cdots + \alpha_n \Omega_n = f_1 \\ \quad \vdots \\ \alpha_1 \Omega_1^{2n-1} + \cdots + \alpha_n \Omega_n^{2n-1} = f_{2n-1} \end{cases}$$



Universiteit Antwerpen

Basics: exponential

finding Ω_i:

$$\prod_{i=1}^{n}(z-\Omega_i) = z^n + b_{n-1}z^{n-1} + \cdots + b_1 z + b_0$$

$$\begin{aligned} 0 &= \sum_{i=1}^{n} \alpha_i \Omega_i^s (\Omega_i^n + b_{n-1}\Omega_i^{n-1} + \cdots + b_0) \\ &= \sum_{i=1}^{n} \alpha_i \Omega_i^{n+s} + \sum_{j=0}^{n-1} b_j \left(\sum_{i=1}^{n} \alpha_i \Omega_i^{j+s} \right) \\ &= f_{s+n} + \sum_{j=0}^{n-1} b_j f_{s+j} \end{aligned}$$

Basics: exponential

$$\begin{pmatrix} f_0 & \cdots & f_{n-1} \\ \vdots & \iddots & \vdots \\ f_{n-1} & \cdots & f_{2n-2} \end{pmatrix} \begin{pmatrix} b_0 \\ \vdots \\ b_{n-1} \end{pmatrix} = - \begin{pmatrix} f_n \\ \vdots \\ f_{2n-1} \end{pmatrix}$$

Hankel matrix:

$$H_n^{(r)} = \begin{pmatrix} f_r & \cdots & f_{r+n-1} \\ \vdots & \iddots & \vdots \\ f_{r+n-1} & \cdots & f_{r+2n-2} \end{pmatrix}$$

Basics: exponential

Hadamard polynomial:

$$H_n^{(0)}(z) = \begin{vmatrix} f_0 & \dots & f_{n-1} & f_n \\ \vdots & \iddots & \vdots & \vdots \\ f_{n-1} & \dots & f_{2n-2} & f_{2n-1} \\ 1 & \dots & z^{n-1} & z^n \end{vmatrix}$$

$$\prod_{i=1}^{n} (z - \Omega_i) = \frac{H_n^{(0)}(z)}{\left|H_n^{(0)}\right|} = z^n + b_{n-1} z^{n-1} + \cdots + b_1 z + b_0$$

Basics: exponential

formally orthogonal polynomial:

$$\gamma : z^s \to f_s, \quad s = 0, 1, \ldots$$

$$\gamma : \exp(\phi_i x_s) = \Omega_i^s \to \sum_{i=1}^{n} \alpha_i \Omega_i^s = f_s$$

$$\gamma : z^i \frac{H_n^{(0)}(z)}{\left|H_n^{(0)}\right|} \to 0, \quad i = 0, \ldots, n-1$$

$$\frac{H_n^{(0)}(z)}{\left|H_n^{(0)}\right|} \perp_\gamma z^i, \quad i = 0, \ldots, n-1$$

[Henrici, 1974]

Basics: exponential

roots of $\dfrac{H_n^{(0)}(z)}{\left|H_n^{(0)}\right|}$ from GEP:

$$H_n^{(0)} = \begin{pmatrix} 1 & \dots & & 1 \\ \Omega_1 & \Omega_2 & \dots & \Omega_n \\ \vdots & & & \vdots \\ \Omega_1^{n-1} & \dots & & \Omega_n^{n-1} \end{pmatrix} \begin{pmatrix} \alpha_1 & & \\ & \ddots & \\ & & \alpha_n \end{pmatrix} \begin{pmatrix} 1 & \Omega_1 & \dots & \Omega_1^{n-1} \\ \vdots & \Omega_2 & & \vdots \\ & \vdots & & \vdots \\ 1 & \Omega_n & \dots & \Omega_n^{n-1} \end{pmatrix}$$
$$= V_n^T D_\alpha V_n$$

$$H_n^{(1)} = V_n^T D_\alpha \begin{pmatrix} \Omega_1 & & \\ & \ddots & \\ & & \Omega_n \end{pmatrix} V_n$$

Basics: exponential

$$\det\left(H_n^{(1)} - \lambda H_n^{(0)}\right) = \det\left(V_n^T D_\alpha \begin{pmatrix} \Omega_1 - \lambda & & \\ & \ddots & \\ & & \Omega_n - \lambda \end{pmatrix} V_n \right)$$
$$= 0 \text{ for } \lambda = \Omega_i, \quad i = 1, \ldots, n$$

[Hua and Sarkar, 1990]

Basics: exponential

finding ϕ_i:

$$\exp(\phi_i) = \exp(\Re(\phi_i))e^{\mathrm{i}\Im(\phi_i)}$$

$$|\Im(\phi_i)| < \frac{M}{2}:$$

$$\begin{aligned}\arg(\Omega_i) &= \arg(\exp(\phi_i\omega)) \\ &= \Im(\phi_i)\,\frac{2\pi}{M} \in \left]-\pi, \pi\right[\end{aligned}$$

Basics: exponential

finding α_i:

$$\sum_{i=1}^{n} \alpha_i \Omega_i^{s+j} = f_{s+j}, \quad s = 0, \ldots, n-1, \quad 0 \le j \le n$$

$$\begin{pmatrix} \Omega_1^j & \ldots & \Omega_n^j \\ \vdots & & \vdots \\ \Omega_1^{j+n-1} & \ldots & \Omega_n^{j+n-1} \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{pmatrix} = \begin{pmatrix} f_j \\ \vdots \\ f_{j+n-1} \end{pmatrix}$$

remaining interpolation conditions are linearly dependent

Basics: exponential

finding n:

$$N < n : \left| H_N^{(r)} \right| \not\equiv 0, \quad r = 0, 1, \ldots$$

$$N = n : \left| H_N^{(r)} \right| \neq 0 \quad \text{if } \Omega_i \neq \Omega_j \text{ for } i \neq j$$ [Kaltofen and Lee, 2003]

$$N > n : \left| H_N^{(r)} \right| \equiv 0, \quad r = 0, 1, \ldots$$

Example: exponential

$$\phi(x) = \sum_{i=1}^{4} \alpha_i \exp(\phi_i x)$$

$\alpha_1 = 1$	$\phi_1 = 0$
$\alpha_2 = 2.4$	$\phi_2 = -5 + 19.97\mathtt{i}$
$\alpha_3 = -2.1$	$\phi_3 = 3 + 45\mathtt{i}$
$\alpha_4 = 0.2$	$\phi_4 = 5.3\mathtt{i}$

evaluate at $x_s = s\frac{2\pi}{100}$, $M = 100$, $|\Im(\phi_i)| < 50$

sequence $f_0, \ldots, f_7, \ldots$ is linearly generated

Example: exponential

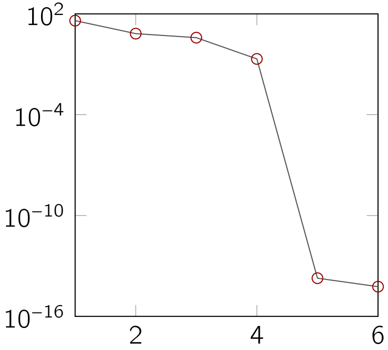


Figure: $H_N^{(0)}$ singular, $N = 6$

$$\frac{|\widetilde{\Omega}_j - \Omega_j|}{|\Omega_j|} \le 2 \times 10^{-12}, \qquad \frac{|\widetilde{\phi}_j - \phi_j|}{|\phi_j|} \le 2 \times 10^{-12}$$

Exercise: exponential

$$\phi(x) = \alpha_1 \exp(\phi_1 x) + \alpha_2 \exp(\phi_2 x)$$

4 unknowns $\phi_1, \phi_2, \alpha_1, \alpha_2$

identify $\phi(x)$ from

$$\begin{aligned}
\phi(0) &= \alpha_1 + \alpha_2 \\
\phi'(0) &= \alpha_1 \phi_1 + \alpha_2 \phi_2 \\
\phi''(0) &= \alpha_1 \phi_1^2 + \alpha_2 \phi_2^2 \\
\phi'''(0) &= \alpha_1 \phi_1^3 + \alpha_2 \phi_2^3
\end{aligned}$$

solution:

$$\left| H_3^{(0)} \right| = \begin{vmatrix} \phi_0 & \phi_0' & \phi_0'' \\ \phi_0' & \phi_0'' & \phi_0''' \\ \phi_0'' & \phi_0''' & \phi_0^{\prime v} \end{vmatrix} = 0 \quad \text{symbolically}$$

$n = 2$

```
> ##########################
  ## Exercise: exponential ##
  ##########################
> restart:
> with(LinearAlgebra):
> Phi[0] := alpha[1] * exp(phi[1]*x) + alpha[2] * exp(phi[2]*x);
```

$$\Phi_0 := \alpha_1 e^{\phi_1 x} + \alpha_2 e^{\phi_2 x} \quad (1$$

```
> for i from 1 to 4 do
   Phi[i] := diff(Phi[i-1], x);
  od;
```

$$\Phi_1 := \alpha_1 \phi_1 e^{\phi_1 x} + \alpha_2 \phi_2 e^{\phi_2 x}$$

$$\Phi_2 := \alpha_1 \phi_1^2 e^{\phi_1 x} + \alpha_2 \phi_2^2 e^{\phi_2 x}$$

$$\Phi_3 := \alpha_1 \phi_1^3 e^{\phi_1 x} + \alpha_2 \phi_2^3 e^{\phi_2 x}$$

$$\Phi_4 := \alpha_1 \phi_1^4 e^{\phi_1 x} + \alpha_2 \phi_2^4 e^{\phi_2 x} \quad (2$$

```
> for i from 0 to 4 do
   f[i] := simplify(subs(x=0, Phi[i]));
  od;
```

$$f_0 := \alpha_1 + \alpha_2$$

$$f_1 := \alpha_1 \phi_1 + \alpha_2 \phi_2$$

$$f_2 := \alpha_1 \phi_1^2 + \alpha_2 \phi_2^2$$

$$f_3 := \alpha_1 \phi_1^3 + \alpha_2 \phi_2^3$$

$$f_4 := \alpha_1 \phi_1^4 + \alpha_2 \phi_2^4 \quad (3$$

```
> H[3] := HankelMatrix([f[0],f[1],f[2],f[3],f[4]],3);
```

$$H_3 := \begin{bmatrix} \alpha_1 + \alpha_2 & \alpha_1 \phi_1 + \alpha_2 \phi_2 & \alpha_1 \phi_1^2 + \alpha_2 \phi_2^2 \\ \alpha_1 \phi_1 + \alpha_2 \phi_2 & \alpha_1 \phi_1^2 + \alpha_2 \phi_2^2 & \alpha_1 \phi_1^3 + \alpha_2 \phi_2^3 \\ \alpha_1 \phi_1^2 + \alpha_2 \phi_2^2 & \alpha_1 \phi_1^3 + \alpha_2 \phi_2^3 & \alpha_1 \phi_1^4 + \alpha_2 \phi_2^4 \end{bmatrix} \quad (4$$

```
> Determinant(H[3]);
```

$$0 \quad (5$$

```
> H[2] := HankelMatrix([f[0],f[1],f[2]],2);
```

$$H_2 := \begin{bmatrix} \alpha_1 + \alpha_2 & \alpha_1 \phi_1 + \alpha_2 \phi_2 \\ \alpha_1 \phi_1 + \alpha_2 \phi_2 & \alpha_1 \phi_1^2 + \alpha_2 \phi_2^2 \end{bmatrix} \quad (6$$

```
> F2 := Matrix([[f[2]],[f[3]]]);
```

$$F2 := \begin{bmatrix} \alpha_1 \phi_1^2 + \alpha_2 \phi_2^2 \\ \alpha_1 \phi_1^3 + \alpha_2 \phi_2^3 \end{bmatrix} \quad (7$$

```
> B:=LinearSolve(H[2],-F2);
```

$$B := \begin{bmatrix} \phi_1 \phi_2 \\ -\phi_1 - \phi_2 \end{bmatrix} \quad (8$$

```
> B_poly := z^2 + B(2)*z + B(1);
```

$$B\_poly := z^2 + \left(-\phi_1 - \phi_2\right) z + \phi_1 \phi_2 \quad (9$$

```
> z_root:=solve(B_poly=0, z);
```

$$z\_root := \phi_2, \phi_1 \quad (10$$

```
> V := Transpose(VandermondeMatrix([z_root[1],z_root[2]],2,2));
```

$$V := \begin{bmatrix} 1 & 1 \\ \phi_2 & \phi_1 \end{bmatrix} \quad (11$$

```
> F0 := Matrix([[f[0]],[f[1]]]);
```

$$F0 := \begin{bmatrix} \alpha_1 + \alpha_2 \\ \alpha_1 \phi_1 + \alpha_2 \phi_2 \end{bmatrix} \quad (12$$

```
> A:=LinearSolve(V,F0);
```

$$A := \begin{bmatrix} \alpha_2 \\ \alpha_1 \end{bmatrix} \quad (13$$

```
> alpha[1] := Pi;
  alpha[2] := 5;
  phi[1] := 2;
  phi[2] := -1/7;
```

$$\alpha_1 := \pi$$

$$\alpha_2 := 5$$

$$\phi_1 := 2$$

$$\phi_2 := -\frac{1}{7} \quad (14$$

```
> for i from 0 to 4 do
   f[i] := simplify(subs(x=0, Phi[i]));
  od;
```

$$f_0 := \pi + 5$$

$$f_1 := 2\,\pi - \frac{5}{7}$$

$$f_2 := 4\,\pi + \frac{5}{49}$$

$$f_3 := 8\,\pi - \frac{5}{343}$$

$$f_4 := 16\,\pi + \frac{5}{2401} \quad (15$$

```
> H[3] := HankelMatrix([f[0],f[1],f[2],f[3],f[4]],3);
```

$$H_3 := \begin{bmatrix} \pi + 5 & 2\pi - \frac{5}{7} & 4\pi + \frac{5}{49} \\ 2\pi - \frac{5}{7} & 4\pi + \frac{5}{49} & 8\pi - \frac{5}{343} \\ 4\pi + \frac{5}{49} & 8\pi - \frac{5}{343} & 16\pi + \frac{5}{2401} \end{bmatrix} \quad (16$$

```
> Determinant(H[3]);
```

$$0 \quad (17$$

```
> H[2] := HankelMatrix([f[0],f[1],f[2]],2);
```

$$H_2 := \begin{bmatrix} \pi + 5 & 2\pi - \frac{5}{7} \\ 2\pi - \frac{5}{7} & 4\pi + \frac{5}{49} \end{bmatrix} \quad (18$$

```
> F2 := Matrix([[f[2]],[f[3]]]);
```

$$F2 := \begin{bmatrix} 4\pi + \frac{5}{49} \\ 8\pi - \frac{5}{343} \end{bmatrix} \quad (19$$

```
> B:=LinearSolve(H[2],-F2);
```

$$B := \begin{bmatrix} -\frac{2}{7} \\ -\frac{13}{7} \end{bmatrix} \quad (20$$

```
> B_poly := z^2 + B(2)*z + B(1);
```

$$B\_poly := z^2 - \frac{13}{7} z - \frac{2}{7} \quad (21$$

```
> z_root:=solve(B_poly=0,z);
```

$$z\_root := 2, -\frac{1}{7} \quad (22$$

```
> V := Transpose(VandermondeMatrix([z_root[1],z_root[2]],2,2));
```

$$V := \begin{bmatrix} 1 & 1 \\ 2 & -\frac{1}{7} \end{bmatrix} \quad (23$$

```
> F0 := Matrix([[f[0]],[f[1]]]);
```

$$F0 := \begin{bmatrix} \pi + 5 \\ 2\pi - \frac{5}{7} \end{bmatrix} \quad (24$$

```
> A:=LinearSolve(V,F0);
```

$$A := \begin{bmatrix} \pi \\ 5 \end{bmatrix} \quad (25$$

Basics: polynomial

Basics: polynomial

Figure: Michael Ben-Or and Prasoon Tiwari [1988]

Basics: polynomial

interpolation problem:

$$\sum_{(k_1,\ldots,k_d)\in K} \alpha_{k_1,\ldots,k_d} x_1^{k_1}\cdots x_d^{k_d}, \quad \#K = n$$

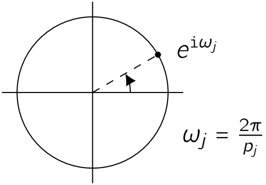
evaluate at

$$(x_1,\ldots,x_d) = (\omega_1^s,\ldots,\omega_d^s)$$

$$\omega_j = \exp(2\pi \mathtt{i}/p_j), \quad p_j > \partial_j p(x_1,\ldots,x_d), \quad p_j \text{ mutually prime}$$

$$p(\omega_1^s,\ldots,\omega_d^s) = f_s, \quad 0 \le s \le 2n-1$$

$$\Omega_i = \omega_1^{k_1^{(i)}}\cdots\omega_d^{k_d^{(i)}}, \quad i = 1,\ldots,n$$



[Giesbrecht, Labahn, and Lee, 2006]

Basics: polynomial

finding Ω_i:

$$\prod_{i=1}^{n}(z-\Omega_i) = z^n + b_{n-1}z^{n-1} + \cdots + b_1 z + b_0$$

$$\begin{aligned} 0 &= \sum_{i=1}^{n} \alpha_i \Omega_i^s (\Omega_i^n + b_{n-1}\Omega_i^{n-1} + \cdots + b_0) \\ &= \sum_{i=1}^{n} \alpha_i \Omega_i^{n+s} + \sum_{j=0}^{n-1} b_j \left(\sum_{i=1}^{n} \alpha_i \Omega_i^{j+s} \right) \\ &= f_{s+n} + \sum_{j=0}^{n-1} b_j f_{s+j} \end{aligned}$$

Basics: polynomial

$$\begin{pmatrix} f_0 & \dots & f_{n-1} \\ \vdots & \iddots & \vdots \\ f_{n-1} & \dots & f_{2n-2} \end{pmatrix} \begin{pmatrix} b_0 \\ \vdots \\ b_{n-1} \end{pmatrix} = - \begin{pmatrix} f_n \\ \vdots \\ f_{2n-1} \end{pmatrix}$$

Ω_i: zeros of formally orthogonal Hadamard polynomial

$$\prod_{i=1}^{n} (z - \Omega_i) = \frac{H_n^{(0)}(z)}{\left| H_n^{(0)} \right|}$$

Basics: polynomial

finding $(k_1^{(i)}, \dots, k_d^{(i)})$:

reverse Chinese remainder theorem

$$m = p_1 \cdots p_d$$

$$\Omega_i = \omega^{k(i)}, \quad \omega = \exp\left(\frac{2\pi \mathtt{i}}{\prod_{j=1}^{d} p_j}\right),$$

$$k(i) = k_1^{(i)} \frac{m}{p_1} + \cdots + k_d^{(i)} \frac{m}{p_d}$$

$$k_j^{(i)} \frac{m}{p_j} \bmod p_j = k(i) \bmod p_j,$$

$$k_j^{(i)} < p_j, \quad \gcd(p_j, m/p_j) = 1, \quad j = 1, \dots, d$$

Basics: polynomial

finding $\alpha_i = \alpha_{k_1^{(i)},\dots,k_d^{(i)}}$:

$$\sum_{i=1}^{n} \alpha_i \Omega_i^{s+j} = f_{s+j}, \quad s = 0, \dots, n-1, \quad 0 \le j \le n$$

$$\begin{pmatrix} \Omega_1^{j} & \dots & \Omega_n^{j} \\ \vdots & & \vdots \\ \Omega_1^{j+n-1} & \dots & \Omega_n^{j+n-1} \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_n \end{pmatrix} = \begin{pmatrix} f_j \\ \vdots \\ f_{j+n-1} \end{pmatrix}$$

remaining interpolation conditions are linearly dependent

Basics: polynomial

finding n:

floating-point arithmetic:

$$\left|H_N^{(r)}\right| \equiv 0, \quad N > n, \quad r = 0, 1, \ldots$$

exact arithmetic:
increase n till

$$\delta_s := f_s + b_{n-1} f_{s-1} + \cdots + b_0 f_{s-n}$$

equals 0, $s \geq 2n$

[Massey, 1969]

Example: polynomial

$$p(x,y) = x^5y + 2.2x^4y^4 - 0.5xy^{11} + 0.1xy^{12}$$

$$p_1 = 6, \quad p_2 = 13, \quad \omega_1 = \exp(2\pi \mathtt{i}/6), \quad \omega_2 = \exp(2\pi \mathtt{i}/13),$$

$p(\omega_1^s, \omega_2^s), \quad s = 0, \ldots, 7 \quad$ (floating-point)

or

$p(p_1^s, p_2^s), \quad s = 0, \ldots, 7 \quad$ (exact arithmetic)

sequence $f_0, f_1, \ldots, f_7, \ldots$ is linearly generated and $\delta_8 = 0$

Hadamard polynomial:

$$z^4 + (-3.67 + 0.0799\mathtt{i})z^3 + (5.35 - 0.216\mathtt{i})z^2$$
$$(-3.67 + 0.216\mathtt{i})z + (0.997 - 0.0805\mathtt{i})$$

Basics: polynomial

$$m = 78, \qquad \omega = \exp(2\pi \mathtt{i}/78),$$

$$\Omega_1 = \omega^{k(1)} = \omega^{71} = \omega^{5\times 13 + 1\times 6}$$

$$\Omega_2 = \omega^{k(2)} = \omega^{76} = \omega^{4\times 13 + 4\times 6}$$

$$\Omega_3 = \omega^{k(3)} = \omega^{79} = \omega^{1\times 13 + 11\times 6}$$

$$\Omega_4 = \omega^{k(4)} = \omega^{85} = \omega^{1\times 13 + 12\times 6}$$

$$K = \{(5,1),(4,4),(1,11),(1,12)\} \Rightarrow \text{ terms } x^5y, x^4y^4, xy^{11}, xy^{12}$$

Vandermonde system

$$\sum_{i=1}^{4} \alpha_i \Omega_i^{s+j} = f_{s+j}, \quad s = 0, \dots, n-1, \quad 0 \le j \le n$$

Exercise: polynomial

$$p(x,y) = (x-3)^5(y+5) + 2.2(x-3)^4(y+5)^4 - 0.5(x-3)(y+5)^{11} + 0.1(x-3)(y+5)^{12}$$
$$= u^5v + 2.2u^4v^4 - 0.5uv^{11} + 0.1uv^{12},$$
$$u = x-3,\ v = y+5$$

$$p_1 = 6, \qquad p_2 = 13,$$
$$\omega_1 = \exp(2\pi \mathtt{i}/6), \qquad \omega_2 = \exp(2\pi \mathtt{i}/13),$$
$$u = \omega_1^s, \qquad v = \omega_2^s$$

$$p(\omega_1^s + 3, \omega_2^s - 5), \quad s = 0, \dots, 7$$

Universiteit Antwerpen

```
> ##########################
  ## Exercise: polynomial ##
  ##########################
> restart:
> with(LinearAlgebra):
> read "./ExpLog.mm":
  read "./RevChRem.mm":
> Digits:= 16:
> poly_uv := u^5*v + 2.2*u^4*v^4-0.5*u*v^11+0.1*u*v^12;
```

$$poly\_uv := u^5\, v + 2.2\, u^4\, v^4 - 0.5\, u\, v^{11} + 0.1\, u\, v^{12} \quad (1)$$

```
> poly_xy := subs(u=x-3,v=y+5, poly_uv);
```

$$poly\_xy := (x-3)^5\,(y+5) + 2.2\,(x-3)^4\,(y+5)^4 - 0.5\,(x-3)\,(y+5)^{11} + 0.1\,(x-3)\,(y+5)^{12} \quad (2)$$

```
> expand(poly_xy);
```

$$-1.464750\;10^5\, x - 1.45595805\;10^7\, y + 1.101600\;10^5 + 72900.0\, x^2 - 3.21998325\;10^7\, y^2 + x^5\, y + 1085.0\, x^4\, y - 13110.0\, x^3\, y + 59130.0\, x^2\, y + 4.7644175\;10^6\, x\, y + 2.2\, x^4\, y^4 + 44.0\, x^4\, y^3 - 26.4\, x^3\, y^4 + 330.0\, x^4\, y^2 - 528.0\, x^3\, y^3 + 118.8\, x^2\, y^4 - 3960.0\, x^3\, y^2 + 2376.0\, x^2\, y^3 + 6.4450749\;10^6\, x\, y^4 + 17820.0\, x^2\, y^2 + 1.07374355\;10^7\, x\, y^3 + 1.07065475\;10^7\, x\, y^2 + 5.5\, x\, y^{11} + 137.5\, x\, y^{10} + 2062.5\, x\, y^9 + 20625.0\, x\, y^8 + 1.443750\;10^5\, x\, y^7 + 7.218750\;10^5\, x\, y^6 + 2.5781250\;10^6\, x\, y^5 + 0.1\, x\, y^{12} + 5\, x^5 + 1300.0\, x^4 - 16050.0\, x^3 - 1.93357593\;10^7\, y^4 - 3.22229985\;10^7\, y^3 - 16.5\, y^{11} - 412.5\, y^{10} - 6187.5\, y^9 - 61875.0\, y^8 - 4.331250\;10^5\, y^7 - 2.1656250\;10^6\, y^6 - 7.7343750\;10^6\, y^5 - 0.3\, y^{12} \quad (3)$$

```
> p1 := 6;
  p2 := 13;
```

$$p1 := 6$$
$$p2 := 13 \quad (4)$$

```
> w1 := evalf(exp(2*Pi*I/p1));
  w2 := evalf(exp(2*Pi*I/p2));
```

$$w1 := 0.5000000000000000 + 0.8660254037844385\, I$$
$$w2 := 0.8854560256532100 + 0.4647231720437684\, I \quad (5)$$

```
> for i from 0 to 7 do
    f[i] := evalf(subs(x=w1^i+3, y=w2^i-5, poly_xy)):
  od;
```

$$f_0 := 2.8 + 0.\, I$$

$$f_1 := 2.602841017834476 - 0.8741573457604577\, I$$
$$f_2 := 2.064816858508122 - 1.589980617941464\, I$$
$$f_3 := 1.329941999443174 - 2.035486797328322\, I$$
$$f_4 := 0.5888159453592064 - 2.177111443869176\, I$$
$$f_5 := 0.02061376974983473 - 2.067629883507751\, I$$
$$f_6 := -0.2610215691867326 - 1.827541993129119\, I$$
$$f_7 := -0.2414681965457515 - 1.605741666597298\, I$$ (6

```
> H0[4] := HankelMatrix([f[0],f[1],f[2],f[3],f[4],f[5],f[6]],4);
```

$H0_4$:= [[2.8 + 0. I, 2.602841017834476 − 0.8741573457604577 I, 2.064816858508122 − 1.589980617941464 I, 1.329941999443174 − 2.035486797328322 I], (7

[2.602841017834476 − 0.8741573457604577 I, 2.064816858508122 − 1.589980617941464 I, 1.329941999443174 − 2.035486797328322 I, 0.5888159453592064 − 2.177111443869176 I],

[2.064816858508122 − 1.589980617941464 I, 1.329941999443174 − 2.035486797328322 I, 0.5888159453592064 − 2.177111443869176 I, 0.02061376974983473 − 2.067629883507751 I],

[1.329941999443174 − 2.035486797328322 I, 0.5888159453592064 − 2.177111443869176 I, 0.02061376974983473 − 2.067629883507751 I, -0.2610215691867326 − 1.827541993129119 I]]

```
> H1[4] := HankelMatrix([f[1],f[2],f[3],f[4],f[5],f[6],f[7]],4);
```

$H1_4$:= [[2.602841017834476 − 0.8741573457604577 I, 2.064816858508122 − 1.589980617941464 I, 1.329941999443174 − 2.035486797328322 I, 0.5888159453592064 − 2.177111443869176 I], (8

[2.064816858508122 − 1.589980617941464 I, 1.329941999443174 − 2.035486797328322 I, 0.5888159453592064 − 2.177111443869176 I, 0.02061376974983473 − 2.067629883507751 I],

[1.329941999443174 − 2.035486797328322 I, 0.5888159453592064 − 2.177111443869176 I, 0.02061376974983473 − 2.067629883507751 I, -0.2610215691867326 − 1.827541993129119 I],

[0.5888159453592064 − 2.177111443869176 I, 0.02061376974983473 − 2.067629883507751 I, -0.2610215691867326 − 1.827541993129119 I, -0.2414681965457515 − 1.605741666597298 I]]

```
> eigenvalue_list := Eigenvalues(H1[4],H0[4], output=list);
```

eigenvalue_list := [0.8451900855580868 + 0.5344658261253400 I, 0.8451900855413997 − 0.5344658261265820 I, 0.9870502626364418 − 0.1604112808397337 I, 0.9967573080994497 + 0.08046656865700708 I] **(9**

```
> total_exp_list := []:
  for i from 1 to 4 do
    total_exp_list := [op(total_exp_list), round(ExpLog(eigenvalue_list[i],6*13))]:
  od:
> total_exp_list;
```

[7, 71, 76, 1] **(10**

```
> exp_list:=[]:
  for i from 1 to 4 do
    exp_list := [op(exp_list), [RevChRem(total_exp_list[i], 6, 6*13), RevChRem(total_exp_list[i], 13,
  6*13)]]:
  od:
> exp_list;
```

[[1, 12], [5, 1], [4, 4], [1, 11]] **(11**

```
> V := Transpose(VandermondeMatrix(eigenvalue_list));
```

V := [[1. + 0. I, 1. + 0. I, 1. + 0. I, 1. + 0. I], **(12**
[0.8451900855580868 + 0.5344658261253400 I, 0.8451900855413997 − 0.5344658261265820 I, 0.9870502626364418 − 0.1604112808397337 I, 0.9967573080994497 + 0.08046656865700708 I],
[0.4286925614298439 + 0.9034504346214993 I, 0.4286925614003088 − 0.9034504346057614 I, 0.9485364419500248 − 0.3166679937654143 I, 0.9870502625782285 + 0.1604112807331159 I],
[−0.1205366802302719 + 0.9927088741336253 I, −0.1205366802550991 − 0.9927088740899947 I, 0.8854560256661491 − 0.4647231719910745 I, 0.9709418173518603 + 0.2393156640939936 I]]

```
> F := Matrix([[f[0]],[f[1]],[f[2]],[f[3]]]);
```

$$F := \begin{bmatrix} 2.8 + 0.\,I \\ 2.602841017834476 - 0.8741573457604577\,I \\ 2.064816858508122 - 1.589980617941464\,I \\ 1.329941999443174 - 2.035486797328322\,I \end{bmatrix}$$ **(13**

```
> A := LinearSolve(V, F);
```

$$A := \begin{bmatrix} 0.1000000000041224 + 1.216000331768808\ 10^{-11}\ \mathrm{I} \\ 1.000000000007150 + 2.565636441003167\ 10^{-11}\ \mathrm{I} \\ 2.200000000261579 - 7.965585370092284\ 10^{-11}\ \mathrm{I} \\ -0.500000002728518 + 4.183975685423307\ 10^{-11}\ \mathrm{I} \end{bmatrix} \quad (14)$$

```
> var_list := [u,v]:

  p := 0:

  for i from 1 to 4 do
     term := A[i,1]:

     for j from 1 to 2 do
        term := term*var_list[j]^exp_list[i,j]:
     od:

     p := p + term:
  od:
```

```
> p;
```

$$(0.1000000000041224 + 1.216000331768808\ 10^{-11}\ \mathrm{I})\ u\ v^{12} + (1.000000000007150 + 2.565636441003167\ 10^{-11}\ \mathrm{I})\ u^5\ v + (2.200000000261579 - 7.965585370092284\ 10^{-11}\ \mathrm{I})\ u^4\ v^4 + (-0.500000002728518 + 4.183975685423307\ 10^{-11}\ \mathrm{I})\ u\ v^{11} \quad (15)$$

```
> poly_uv;
```

$$u^5\ v + 2.2\ u^4\ v^4 - 0.5\ u\ v^{11} + 0.1\ u\ v^{12} \quad (16)$$

Approximation theory

Approximation theory

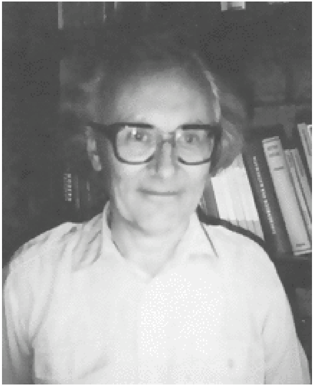


Figure: Henri Padé [1892] and Christian Pommerenke [1973]

Approximation theory

$$f_s = \sum_{i=1}^{n} \alpha_i \exp(\phi_i x_s), \qquad s = 0, 1, \ldots, 2n-1$$

$$f(z) = \sum_{j=0}^{\infty} f_j z^j, \qquad f_j = 0, \qquad j < 0$$

$$p(z) = \sum_{i=0}^{\ell} a_i z^i,$$

$$q(z) = \sum_{i=0}^{m} b_i z^i$$

$$\left(\sum_{j=0}^{\infty} f_j z^j\right) q(z) - p(z) = \sum_{i \geq \ell+m+1} c_i z^i$$

Approximation theory

$$\begin{cases} f_0 b_0 = a_0 \\ f_1 b_0 + f_0 b_1 = a_1 \\ \quad \vdots \\ f_\ell b_0 + \cdots + f_{\ell-m} b_m = a_\ell \end{cases}$$

$$b_0 = 1$$

$$\begin{cases} f_{\ell+1} b_0 + \cdots + f_{\ell-m+1} b_m = 0 \\ \quad \vdots \\ f_{\ell+m} b_0 + \cdots + f_\ell b_m = 0 \end{cases}$$

$$H_m^{(\ell+1-m)} \begin{pmatrix} b_m \\ \vdots \\ b_1 \end{pmatrix} = - \begin{pmatrix} f_{\ell+1} \\ \vdots \\ f_{\ell+m} \end{pmatrix}$$

$$[\ell/m](z) := p(z)/q(z)$$

Approximation theory

$$f_s = \sum_{i=1}^{n} \alpha_i \exp(\phi_i x_s) = \sum_{i=1}^{n} \alpha_i \Omega_i^s$$

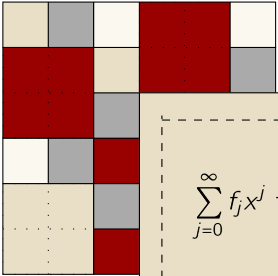
$$f(z) = \sum_{j=0}^{\infty} f_j z^j = \sum_{i=1}^{n} \frac{\alpha_i}{1 - z\Omega_i} = \text{Laplace transform of } \sum_{i=1}^{n} \alpha_i \exp(\phi_i x)$$

Approximation theory

$$[n-1/n](z) = p(z)/q(z)$$

$$\begin{aligned} q(z) &= \prod_{i=1}^{n}(1 - z\Omega_i) \\ &= z^n \frac{H_n^{(0)}(1/z)}{\left|H_n^{(0)}\right|} \\ &= b_0 z^n + b_1 z^{n-1} + \cdots + b_{n-1} z + 1 \end{aligned}$$

Approximation theory



$\rightarrow H_m^{(n-m)}$ regular, $m \geq n$

$\sum_{j=0}^{\infty} f_j x^j$ from rational function

$H_\nu^{(r)}$ singular, $\nu > n$, $r > n - \nu$

$\downarrow$

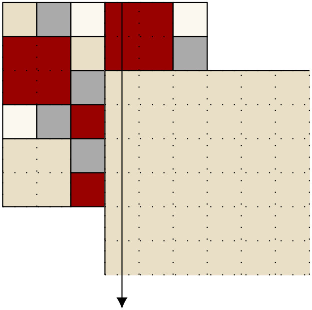
$H_n^{(\ell+1-n)}$ regular, $\ell + 1 \geq n$

Approximation theory

$f(z) + \varepsilon(z)$ meromorphic with poles in $0 \leq |z| < R$ of total multiplicity n [de Montessus de Ballore, 1905]

$$\Downarrow$$

$[\nu/n](z) \to f(z) + \varepsilon(z)$ uniformly on compact sets excluding poles, with poles of $f(z) + \varepsilon(z)$ attracting poles of $[\nu/n](z)$ according to their multiplicity



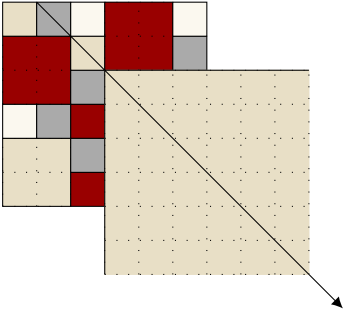
Approximation theory

$f(z)+\varepsilon(z)$ analytic except for a countable number of poles [Nuttall, 1970] and essential singularities [Pommerenke, 1973]

$$\Downarrow$$

$[\nu-1/\nu](z) \to f(z)+\varepsilon(z)$ in measure on compact sets, i.e.

$$\Lambda_2\left(\{z : |f(z)+\varepsilon(z)-[\nu-1/\nu](z)| \geq \tau\}\right) \to 0$$



Universiteit Antwerpen

Approximation theory

mathematical (noise free):

1. build $H_\nu^{(0)}, \quad \nu = 0, 1, 2, \dots$
2. $H_\nu^{(0)} = U \Sigma V^T$ singular value decomposition
3. $\Sigma = \begin{pmatrix} \sigma_1 & & \\ & \ddots & \\ & & \sigma_\nu \end{pmatrix}, \quad \sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_n > \sigma_{n+1} = \cdots = \sigma_\nu = 0$
4. find $\Omega_i, \phi_i, \alpha_i, i = 1, \dots, n$

Approximation theory

numerical (with noise):

1. take ν large enough so that noise is clearly separated from n
2. solve $H_\nu^{(1)} v_i = \lambda_i H_\nu^{(0)} v_i, \quad i = 1, \ldots, \nu, \quad \lambda_i = \Omega_i, \quad i = 1, \ldots, n$
3. find ϕ_i
4. solve $\sum_{i=1}^{n} \alpha_i \exp(\phi_i x_j) = f_j, \quad 0 \le j \le 2\nu - 1$

Example: noise

$$\phi_1 = 0, \qquad \alpha_1 = 1,$$

$$\phi_2 = -0.2 + 39.5\mathtt{i}, \qquad \alpha_2 = 2, \qquad x_s = s\frac{2\pi}{100},$$

$$\phi_3 = -0.5 + 40\mathtt{i}, \qquad \alpha_3 = 4, \qquad M = 100$$

$$\phi_4 = -1, \qquad \alpha_4 = 8,$$

$$\|\varepsilon(z)\|_\infty = 10^{-2}, \qquad \text{uniform random noise}$$

Example: noise

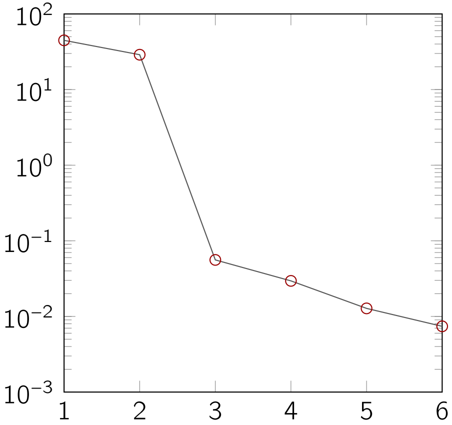


Figure: Singular values $H_\nu^{(0)}$ with $n = 4, \nu = 6$

Example: noise

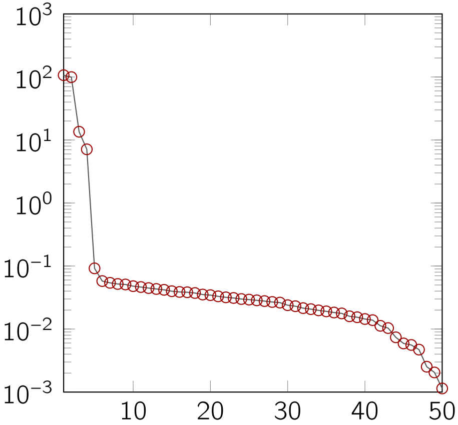


Figure: Singular values $H_\nu^{(0)}$ with $n = 4, \nu = 50$

Exercise: approximation

$$\phi_1 = 1.5\mathtt{i}, \qquad \alpha_1 = 10^{-3},$$

$$\phi_2 = 12.7\mathtt{i}, \qquad \alpha_2 = 2, \qquad x_s = s\frac{2\pi}{100},$$

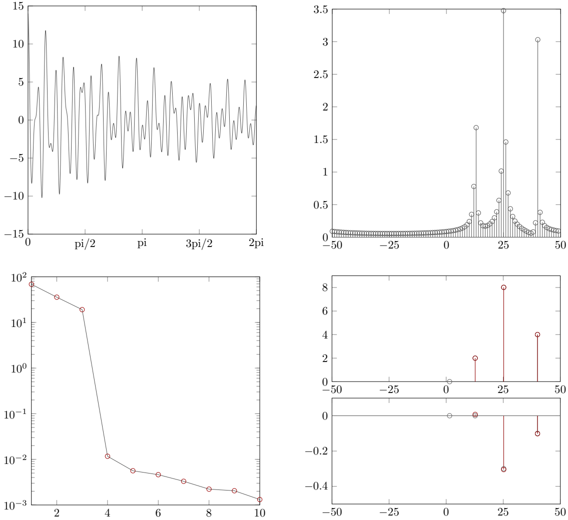
$$\phi_3 = -0.1 + 40\mathtt{i}, \qquad \alpha_3 = 4, \qquad M = 100$$

$$\phi_4 = -0.3 + 25.2\mathtt{i}, \qquad \alpha_4 = 8,$$

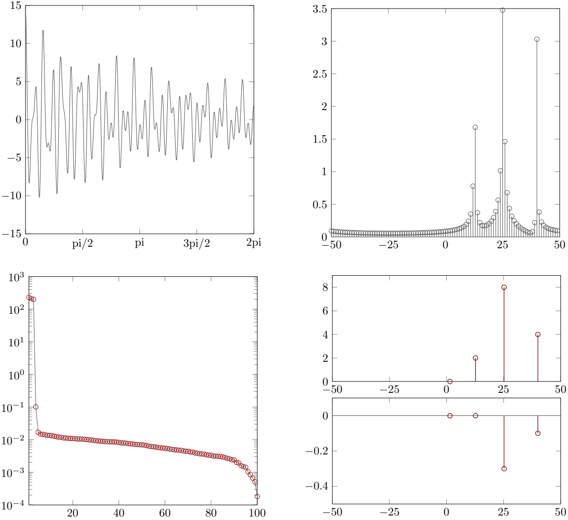
$\|\varepsilon(z)\|_\infty = 2 \times 10^{-3}$, uniform random noise

```
format long;

phi = [1.5*1i, 12.7*1i, -0.1+40*1i, -0.3+25.2*1i];
alpha = [10^(-3), 2, 4, 8];

eps = 2*10^(-3);

M = 100;

plot_signal

pause

plot_fft

pause

% synthesized input data with added noise
N = input('Enter the dimension for SVD: ');
% (10, 6, 3), (100, 100, 4)

randn('seed',0);

omega = 2*pi/M*(0:2*N-1);
f = syn_exp(alpha, phi, omega);

v = randn(size(f))+randn(size(f))*1i;
vv = v/norm(v,Inf);
f = f + eps*vv;

% form Hankel matrices H0 and H1 from y sequence
[H0,H1] = mat_ge(f);

plot_svd

pause

% reconstruct the parameters via generalized eigenvalues
n = input('Size of the model: ');
```

```
% compute the generalized eigenvalues and form the Vandermonde system
E = eig(H1(1:n,1:n),H0(1:n,1:n));
V = rot90(vander(E));

% amplitudes
A = V\f(1:n).';

% frequencies and damping factors
alpha_rec = A;
phi_rec = log(E)*M/(2*pi);

pause

% extract the non-zero terms
extract

% plot computed parameters
plot_recontructed_parameters
```



15
10
5
0
−5
−10
−15
0
pi/2
pi
3pi/2
2pi
3.5
3
2.5
2
1.5
1
0.5
0
−50
−25
0
25
50
10^{2}
10^{1}
10^{0}
10^{-1}
10^{-2}
10^{-3}
2
4
6
8
10
8
6
4
2
0
−0.2
−0.4


0
pi/2
pi
3pi/2
2pi

Applications

Applications

Exponential analysis in physical phenomena:

- power system transient detection
- motor fault diagnosis
- drug clearance / glucose tolerance
- magnetic resonance / infrared spectroscopy
- vibration analysis
- seismic data analysis
- music signal processing
- corrosion rate / crack initiation
- odour recognition with electronic nose
- typed keystroke recognition
- liquid (explosive) identification
- ...

Transients

Figure: Transient detection and characterization

Transients

short lived high frequency signal:

- speech processing
- turbulent flow
- power lines
- ...

Transients

- model with $\phi_i = 120\pi\mathtt{i}$,

$$\sum_{i=1}^{n} \alpha_i \cos(120\pi x + \gamma_i)\mathbf{1}_{[A_i, Z_i[}$$

- $n = 3, \alpha_i = 1, \gamma_{1,3} = -\pi/2, \gamma_2 = 3\pi/4$
- $[A_1, Z_1[= [0, 0.0308[$
  $[A_2, Z_2[= [0.0308, 0.0625[$
  $[A_3, Z_3[= [0.0625, 0.1058[$
- $M = 1200$
- uniformly distributed noise in $[-0.05, 0.05]$

Transients

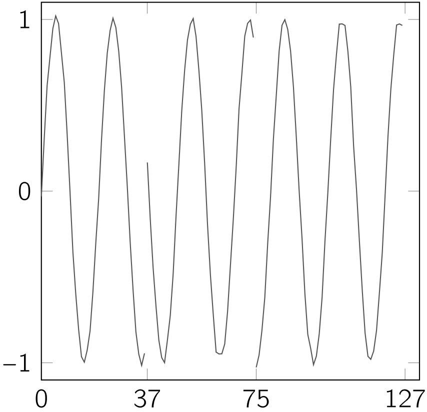


Figure: Given transient signal

Transients

- at each instance: 2 exponential terms
- characteristics of terms change
- inspect rank of

$$H_4^{(1)} = \begin{pmatrix} f_1 & f_2 & f_3 & f_4 \\ f_2 & f_3 & f_4 & f_5 \\ f_3 & f_4 & f_5 & f_6 \\ f_4 & f_5 & f_6 & f_7 \end{pmatrix}$$

$$[A_1, Z_1[= [0/M, 37/M[,$$
$$[A_2, Z_2[= [37/M, 75/M[,$$
$$[A_3, Z_3[= [75/M, 127/M[$$

Transients

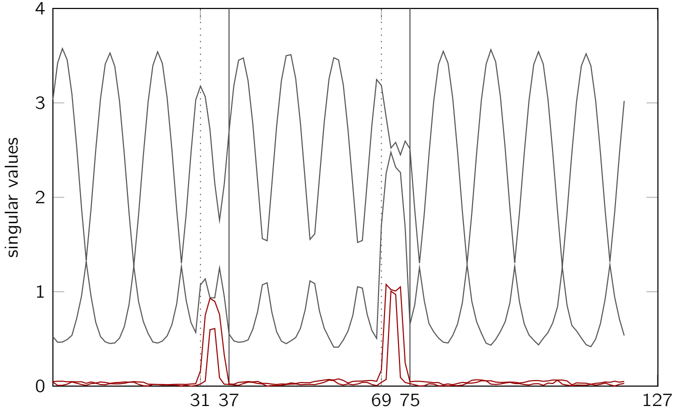


Figure: Numerical rank of $H_4^{(r)}$ evolving over time x

Audio signals

Figure: Reconstructing undersampled audio signals

Audio signals

song containing 29 notes of 0.25 seconds each:

- $M = 44100$ (Hz)
- 11025 samples per note, 319725 in total
- $16.35 \leq \phi_i \leq 4978.03, \; i = 1, \ldots, 100$
- complex exponential model

Audio signals

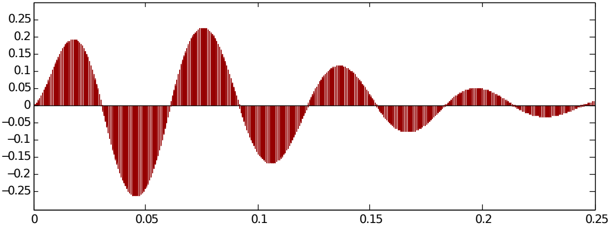


Figure: Sampled signal produced by 1 note

Audio signals

compressive sensing (optimisation, probabilistic)

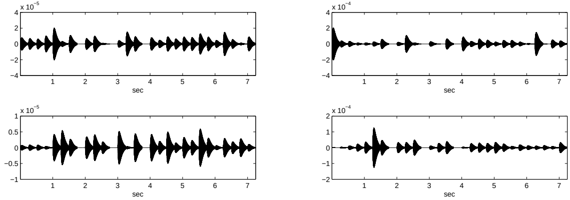


Figure: 4 runs with 1229 samples

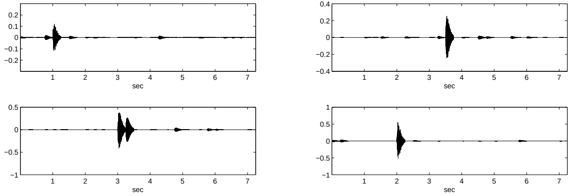


Figure: 4 runs with 456 samples

Audio signals

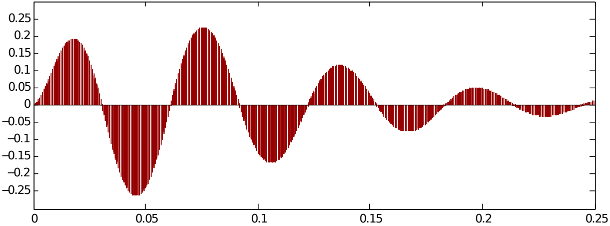


Figure: Full set of samples per note

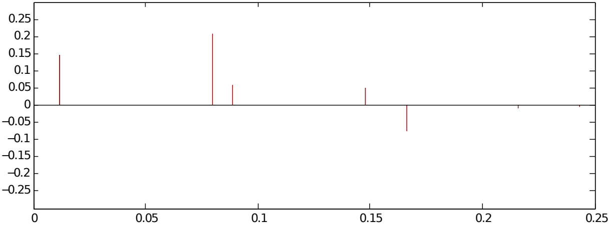


Figure: Sparse interpolation with 7 samples per note

MCSA

Figure: Preventive diagnosis of a broken rotor bar

MCSA

3-phase induction motors:

- consume 40 – 50% of all electricity in industrialized countries
- rotor made up metal bars
- stator current signal analysed
- broken bar(s) characterized by sideband frequencies
- difficult to diagnoze under low or no load

Figure: Stator and rotor

MCSA

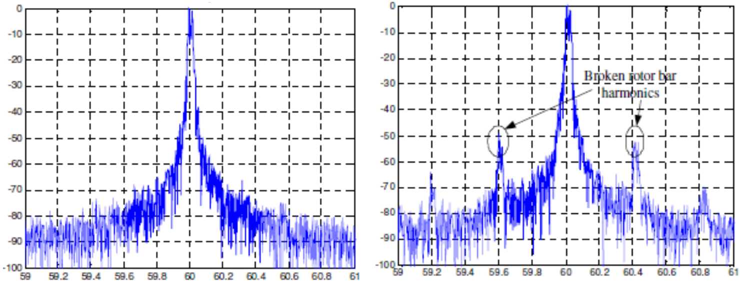


Figure: Stator current FFT spectra: healthy and with 1 broken bar

MCSA

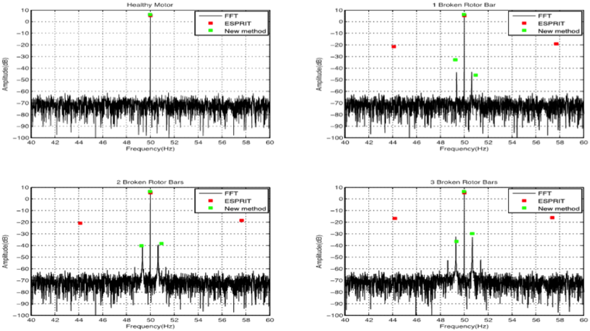


Figure: 10% load, 16dB noise, $\nu = 400$

Universiteit Antwerpen

Bio-electrical

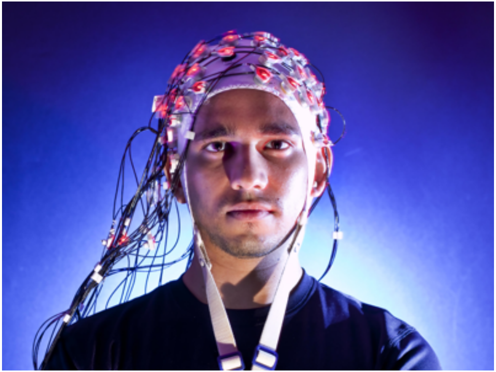


Figure: Sparse EEG approximation

Bio-electrical

Bio-electrical signals:

- electrical activity of cells and tissues
- clinical studies of health status
- ECG, EEG, EMG, EOG, ...
- sparse model ($n = 8$) is approximate

Bio-electrical

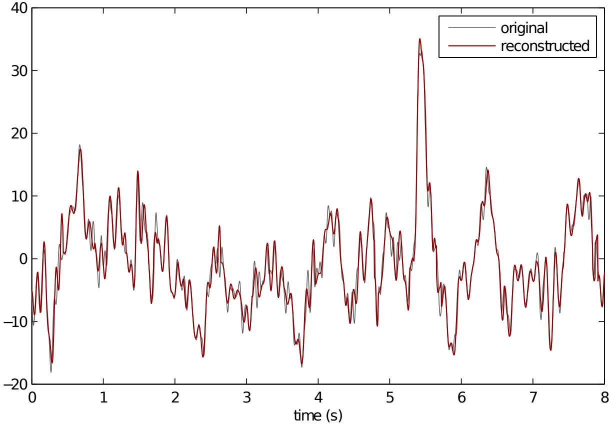


Figure: Reconstruction of 8 second [1 – 20] Hz bandpass filtered EEG

Bio-electrical

Figure: Sparse EOG approximation

Bio-electrical

Polysomnogram:

- 12 channels
- 22 wire attachments to patient
- heart rate, leg measurement, airflow (chest, abdomen), chin muscle, EEG, EOG, ...

Bio-electrical

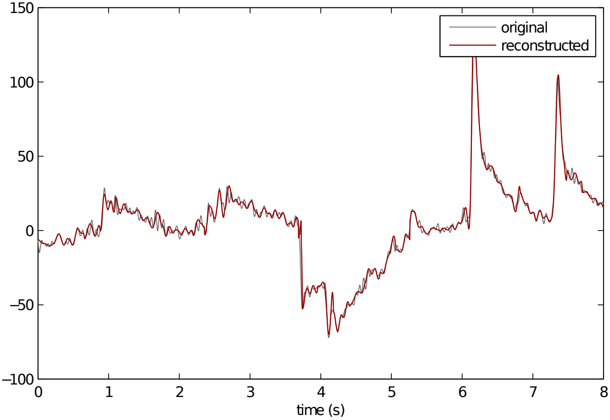


Figure: Reconstruction of 8 second EOG (CC = 99.2%)

Spectroscopy

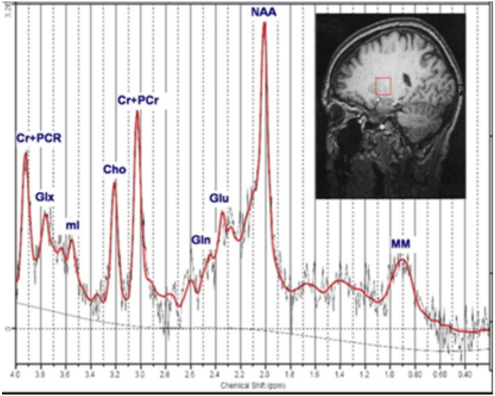


Figure: Spectral analysis of FID

Spectroscopy

Magnetic resonance spectroscopy:

- physical and chemical properties of molecules
- a.o. concentration of metabolites in the brain
- frequencies clustered → high frequence resolution
- free induction decay → time constraint
- Fourier methods need additional tools

Spectroscopy

$$\phi(x) = 5 \times 10^{-2} + 2e^{(-0.97+\mathrm{i}79.94\pi)x} + 4e^{(-1+\mathrm{i}80\pi)x} + 8e^{-1.1x} + \varepsilon(x)$$

$\|\varepsilon(x)\|_\infty = 10^{-3}$, circular Gaussian noise

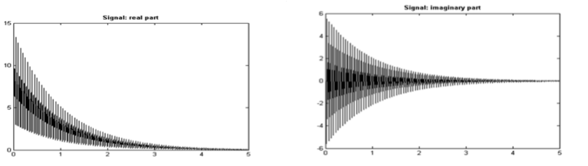


Figure: The real (left) and imaginary (right) part of $\phi(x)$

Universiteit Antwerpen

Spectroscopy

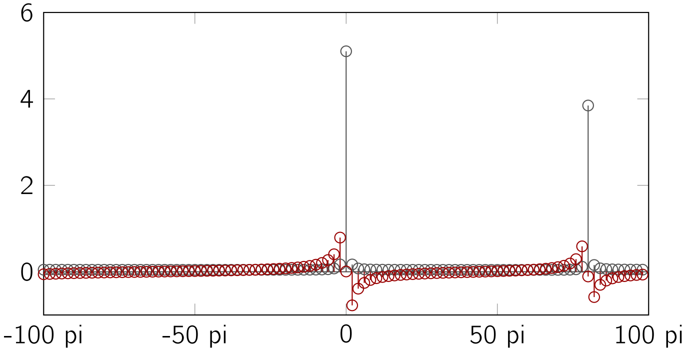


Figure: Real (black) and imaginary (red) parts of FFT

Spectroscopy

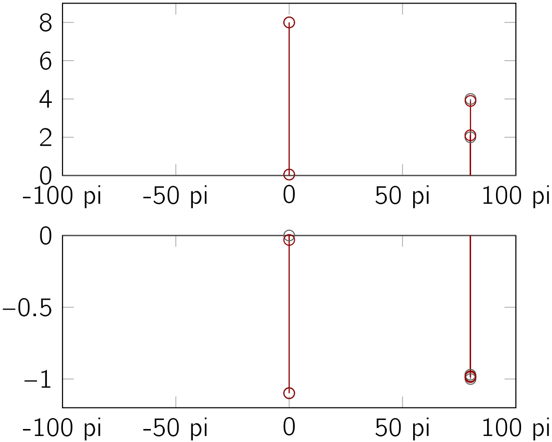


Figure: Amplitudes (top) and damping factors (bottom) of $\phi(x)$

References

References

M. Ben-Or and P. Tiwari. A deterministic algorithm for sparse multivariate polynomial interpolation. In *STOC '88: Proceedings of the twentieth annual ACM symposium on Theory of computing*, pages 301–309, New York, NY, USA, 1988. ACM. http://dx.doi.org/10.1145/62212.62241.

M. de Montessus de Ballore. Sur les fractions continues algébriques. *Rend. Circ. Mat. Palermo*, 19:185–257, 1905. http://dx.doi.org/10.1007/BF03014011.

R. de Prony. 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. *J. Ec. Poly.*, 1:24–76, 1795.

M. Giesbrecht, G. Labahn, and W. Lee. Symbolic-numeric sparse interpolation of multivariate polynomials. In *ISSAC'06*, 2006. http://dx.doi.org/10.1145/1145768.1145792. 9-12 July.

P. Henrici. *Applied and computational complex analysis I*. John Wiley & Sons, New York, 1974.

Y. Hua and T. K. Sarkar. Matrix pencil method for estimating parameters of exponentially damped/undamped sinusoids in noise. *IEEE Trans. Acoust. Speech Signal Process.*, 38: 814–824, 1990. http://dx.doi.org/10.1109/29.56027.

E. Kaltofen and W.-s. Lee. Early termination in sparse interpolation algorithms. *J. Symbolic Comput.*, 36(3-4):365–400, 2003. http://dx.doi.org/10.1016/S0747-7171(03)00088-9. International Symposium on Symbolic and Algebraic Computation (ISSAC'2002) (Lille).

J. L. Massey. Shift-register synthesis and BCH decoding. *IEEE Trans. Inform. Theory*, 15 (1):122–127, 1969. http://dx.doi.org/10.1109/TIT.1969.1054260.

J. Nuttall. The convergence of Padé approximants of meromorphic functions. *J. Math. Anal. Appl.*, 31:147–153, 1970. http://dx.doi.org/10.1016/0022-247X(70)90126-5.

H. Padé. *Sur la représentation approchée d'une fonction par des fractions rationnelles*. PhD thesis, Faculté des sciences de Paris, 1892.

C. Pommerenke. Padé approximants and convergence in capacity. *J. Math. Anal. Appl.*, 41: 775–780, 1973. http://dx.doi.org/10.1016/0022-247X(73)90248-5.