Polynomial equations, structured matrices and approximate computations

B. Mourrain

INRIA, SAGA
2004 route des Lucioles, BP 93
06902 Sophia-Antipolis Cedex

Email: mourrain@sophia.inria.fr
http://www.inria.fr/saga/mourrain/

Towards certified and efficient solvers

1. Analytic methods: *Solve from the value of the function and its variation with respect to the input parameters.*

Proposition: $\{z \in \mathbb{C}; |p(z)| < \epsilon \|\mathbf{z}\|\} = \{z \in \mathbb{C}; \exists \tilde{p} \text{ with } \tilde{\mathrm{p}}(\mathrm{z}) = 0; \|\mathrm{p} - \tilde{\mathrm{p}}\| < \epsilon\}$.

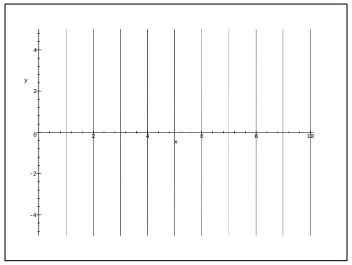
- Newton type methods: No global guaranty on the convergence. Local methods.

  Conjecture: Newton iteration always converges for the system of relations between roots $x_1, \ldots, x_d$ and coefficients a_i.

  Local guaranty: Kantorovitch, α-theory, ...

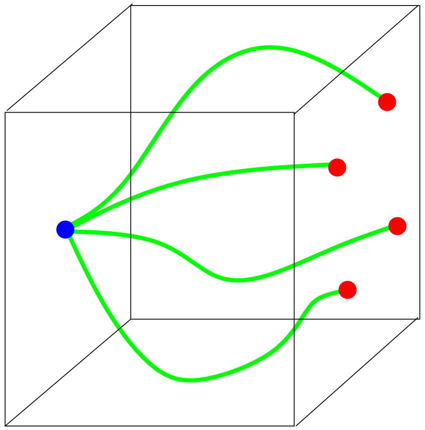
  Theorem: If $\|\nabla \mathbf{f}(x_0)^{-1}\| < a$, $\|\nabla \mathbf{f}(x_0)^{-1} \mathbf{f}(x_0)\| < b$, $\sum_j |\frac{\partial \mathrm{f}(z)}{\partial x_i \partial x_j}| < c$ in B, $2\, n\, a\, b\, c < 1$, then Newton scheme convergences in B.
- Exclusion methods, interval analysis, topological degree: uses information on the value of the function (sign), on the way it is computed, ...

```
> plot(product((x-i),i=1..10),x=0..10.1,y=-5..5);
```



Solving by bisection, slow when the dimension increase.

– Homotopy: deformation from a known situation to the one we want to solve. Genericity is required. The paths have to be certified carefully. Possibility to exploit continuity.



2. Algebraic methods: *Solve from the constraints satisfied by the "unknowns".*

- Gröbner basis like methods: Do exact computation. Require extended arithmetic. Pointwise approach. No information on the neighborhood.
- Matrix based methods: Take into account the continuity of the problem. Combine symbolic and numeric methods. Applicable with approximate computations. Requires a pre-analysis of the geometry of the problem.

Linear algebra for solving non-linear equations

- **Certification** is obtained by a control of the error through *numerical linear algebra* tools.
- **Efficiency** is achieved by exploiting the *structure* of the objects.

Applications:

- Compute approximations of the roots and estimate the error.
- Solve overdetermined system with noise.
- Controlled iterative methods.
- Allow preprocessing steps in order to generate a solver tuned to the problem.

Algebraic approach

$\Rightarrow$ Polynomial equations: $f_1, \ldots, f_m \in \mathbb{K}[x_1, \ldots, x_n] = \mathbb{K}[\mathbf{x}]$.

$\Leftarrow$ The roots $\mathcal{Z}(f_1, \ldots, f_n) = \{\zeta \in \overline{\mathbb{K}}^n; f_i(\zeta) = 0, i = 1, \ldots, m\}$.

Use computations modulo the constraints to deduce information on the geometry of the roots.

Study of

- the ideal I generated by $f_1, \ldots, f_m$,
- the quotient algebra $\mathcal{A} = \mathbb{C}[\mathbf{x}]/I$,
- its dual $\widehat{\mathcal{A}}$.

Theorem: The quotient algebra $\mathcal{A}$ is of finite dimension D over $\mathbb{K}$ iff $\mathcal{Z}(I)$ is finite.

Multiplication maps in the quotient ring $\mathcal{A} = \mathbb{K}[\mathbf{x}]/I$

We assume $\mathcal{A}$ of finite dimension D over $\mathbb{K}$.

$$\begin{array}{rcl} M_a : \mathcal{A} & \to & \mathcal{A} \\ u & \mapsto & a\,u \end{array} \qquad \begin{array}{rcl} M_a^{\mathsf{t}} : \widehat{\mathcal{A}} & \to & \widehat{\mathcal{A}} \\ \Lambda & \mapsto & a \cdot \Lambda : u \mapsto \Lambda(au) \end{array}$$

Theorem:

1. ***The eigenvalues of M_a are $\{a(\zeta_1), \ldots, a(\zeta_d)\}$.***
2. ***The eigenvectors of all $(M_a^{\mathsf{t}})_{a\in\mathcal{A}}$ are (up to a scalar) $\mathbf{1}_{\zeta_i} : p \mapsto p(\zeta_i)$.***

Theorem: *In a basis of $\mathcal{A}$, all the matrices M_a ($a \in \mathcal{A}$) are of the form*

$$\mathtt{M}_a = \begin{bmatrix} \mathtt{N}_a^1 & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \mathtt{N}_a^d \end{bmatrix} \text{ avec } \mathtt{N}_a^i = \begin{bmatrix} a(\zeta_i) & & \star \\ & \ddots & \\ \mathbf{0} & & a(\zeta_i) \end{bmatrix}$$

Compute the simple roots of a polynomial system $f_1 = \cdots = f_m = 0$.

1. Computed the transpose of the matrix of multiplication by $a \in \mathcal{A}$ in a basis of the form $(1, x_1, \ldots, x_n, \ldots)$.
2. Compute its eigenvectors $\wedge_i = [v_{i,1}, v_{i,x_1}, \ldots, v_{i,x_n}, \ldots]$ for $i = 1, \ldots, d$.
3. For $i = 1, \ldots, d$, compute and output

$$\zeta_i = (\frac{v_{i,x_1}}{v_{i,1}}, \ldots, \frac{v_{i,x_n}}{v_{i,1}}).$$

Compute the roots by simultaneous triangulation.

1. Compute a Schur triangular decomposition $\mathtt{T}_{x_1} = \mathtt{P}\mathtt{M}_{x_1}\mathtt{P}^{-1}$.
2. Compute $\mathtt{T}_{x_i} = \mathtt{P}\,\mathtt{M}_{x_i}\mathtt{P}^{-1}$, $i = 2, \ldots, n$ which are also triangular.

The roots are the diagonal vectors $\mathbf{t}_i = (t^1_{i,i}, \ldots, t^n_{i,i})$ of the matrices $\mathtt{T}_{x_i}$.

Matrix of resultant in one variable

Let $f_0 = c_{0,0} + \cdots + c_{0,d_0}\, x^{d_0}$, $f_1 = c_{1,0} + \ldots + c_{1,d_1}\, x^{d_1}$ (with $d_0 \leq d_1$).

Sylvester (1840):

$$
\begin{array}{c}
\overbrace{\begin{array}{cccccc} f_0 & \cdots & x^{d_1-1} f_0 & f_1 & \cdots & x^{d_0-1} f_1 \end{array}}^{d_0+d_1} \\
\left[\begin{array}{ccc|ccc}
c_{0,0} & & \mathbf{0} & c_{1,0} & & \mathbf{0} \\
\vdots & \ddots & & \vdots & \ddots & \\
\vdots & & c_{0,0} & \vdots & & c_{1,0} \\
c_{0,d_0} & & \vdots & c_{1,d_1} & & \vdots \\
& \ddots & \vdots & & \ddots & \vdots \\
\mathbf{0} & & c_{0,d_0} & \mathbf{0} & & c_{1,d_1}
\end{array}\right]
\end{array}
\qquad
\left.\begin{array}{l}
1 \\ x \\ . \\ x^{d_1-1} \\ . \\ x^{d_0+d_1-1}
\end{array}\right\} d_0 + d_1
$$

Bézout (1779): $\mathcal{B}_{f_0,f_1}(x,y) := \frac{f_1(x)\, f_0(y) - f_1(y)\, f_0(x)}{y-x} = \sum_{i=0}^{d_1-1} \sum_{j=0}^{d_1-1} b_{i,j}\, x^i\, y^j$.
Its matrix is $\mathtt{B}_{f_0,f_1} = (b_{i,j})_{0 \leq i,i \leq d_1}$.

Generalization to several variables

Condition on $(c_{i,j})$ such that the system has a solution in a projective variety X:

$$\left\{ \begin{array}{l} f_0 = \sum_{\alpha \in A_0} c_{0,\alpha}\, x_0^{\alpha_0} \cdots x_n^{\alpha_n}, \\ \vdots \\ f_n = \sum_{\alpha \in A_n} c_{n,\alpha}\, x_0^{\alpha_0} \cdots x_n^{\alpha_n}, \end{array} \right.$$

Matrix of an application of the form:

$$\begin{array}{rcl} \mathcal{S} : \langle \mathbf{x}^{E_0} \rangle \times \cdots \langle \mathbf{x}^{E_n} \rangle & \to & \langle \mathbf{x}^{F} \rangle \\ (q_0, \ldots, q_n) & \mapsto & \sum_{i=0}^{n} q_i\, f_i. \end{array}$$

Macaulay (1902): $A_i = \{\alpha \; ; \; |\alpha| \leq d_i\}$, $\nu = \sum_{i=0}^{n} (d_i - 1) + 1$, $F = \{\beta \; ; \; |\beta| \leq \nu\}$, $E_n = \{\ldots\}$, $\ldots$, $E_0 = \{\alpha \; ; \; 0 \leq \alpha_i \leq d_i - 1\}$

Toric (1975): A_i polytope (support). $F = A_0^{\delta} \oplus \cdots \oplus A_n^{\delta}$, $E_i = \oplus_{j \neq i} A_j^{\delta}$.

```
> read multires;
> f1 := 13*x[1]^2+8*x[1]*x[2]+4*x[2]^2-8*x[1]-8*x[2]+2;
> f2 := x[1]^2+x[1]*x[2]-x[1]-1/6;
> S := mresultant([u[0]+u[1]*x[1]+u[2]*x[2],f1,f2],[x[1],x[2]]);
```

$$S := \left[\begin{array}{cccc|ccc|ccc} u_0 & 0 & 0 & 0 & 2 & 0 & 0 & -\frac{1}{6} & 0 & 0 \\ u_2 & u_0 & 0 & 0 & -8 & 2 & 0 & 0 & -\frac{1}{6} & 0 \\ u_1 & 0 & u_0 & 0 & -8 & 0 & 2 & -1 & 0 & -\frac{1}{6} \\ 0 & u_1 & u_2 & u_0 & 8 & -8 & -8 & 1 & -1 & 0 \\ 0 & 0 & 0 & u_2 & 0 & 8 & 4 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 13 & 0 & 0 & 1 \\ 0 & 0 & 0 & u_1 & 0 & 13 & 8 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 4 & 0 & 0 & 0 & 0 \\ 0 & 0 & u_1 & 0 & 13 & 0 & -8 & 1 & 0 & -1 \\ 0 & u_2 & 0 & 0 & 4 & -8 & 0 & 0 & 0 & 0 \end{array}\right]$$

```
> det(S);
```

$$\frac{1}{36}\left(u_0+\frac{1}{3}u_1+\frac{7}{6}u_2\right)^2\left(u_0-\frac{1}{3}u_1+\frac{5}{6}u_2\right)^2$$

Multivariate Bezoutian

$$\Theta(f_0, f_1, \ldots, f_n) = \begin{vmatrix} f_0(X_{(0)}) & \theta_1(f_0) & \cdots & \theta_n(f_0) \\ \vdots & \vdots & & \vdots \\ f_n(X_{(0)}) & \theta_1(f_n) & \cdots & \theta_n(f_n) \end{vmatrix} = \sum_{\alpha \in E, \beta \in F} b_{\alpha,\beta}\, \mathbf{x}^{\alpha}\, \mathbf{y}^{\beta},$$

- $X_{(0)} = (x_1, \ldots, x_n)$, $X_{(1)} = (y_1, x_2 \ldots, x_n), \ldots,$ $X_{(n)} = (y_1, \ldots, y_n)$,
- $\theta_i(P) = \frac{P(X_{(i)}) - P(X_{(i-1)})}{y_i - x_i}$.

Its matrix $B_{f_0,\ldots,f_n} = [b_{\alpha,\beta}]_{\alpha \in E, \beta \in F}$ is the matrix associated to the map

$$\begin{array}{ccl} \widehat{R} & \to & R \\ \Lambda & \mapsto & \Theta(\mathbf{f})(\Lambda) = \sum_{\alpha \in E, \beta \in F} b_{\alpha,\beta}\, \mathbf{x}^{\alpha}\, \Lambda(\mathbf{y}^{\beta}), \end{array}$$

```
> read multires;
> Theta([u[0]+u[1]*x[1]+u[2]*x[2],f1,f2],[x[1],x[2]]);
```

$$-4\,u_0 y_1 y_2 + 5\,u_0 {y_1}^2 + \left(4\,u_0 - \frac{2}{3}\,u_1\right) y_2 + \left(8\,u_0 - \frac{10}{3}\,u_1 + \frac{25}{6}\,u_2\right) y_1 + \left(\frac{4}{3}\,u_1 - 8\,u_0 - \frac{10}{3}\,u_2\right)$$
$$+ \left(-4\,u_1 y_1 y_2 + 5\,u_1 {y_1}^2 - 4\,u_0 y_2 + (8\,u_1 + 5\,u_2 + 5\,u_0)\,y_1 + \left(8\,u_0 + \tfrac{25}{6}\,u_2\right)\right) x_1$$
$$+ \left(-4\,u_2 y_1 y_2 + 5\,u_2 {y_1}^2 + (4\,u_2 - 4\,u_0)\,y_2 + (8\,u_1 - 4\,u_0)\,y_1 + \left(\tfrac{10}{3}\,u_2 + 12\,u_0 - \tfrac{2}{3}\,u_1\right)\right) x_2$$
$$+ (-4\,u_2 y_2 - 4\,u_1 y_1 - 4\,u_0)\,x_1 x_2 + (-4\,u_2 y_2 - 4\,u_1 y_1 - 4\,u_0)\,{x_2}^2$$

```
> mbezout([u[0]+u[1]*x[1]+u[2]*x[2],f1,f2],[x[1],x[2]]);
```

$$\begin{bmatrix} -4\,u_0 & 5\,u_0 & 4\,u_0 - \frac{2}{3}\,u_1 & 8\,u_0 - \frac{10}{3}\,u_1 + \frac{25}{6}\,u_2 & 4/3\,u_1 - 8\,u_0 - \frac{10}{3}\,u_2 \\ -4\,u_1 & 5\,u_1 & -4\,u_0 & 8\,u_1 + 5\,u_2 + 5\,u_0 & 8\,u_0 + \frac{25}{6}\,u_2 \\ -4\,u_2 & 5\,u_2 & -4\,u_0 + 4\,u_2 & 8\,u_1 - 4\,u_0 & -\frac{2}{3}\,u_1 + \frac{10}{3}\,u_2 + 12\,u_0 \\ 0 & 0 & -4\,u_2 & -4\,u_1 & -4\,u_0 \\ 0 & 0 & -4\,u_2 & -4\,u_1 & -4\,u_0 \end{bmatrix}$$

```
> factor(det(submatrix(",1..4,2..5)));
```

$$\frac{5}{11664}\left(u_0 + \frac{1}{3}\,u_1 + \frac{7}{6}\,u_2\right)^2 \left(u_0 - \frac{1}{3}\,u_1 + \frac{5}{6}\,u_2\right)^2$$

Solving $f_1 = \cdots = f_n = 0$ by eigencomputation

1. Construct the resultant matrix of $f_0, f_1, \ldots, f_n$:

$$\mathtt{S} = \begin{array}{cc} & \begin{array}{cc} E_0 & E_1 \cdots E_n \end{array} \\ \begin{array}{c} E_0 \\ \\ F' \end{array} & \left[\begin{array}{c|c} \mathtt{A} & \mathtt{B} \\ \hline \\ \mathtt{C} & \mathtt{D} \\ \\ \end{array}\right] \end{array}.$$

2. Compute the Matrix of multiplication by f_0 modulo $f_1, \ldots, f_n$:

$$\mathtt{M}_{f_0} = \mathtt{A} - \mathtt{B}\,\mathtt{D}^{-1}\mathtt{C}.$$

3. Compute its eigenvectors/values and deduce the roots of $f_1 = \cdots = f_n = 0$.

Solving $f_1 = \cdots = f_n = 0$ by hiding a variable

1. Construct the resultant matrix $\mathtt{S}(x_n)$ of $f_1, \ldots, f_n$ as polynomials in $x_1, \ldots, x_{n-1}$ with coefficients in $\mathbb{K}[x_n]$.
2. Solve $\mathtt{v}\,\mathtt{S}(x_n) = 0$.
   - Either by solving $\det(S(x_n)) = 0$ and by deducing the corresponding v.
   - or by reducing it to an eigenproblem:

$$\mathtt{v}\,(S_d\,x_n^d + S_{d-1}x_n^{d-1} + \cdots + S_0) = 0 \Leftrightarrow$$

$$\mathtt{v}\left(\begin{bmatrix} 0 & \cdots & 0 & -S_0 S_d^{-1} \\ \mathbb{I} & \ddots & \vdots & \vdots \\ \vdots & \ddots & 0 & -S_{d-2} S_d^{-1} \\ 0 & \cdots & \mathbb{I} & -S_{d-1} S_d^{-1} \end{bmatrix} - x_n \mathbb{I}_{ds}\right) = 0,$$

3. Deduce the other coordinates of the roots from $\mathtt{v}$.

Other applications

– Solving overdetermined systems.
– Elimination of variables, projection on subspaces.
– From parameterized to implicit equations.
– Intersection of parameterized curves, surfaces.
– Offset curves, surfaces.
– Univariate representation of the isolated points.
– Geometrical decomposition of varieties.
– Certification of geometric predicates.
– . . .

$\boxed{\text{Multiplication and Toeplitz matrices}}$

- A polynomial $t(x) = t_0 + t_1\, x + \cdots + t_{2n}\, x^{2n} \in \mathbb{K}[x]$.
- Multiplication by t: $\mathcal{T}_t : \mathbb{K}[x] \to \mathbb{K}[x] \quad q \mapsto q\, t$
- Its matrix:

$$\begin{array}{c} 1 \\ \vdots \\ x^n \\ \vdots \\ x^{2n} \\ \vdots \\ \end{array} \left[\begin{array}{cccc} t_0 & & 0 & \cdot \\ \vdots & \ddots & & \cdot \\ \hline t_n & & t_0 & \cdot \\ \vdots & \ddots & \vdots & \cdot \\ t_{2n} & & t_n & \cdot \\ \hline & \ddots & \vdots & \cdot \\ 0 & & t_{2n} & \cdot \end{array}\right] \begin{array}{l} \\ \left.\begin{array}{l} \\ \\ \\ \end{array}\right\} T \\ \\ \end{array} \tag{1}$$

- The product $T\,\mathrm{v}$ is obtained by computing $t(x)\, v(x)$ (où $v(x) = v_0 + \cdots + v_n x^n$).
- Arithmetic complexity: $\mathcal{O}(n \log(n) \underline{\log\log(n)}) = \mathcal{O}(n \log(n) *)$ by FFT.

Duality et Hankel matrices

- The dual of $\mathbb{K}[x]$: $\widehat{\mathbb{K}[x]} = \{\Lambda : \mathbb{K}[x] \to \mathbb{K} \text{ linear}\}$, representation by series:
$$\Lambda = \sum_{i=0}^{\infty} \Lambda(x^i)\, \mathbf{d}^i \in \mathbb{K}[[\mathbf{d}]]$$
with $x \cdot \Lambda : p \to \Lambda(x\,p) = \mathbf{d}^{-1}\left(\sum_{i=1}^{\infty} \Lambda(x^i)\mathbf{d}^i\right).$
- Multiplication by $\Lambda = h_0 + h_1\,\mathbf{d} + \cdots$: $\mathcal{H}_\Lambda : \mathbb{K}[x] \to \widehat{\mathbb{K}[x]}$; $q \mapsto q \cdot \Lambda$
- Its matrix:

$$\begin{array}{c} 1 \\ \vdots \\ \mathbf{d}^n \\ \vdots \\ \mathbf{d}^{2n} \\ \\ \end{array} \left[\begin{array}{cccc} h_0 & & h_n & \cdot \\ \vdots & \iddots & \vdots & \cdot \\ h_n & & h_{2n} & \cdot \\ \hline \vdots & \iddots & \cdot & \cdot \\ h_{2n} & & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \end{array}\right] \quad \left.\begin{array}{c} \\ \\ \\ \end{array}\right\} H \qquad (2)$$

- The product $H\,\mathrm{v}$ is deduced from the coefficients of $h(\mathbf{d})\,v(\mathbf{d}^{-1})$.
- Arithmetic complexity: $\mathcal{O}(n\log(n)*)$ by FFT.

Other structures

1. **Bezoutian matrices**. Consider the bivariate polynomial

$$\Theta(x, y) = \frac{p(x)q(y) - p(y)q(x)}{x - y} = \sum_{0 \le i,j \le n-1} b_{i,j}\, x^i\, y^j.$$

and the associated operator:

$$\begin{array}{rcl} \mathcal{B}_{p,q} : \widehat{\mathbb{K}[x]} & \to & \mathbb{K}[x] \\ \Lambda & \mapsto & \displaystyle\sum_{0 \le i,j \le n-1} b_{i,j}\, x^i\, \Lambda(y^j) \end{array}$$

Its matrix is $B_{p,q} := [b_{i,j}]_{0 \le i,j \le n-1}$.
– Multiplication by a Vector: $\mathcal{O}(n \log(n) *)$.
– Resolution of a linear system $\mathcal{O}(n \log^2(n) *)$. $B_{1,p}^{-1} = H_{\tau p}$.

2. Compagnon matrix of $p = x^n + a_{n-1}x^{n-1} + \cdots + a_0$:

$$\begin{array}{rcl} M_x : \mathbb{K}[x]/(p) & \to & \mathbb{K}[x]/(p) \\ \overline{q} & \mapsto & \overline{x\,q} \end{array}$$

Its matrix:

$$\begin{bmatrix} & & \mathbf{0} & -a_0 \\ 1 & & & \vdots \\ & \ddots & & \vdots \\ \mathbf{0} & & 1 & -a_{n-1} \end{bmatrix}$$

- Multiplication of M_r by a vector: $\mathcal{O}(n\log(n)*)$.
- Resolution of a linear system $\mathcal{O}(n\log^2(n)*)$. We have $M_r^{-1} = M_{r^{-1}}$.

3. **Vandermonde matrices**.

$$\mathbf{V}(\alpha, \zeta) = (\zeta_i^{\alpha_j})_{i,j}$$

Change from the dual of the monomial basis to the basis of evaluation at the points ζ_i.
Transposed of the change from monomial to Lagrange basis.
Multiplication and resolution of linear system in $\mathcal{O}(n \log^2(n)*)$.

4. **Cauchy, Loewner matrices**.

$$L = \left(\frac{u_i - v_j}{s_i - t_j}\right)_{i,j}$$

Multiplication and resolution of linear system in $\mathcal{O}(n \log^2(n)*)$.

5. **Circulant matrices**.

$$\begin{cases} i < j & C_{i,j} = c_{i-j \ mod \ n} \\ i \geq j & C_{i,j} = f\, c_{i-j \ mod \ n} \end{cases}$$

Multiplication and resolution of linear system in $\mathcal{O}(n \log(n)*)$.

6. **Toeplitz-like, Hankel-like,... matrices**.
Characterized by the displacement operators.

7. **Schur complement, inverse of structured matrices**.

Generalization to the multivariate case

- The rows and columns of the matrices $M = (m_{\alpha,\beta})$ are indexed by monomials $(\mathbf{x}^{\alpha})_{\alpha \in E}$, $(\mathbf{y}^{\beta})_{\beta \in F}$.
- M is an **quasi-Toeplitz** matrix iff the entries $m_{\alpha,\beta} = t_{\alpha-\beta}$ depend only on $\alpha - \beta$. The associated polynomial is $T = \sum_{\gamma \in E-F} t_{\gamma} \mathbf{x}^{\gamma}$.
- M is an **quasi-Hankel** matrix iff the entries $m_{\alpha,\beta} = h_{\alpha+\beta}$ depend only on $\alpha + \beta$. The associated polynomial is $H = \sum_{\gamma \in E+F} h_{\gamma} \mathbf{x}^{-\gamma}$.

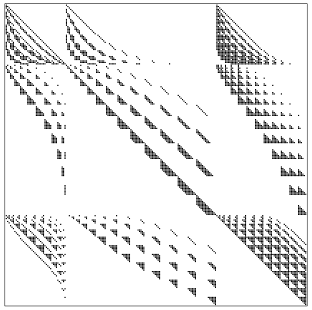
Definition: $S = \mathbb{K}[x_1^{\pm 1}, \ldots, x_n^{\pm 1}]$. $\mathcal{P}_E : S \rightarrow S$: projection on the subspace $\langle \mathbf{x}^{\alpha} \rangle_{\alpha \in E}$.

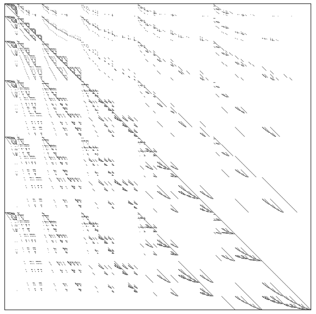
Proposition:

- If $\mathcal{T} : S \rightarrow S$ is Quasi-Toeplitz, $\mathcal{T} = \mathcal{P}_E \, \mathcal{M}_T \, \mathcal{P}_F$.
- If $\mathcal{H} : S \rightarrow S$ is Quasi-Hankel, $\mathcal{H} = \mathcal{P}_{-E} \, \mathcal{M}_H \, \mathcal{P}_F$.

The structure of resultant matrices

$$F \left\{ \begin{array}{c} \mathbf{x}^{\alpha_1} \\ \vdots \\ \mathbf{x}^{\alpha_N} \end{array} \right. \quad \begin{array}{c} \begin{array}{cccc} \overbrace{\hspace{3cm}}^{E_0} & \overbrace{\hspace{3cm}}^{E_1} & \hspace{2cm} & \overbrace{\hspace{3cm}}^{E_n} \end{array} \\ \left[\begin{array}{cc|cc|c|cc} \vdots & & \vdots & & & \vdots & \\ \mathbf{x}^{\beta_{0,1}} f_0 & \cdots & \mathbf{x}^{\beta_{1,1}} f_1 & \cdots & \cdots\cdots\cdots & \mathbf{x}^{\beta_{n,1}} f_n & \cdots \\ \vdots & & \vdots & & & \vdots & \end{array} \right] \end{array} .$$





Real roots and QuasiHankel matrices

Let $f_i \in \mathbb{R}[\mathbf{x}]$, $i = 1, \ldots, m$ and $\mathcal{V} = \{\zeta \in \mathbb{C}^n, f_1(\zeta) = \cdots = f_m(\zeta) = 0\}$.

Consider the *linear form* $Tr : a \mapsto trace(M_a) = \sum_{\zeta \in \mathcal{V}} \mu_\zeta \, a(\zeta)$ and the associated *quadratic form*:

$$Q_h : (a, b) \mapsto Tr(h\, a\, b).$$

Its matrix in the basis $(\mathbf{x}^\alpha)_{\alpha \in E}$ is $\left(Tr(h\, \mathbf{x}^{\alpha+\beta})\right)_{\alpha,\beta \in E}$.

Theorem: (Hermite) Let $h \in \mathbb{R}[\mathbf{x}]$.

1. The rank of Q_h is the number of (complex) roots $\zeta \in \mathcal{V}$ st. $h(\zeta) \neq 0$.
2. The signature of Q_h is
$\#\{\ \zeta$ real root such that $h(\zeta) > 0\ \} - \#\{\ \zeta$ real root such that $h(\zeta) < 0\ \}$.

Relations between these structured matrices

Modulo $f_1, \ldots, f_n$:

- $B_a = B_1\, M_a^t = M_a B_1$.
- $H_{a\cdot\Lambda} = H_\Lambda\, M_a = M_a^t H_\Lambda$.
- If $v\, w = 1$ in B, then $B_v H_{w\cdot\tau} = Id$.
- $rk(B_a) = rk(H_{a\cdot\tau}) = rk(M_a) = dim_{\mathbb{K}}(0 : a)$.
- $B_a = B_1\, H_{a\cdot\tau}\, B_1$,
- $H_{a\cdot\tau} = H_\tau\, B_a\, H_\tau$,
- $B_a = V_E(\mathcal{Z})^{-1} \mathrm{diag}(a(\zeta_1)\, J(\zeta_1), \ldots, a(\zeta_D)\, J(\zeta_D)) V_E(\mathcal{Z})^{-t}$ (simple roots).
- $M_a = V_E^{-1}(\mathcal{Z}) \mathrm{diag}(a(\zeta_1), \ldots, a(\zeta_d)) V_E(\mathcal{Z})$ (simple roots).

Controlled iterative methods

We look at the elements in $\mathcal{A}$ as multiplicative operators. Induction in the quotient $\mathcal{A}$.

1. $u_{n+1} = f_0 u_n$ (Bernoulli).
2. $u_{n+1} \equiv u_n^2$ (Sebastio e Sylva).
3. $u_{n+1} \equiv \frac{1}{2}(u_n \pm u_n^{-1})$ (Joukovski).

Convergence to one (or a subsum) of the "eigenvectors" or **idempotents** associated with the roots.
Convergence from the beginning and quadratic for 2, 3.

Applications:

- ☐ Compute the root(s) which maximize(s), minimize(s) $|h|$.
- ☐ Count/compute the roots which are almost real.
- ☐ Count/compute the roots in a box.

Implicit inverse power method

Compute the root which minimizes $|f_0|$

1. Compute the $\mathtt{S}$ resultant matrix of $f_0, \ldots, f_n$.
2. Choose $\mathbf{w}_0$ at random. Solve $\mathtt{S}^t \begin{bmatrix} \mathbf{w}_{n+1} \\ \mathbf{v}_{n+1} \end{bmatrix} = \begin{bmatrix} \mathbf{w}_n \\ \mathbf{0} \end{bmatrix}$ $(\mathbf{w}_{n+1} = \frac{1}{\|\mathbf{w}_n\|}(\mathtt{M}^{\mathbf{t}}_{f_0})^{-1}\mathbf{w}_n)$ until $\|\mathbf{w}_{n+1} - \mathbf{w}_n\| < \epsilon$. The root is $\zeta = (\frac{w x_1}{w_1}, \ldots, \frac{w x_n}{w_1})$.

(ALP)	N	S	D	n	k	T
s44	36	138	16	2	7	0.050s
s442	165	821	32	3	6	0.151s
s4422	715	3704	64	4	8	1.179s
s455	364	1664	100	3	6	2.331s
s2445	1820	8795	160	4	8	4.323s
s22445	8568	41942	320	5	8	28.213s
sq4	126	585	16	4	5	0.313s
sq5	462	2175	32	5	44	2.135s
sing	210	4998	21	2	14	0.438s
kruppa	792	15822	1	5	1	0.698s

```
  cout<<setprecision(12);
  Pol x1("x1;"), x2("x2;"), p1("x1^2-2*x2-x1+1;"), p2("x2^2+x1*x2-x2-2;");

> Polynomials:
>  (1)+(-1)*x1+(-2)*x2+(1)*x1^2
>  (-2)+(-1)*x2+(1)*x1*x2+(1)*x2^2

  list<Pol> l;
  l.push_back(x1);   l.push_back(x1*x1-p1);   l.push_back(x1*x2);   l.push_back(x1*x1*x2-x2*p1-p2*2);

  MAT A = MatrixOf(l,lapack<double>());
> Matrix of multiplication by x1:
> [ 0 -1 0 4 ]
> [ 1 1 0 0 ]
> [ 0 2 0 1 ]
> [ 0 0 1 -1 ]

  Eigenval(A, Error);
> Eigenvaluse of A:
> [(0.164857675968,1.60003037108), (0.164857675968,-1.60003037108), (1.70750475735,0), (-2.03722010929,0)]
> Error on the eigenvalues:
> [7.76930157403e-15, 7.76930157403e-15, 6.96147682479e-15, 8.3833375683e-15]

 Solve(A,2,Eigen());
> [ (0.164857675968,-1.60003037108) (0.164857675968,1.60003037108) (1.70750475735,0) (-2.03722010929,-0) ]
> [ (-0.848888405503,0.536237897085) (-0.848888405503,-0.536237897085) (1.10403386952,0) (3.59374294149,-0) ]
```

Approximate gcd

The polynomials: (`UPolyDense<upar<double> >`)

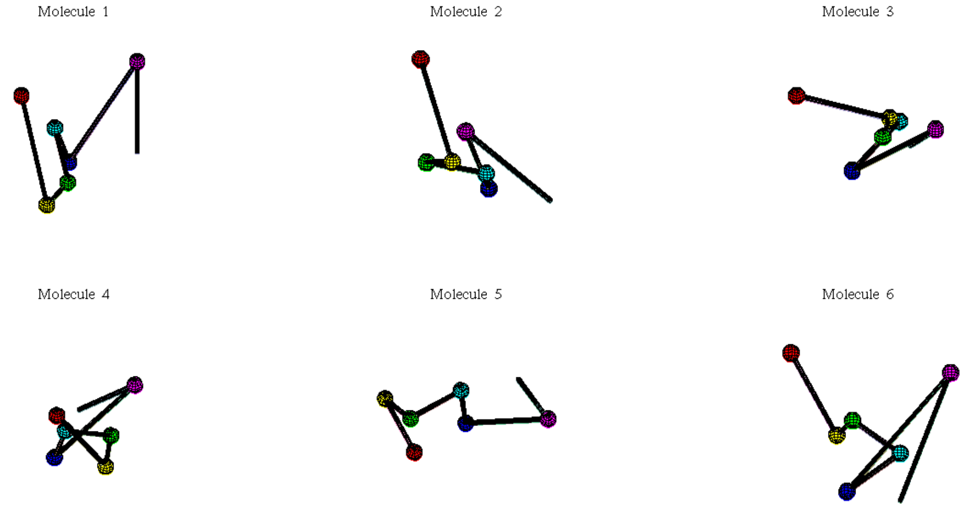
$$a = (x^2 - 1.001)\,(x^2 + 1.0000001)\,(x^3 + 2\,x^2 - 2.2124)$$
$$b = (x^2 - 0.999)\,(x^2 + 1.0000003)\,(x^4 + 3\,x - 1.000227)$$

```
Singular values of S (MatStd<lapack<double> >):
[1, 0.964545850108389, 0.905519306037247, 0.839840910634326,
0.557478407804673, 0.547482591513168, 0.501907498546123, 0.376699682983808,
0.23971114795765, 0.168487023795092, 0.0278406949880667,
0.000212983307990487, 0.000146757553148298,
6.49643592843973e-08, 6.11881302732178e-08]

Approximate gcd for epsilon=1e-08 :
 (1)
Approximate gcd for epsilon=1e-07 :
 (1)*x^2+(-1.28880336045428e-05)*x^1+(0.999969574944473)
Approximate gcd for epsilon=0.001 :
 (1)*x^4+(0.0013301843224)*x^3+(0.0010355450046)*x^2+(0.0013302481031)*x^1+(-0.99896440408)
```

Conformation of molecules

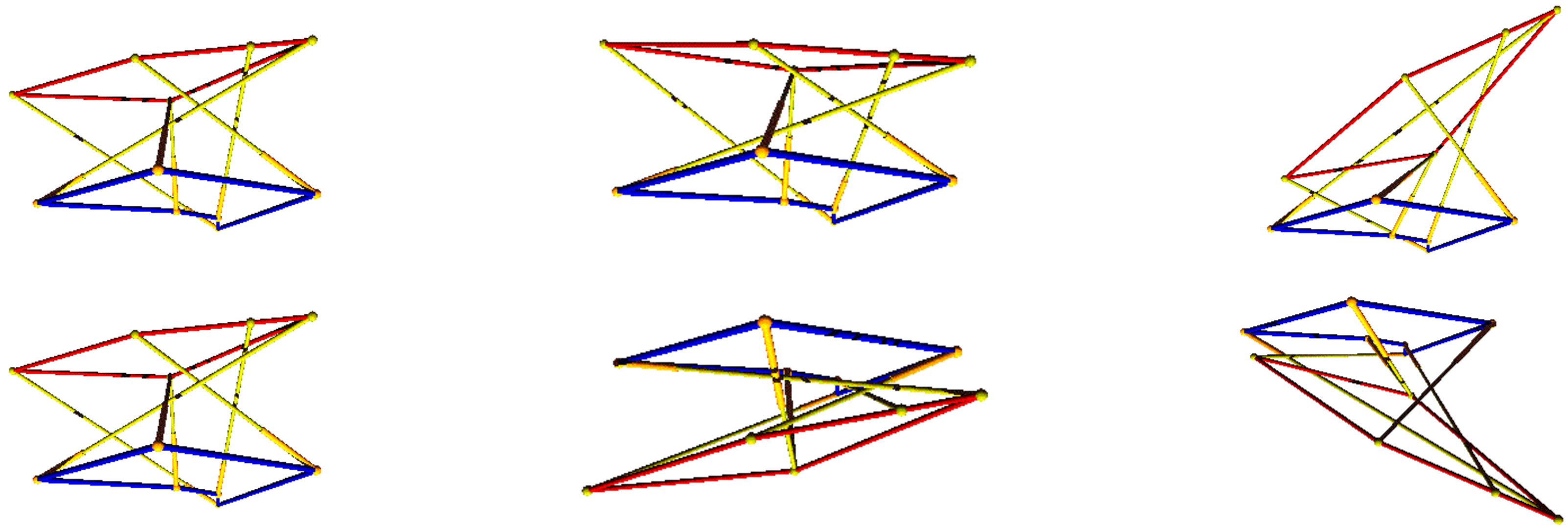
Problem: Compute the possible conformations of a molecule when the position and orientation of the extremity is fixed (by O. Ruatta).



Error: $\|A_1 \circ \cdots \circ A_6 - H\| < 10^{-6}$ **Time:** $0.090s$

Direct kinematic problem of a parallel robot

Problem: Compute the position of the platform for fixed lengths of the arms (by H. Prieto).



Error: $|\|R\,Y_i + T - X_i\| - d_i^2| < 10^{-6}$

Time: $132s + 0.5s$