Méthodes matricielles pour calculer des conformations moléculaires

Ioannis Z. Emiris
INRIA Sophia-Antipolis

System solving: Add polynomial

Given $f_1, \ldots, f_n \in K[x_1, x_1^{-1}, \ldots, x_n, x_n^{-1}]$.

Hypothesis: 0-dimensional ideal, distinct zeros.

Add polynomial $f_0 = u + r_1 x_1 + \cdots + r_n x_n$,
with random r_i, indeterminate u.

Construct resultant matrix $M(u)$ for $f_0, f_1, \ldots, f_n$.
At root α, $u = -\sum r_i \alpha_i$,

$$\begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22}(u) \end{bmatrix} \begin{bmatrix} \vdots \\ \alpha^p \\ \vdots \\ \alpha^q \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ \alpha^a f_i(\alpha) \\ \vdots \\ \alpha^b f_0(u, \alpha) \\ \vdots \end{bmatrix} = \begin{bmatrix} \vdots \\ 0 \\ \vdots \\ 0 \\ \vdots \end{bmatrix}.$$

If $\det M_{11} \neq 0$, let $M'(u) = M_{22}(u) - M_{21} M_{11}^{-1} M_{12}$,
where M' has dimension $\mathrm{MV}(f_1, \ldots, f_n)$:

$$(M' + uI) v'_\alpha = 0.$$

- Ratios of the entries of eigenvectors v'_α yield α, if integer points q affinely span $\mathbb{Z}^n$.
- Otherwise, use some entries of $v_\alpha = -M_{11}^{-1} M_{12} v'_\alpha$, where $(v_\alpha, v'_\alpha)^T$ is the respective eigenvector of M.

[AuzingerStetter]

System solving: Hide variable

"Hide" a variable in the coefficient field:
$f_0, f_1, \ldots, f_n \in K[x_0][x_1, x_1^{-1}, \ldots, x_n, x_n^{-1}]$

Hypothesis: x_0-coordinates distinct, $|M(x_0)| \not\equiv 0$.

$$\det M(x_0) = \begin{vmatrix} M_{11} & M_{12}(x_0) \\ M_{21} & M_{22}(x_0) \end{vmatrix} = \begin{vmatrix} M_{11} & M_{12}(x_0) \\ 0 & M'(x_0) \end{vmatrix},$$

$$\begin{aligned} \det M'(x_0) &= \det(A_d x_0^d + \cdots + A_1 x_0 + A_0) \\ &= \det A_d \det(x_0^d + \cdots + A_d^{-1} A_1 x_0 + A_d^{-1} A_0). \end{aligned}$$

- If $\det A_d \neq 0$, define companion matrix C:

$$C = \begin{bmatrix} 0 & I & & 0 \\ \vdots & & \ddots & \\ 0 & 0 & & I \\ -A_d^{-1}A_0 & -A_d^{-1}A_1 & \cdots & -A_d^{-1}A_{d-1} \end{bmatrix}$$

The eigenvalues of C are the x_0-coordinates of the solutions and its eigenvectors contain the values of the monomials indexing M' at the roots.

- Improve the condition number of A_d by $x \mapsto \frac{t_1 y + t_2}{t_3 y + t_4}$, $t_i \in \mathbb{Z}$ (rank balancing).

- If A_d remains ill-conditioned, solve the generalized eigenproblem

$$\begin{bmatrix} I & & & \\ & \ddots & & \\ & & I & \\ & & & A_d \end{bmatrix} x + \begin{bmatrix} 0 & -I & & \\ & & \ddots & \\ & & & -I \\ A_0 & A_1 & \cdots & A_{d-1} \end{bmatrix}.$$

Software: Combinatorial geometry

- Mixed Volume

Input: Polynomial supports (well-constrained)

Output: Monomial basis of quotient,
bound on the number of roots,
starting system of homotopy.

+ Distributed version using `Central Control`.

- Mixed Subdivision

Input: Polynomial supports (well-constrained)

Output: Complete mixed subdivision,
including output of Mixed Volume.

Portability: SunOS, DECosf, Linux, SGI.

Interface for Mixed Volume: ALP.

Input formats: Maple (or ALP).

Code: Ansi-C.

Availability: `www.inria.fr/saga/emiris`,
`ftp-sop.inria.fr/saga/emiris`

Software: Symbolic algebra

Input: Polynomial supports (over-constrained)
Output: Nontrivial multiple of resultant,
solving is reduced to eigenproblem,
optimal size in specified polynomial.

- Incremental sparse resultant matrix

Feature: Exact matrix whenever possible [SZ94].
Code: Ansi-C.
Portability: SunOS, DECosf, Linux.
Interface: ALP.
Input formats: Maple, ALP.

- Subdivision-based sparse resultant matrix

Features: Bounded matrix dimension,
conjectured exact rational expression.
Code: Maple.

Availability: `www.inria.fr/saga/emiris`,
`ftp-sop.inria.fr/saga/emiris`

Software: Numerical linear algebra

- Polynomial system solving

Input: Polynomial coefficients,
resultant matrix

Output: Superset of common roots

Features: Numerical linear algebra,
trade-off between speed and accuracy,
can factor out constant submatrix,
rank balancing of matrix polynomial,
regular or generalized eigenproblem.

Portability: SunOS, Linux.

Interface: Incremental matrix, LAPACK.

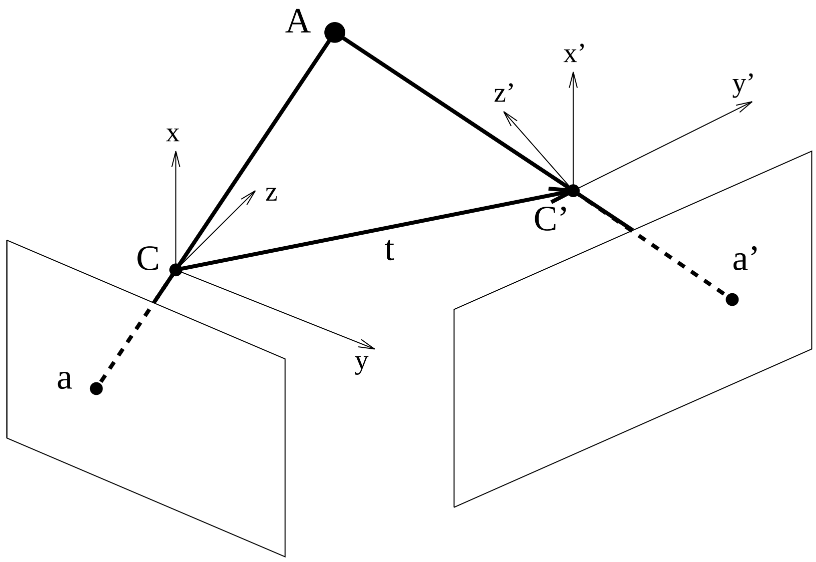
Code: Ansi-C.

Availability: `www.inria.fr/saga/emiris`,
`ftp-sop.inria.fr/saga/emiris`

Future work: Exploit quasi-Toeplitz structure,
exploit sparseness (many zeroes),
handle degenerate inputs,
integrate with ALP.

Camera Motion

Compute the displacement of a camera from the coordinates of 5 points in two images (minimum number of points).



Rotation and translation quaternions $\dot{q}, \dot{t}$; points $\dot{a}_i, \dot{a}'_i$.
Introduce $\dot{d} = \dot{t}\dot{q}$: bilinear 6×6 system in $\dot{q}, \dot{d}$.

$$\begin{aligned} a_i^T(t \times R a'_i) = 0 \quad &\Leftrightarrow \quad (\dot{a}_i \dot{q})^T(\dot{d}\dot{a}'_i) = 0, \quad i = 1, \ldots, 5, \\ \dot{t} = [0, t] \quad &\Rightarrow \quad \dot{d}^T \dot{q}^* = 0. \end{aligned}$$

Set $\dot{q} = [1, q]$, $\dot{d} = [1, d]$.

Exact Mixed Volume $= 20$ (1m 5s, 80 MHz).
Bezout bound $= 64$.

Camera Motion Computations

Hide q_1: overconstrained system is bilinear.
First five polynomials are multigraded of type $(2,3;1,1)$.

Algorithm II constructs exact sparse resultant:
dimension$(M) = 60$, dimension$(M') = 20$.

Off-line time (DEC Alpha 3000): 13s
(5-fold Mixed Volumes, integer point sets $\mathcal{E}_i$, matrix M).

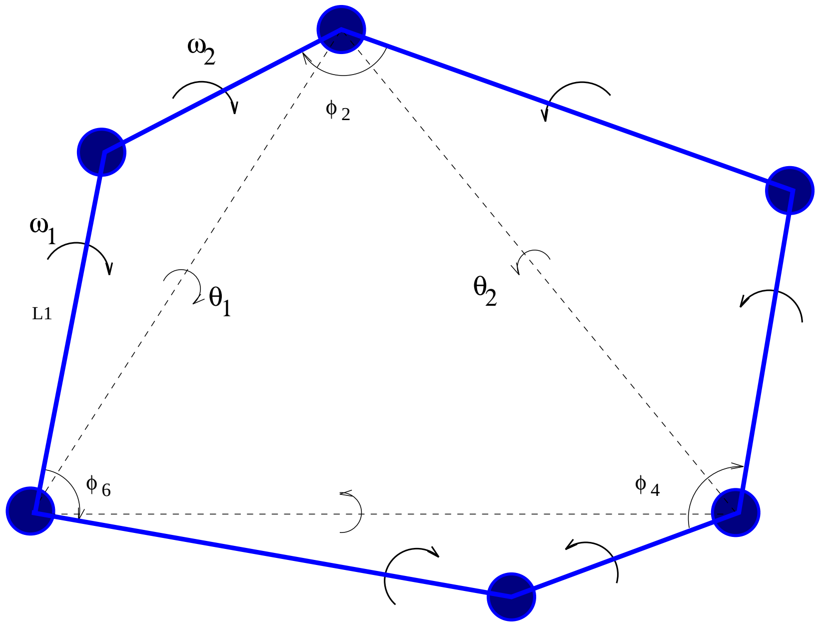
1. "Generic" instance [Demazure]: 20 real solutions.
20×20 monic matrix polynomial, standard eigenproblem.

2. "Special" instance [Faugeras,Maybank]: 8 real roots.
$\tau = 60.01455°$ rotation about $\widehat{z}$, $t = (.01, .01, -1)$.
$(M_{11}) > 10^4$, $(A_d) > 10^8$: 60×60 matrix polynomial, generalized eigenproblem.

	max error	on-line time (M', roots)
1	10^{-6}	.2s (80 MHz)
2	$7 \cdot 10^{-5}$	1s (60 MHz)

[Tsai,Huang'84;Luong'92]: Least-squares on ≥ 8 points.
Time .08s (Sun Sparc 20/61).
On (2) returns $\tau \simeq 60.0013°$ about $(-10^{-5}, 10^{-5}, 1)$
and $t \simeq (-10^{-5}, -10^{-5}, -1)$.

Cyclic Molecules



Given bond lengths L_i and bond angles ϕ_i, compute dihedral angles ω_i: inverse kinematics.

Consecutive axes intersect: 3×3 problem [Parsons]: Rigid flaps form flap angles τ_1, τ_2, τ_3 with dashed triangle; 3 remaining constraints on ϕ_2, ϕ_4, ϕ_6 give symmetric equations:

$$c_{i1} + c_{i2}t_j^2 + c_{i3}t_k^2 + c_{i4}t_j^2 t_k^2 + c_{i5}t_j t_k = 0, \quad (i,j,k) \in S(1,2,3)$$
$$t_i = tan(\tau_i/2).$$

Exact Mixed Volume $= 16$ (.2s, 80 MHz).
Bezout bound $= 64$.

Computing Conformations

1. 8 roots $\pm(1,1,1), \pm(5,-1,-1), \pm(-1,5,-1), \pm(-1,-1,5)$.
Add u-polynomial: sparse resultant degree $= 52$,
M (by algorithm II) of dimension 86.
30×30 u-matrix, generalized eigendecomposition.

2. Pure cyclohexane: all ϕ_i, L_i equal.
Perturb randomly by $\leq 10\%$: 4 real solutions.

Hide t_3: sparse resultant degree $= 12$,
M (by algorithm II) of dimension 16, $M = It_3^2 + B_1 t_3 + B_0$.
32×32 companion matrix, standard eigendecomposition.

	off-line (60 MHz)	accuracy	on-line (60 MHz)
1	.2s	6 digits	.4s
2	.1s	8 digits	.2s

[Manocha et.al.]: 32×32 eigenproblem: .045s (SGI Onyx).

3. 16 real solutions.
Same approach, similar results and performance.

Distance matrices

Definition. A distance matrix D is square, $D_{ii} = 0$ and $D_{ij} = D_{ji} \geq 0$.

Definition. Distance matrix D is euclidean and embeddable in $\mathbb{R}^k$ if, for points $p_i \in \mathbb{R}^k$, $D_{ij} = \frac{1}{2}\text{dist}(p_i - p_j)^2$.

Theorem. [Schoenberg35]
Distance matrix D is embeddable in $\mathbb{R}^k$
iff $PDP \leq 0$ and $\text{rank}(PDP) \leq k$,
where $P = I - \frac{1}{n}ee^T$, $e^T = [1, \ldots, 1]$.

Theorem. [ChaCro84]
Distance matrix D is euclidean iff
$B = \begin{bmatrix} -D & e \\ e^t & 0 \end{bmatrix}$ has exactly one eigenvalue < 0.
Further, D embeds in $\mathbb{R}^k$ iff $\text{rank}(B) = k + 2$.

Relevant work

• Computational biology [Havel,Crippen,Najfeld]
Minimize rank, apply optimization on error as function of intervals from NMR experiments. Efficient programs for large molecules.

• **Theorem.** [Mathar85] Let $e^T = [1, \ldots, 1]$.
For any vector $s : s^T e = 1$, let $P = I - es^T$ and define norm $|M|_s = \| PMP^T \|$.
Given distance matrix D and k, for any s as above, we can construct distance matrix E embeddable in $\mathbb{R}^k$, with minimum $|D - E|_s$.

• **Theorem.** [WicksDecarlo95]
Given matrices B, R s.t. the singular values of $B - tR$ are distinct for almost all t. Then $f(t) = \sigma_n(B - tR)$ is continuous, locally differentiable and $f'(t) = -u^T Rv$, where $u^T(B - tR)v = \sigma_n$.
$\rightarrow$ Newton-based iteration minimizing σ_n.

Extension to higher-dimensional linear spaces for R, reflecting problem's structure.
Matlab code [NikitopE99] applied to σ_k, $k < n$, solves molecules with < 40 atoms.