Finding one root of a polynomial system

A *brief* review of Smale's 17th problem

Pierre Lairez

Inria Saclay

SIAM AG 17

SIAM conference on applied algebraic geometry

2 august 2017, Atlanta

Solving polynomial systems in polynomial time?

Can we compute the roots of a polynomial system in polynomial time?

Likely not, deciding *feasibility* is NP-complete.

Can we compute the complex roots of n equations in n variables in polynomial time?

No, there are too many roots.

Bézout bound vs. input size (n polynomial equations, n variables, degree D)

degree	D	2	n	$D \gg n$
input size	$n\binom{D+n}{n}$	$\sim \frac{1}{2} n^3$	$\sim \frac{1}{\sqrt{\pi}} n^{\frac{1}{2}} 4^n$	$\sim \frac{1}{(n-1)!} D^n$
#roots	D^n	2^n	n^n	D^n

Finding one root: a purely numerical question

#roots $\gg$ input size	To compute a single root, do we have to pay for #roots?
using exact methods	Having one root is having them all (generically).
using numerical methods	One may approximate one root disregarding the others.
polynomial complexity?	Maybe, but only with numerical methods.

This is Smale's question

Smale 17th problem

"Can a zero of n complex polynomial equations in n unknowns be found approximately, on the average, in polynomial time with a uniform algorithm?"

— *S. Smale, 1998*

approximate root A point from which Newton's iteration converges quadratically.

polynomial time with respect to the input size.

on the average with respect to some input distribution.

uniform algorithm A Blum–Shub–Smale machine (a.k.a. real random access machine):

- registers store exact real numbers,
- unit cost arithmetic operations,
- branching on positivity testing.

Infinite precision?! Yes, but we still have to deal with stability issues.

The model is very relevant for this problem.

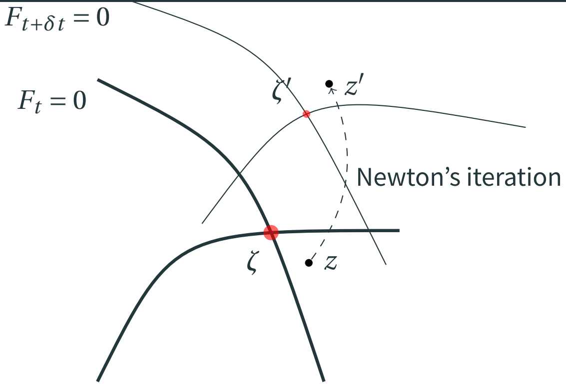
Another brick in the wall

Problem solved!

Shub, Smale (1990s)	Quantitative theory of Newton's iteration
	Complexity of numerical continuation
Beltrán, Pardo (2009)	Randomization
Bürgisser, Cucker (2011)	Deterministic polynomial average time when $D \ll n$ or $D \gg n$
	Smoothed analysis
Lairez (2017)	Derandomization

Numerical continuation

Complexity of numerical continuation



How to choose the step size δt?

Too big, we loose the root.

Too small, we waste time.

Theorem (Shub 2009)

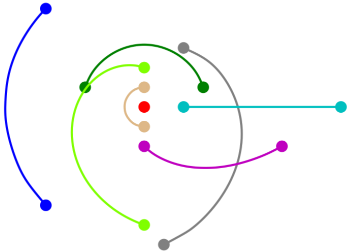
One can compute an approximate root of F_1 given an approximate root of F_0 with

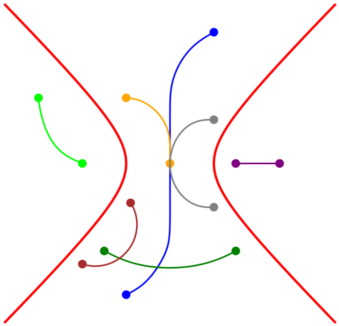
$$\#\text{steps} \leqslant 136 D^{\frac{3}{2}} \int_0^1 \mu(F_t, \zeta_t)^2 \|\dot{F}_t\| \mathrm{d}t, \text{ where } \mu(F,\zeta) \text{ is the condition number.}$$

How to choose the path?

linear interpolation $F_t = tF_1 + (1-t)F_0$

a better path? That exists (Beltrán, Shub 2009) but there is no algorithm so far.





(Pictures by Juan Criado del Rey.)

Randomization of the start system

A randomized start system

Conditioning of a random system

- F a random polynomial system, uniformly distributed on some sphere
- ζ a random root of $F = 0$, uniformly chosen among the D^n roots.

Theorem (Beltrán, Pardo 2011; Bürgisser, Cucker 2011)

$\mathbb{E}(\mu(F,\zeta)^2) \leqslant n \cdot$ (the input size)

- Is the conditioning good all along the continuation path?
  How to sample (F,ζ)? Chicken-and-egg problem?

Complexity of numerical continuation with random endpoints

F_0, F_1 random polynomial systems of norm 1, uniformly distributed.

ζ_0 a random root of F_0, uniformly distributed.

F_t linear interpolation (normalized to have norm 1).

ζ_t continuation of ζ_0.

lemma $\forall t$, F_t is uniformly distributed and ζ_t is uniformly distributed among its roots.

$$\#\text{steps} \leqslant 136\, D^{\frac{3}{2}}\, d_{\mathbb{S}}(F_0, F_1) \int_0^1 \mu(F_t, \zeta_t)^2 \mathrm{d}t \quad \text{(Shub 2009)}$$

$$\mathbb{E}[\#\text{steps}] \leqslant 136\pi\, D^{\frac{3}{2}}\, \mathbb{E}\left[\int_0^1 \mu(F_t, \zeta_t)^2 \mathrm{d}t\right]$$

$$\leqslant 136\pi\, D^{\frac{3}{2}} \int_0^1 \mathbb{E}\left[\mu(F_t, \zeta_t)^2\right] \mathrm{d}t \quad \text{(Tonelli's theorem)}$$

$$= \mathcal{O}\left(nD^{\frac{3}{2}}\ \text{(input size)}\right) \quad \text{(Beltrán, Pardo 2011; Bürgisser, Cucker 2011)}$$

first try Sample $\zeta \in \mathbb{P}^n$ uniformly,
sample F uniformly in $\{F \text{ s.t. } F(\zeta) = 0\} \cap \mathbb{S}$.

✖ F is not uniformly distributed.

BP method Sample a *linear* system L uniformly,
compute its unique root $\zeta \in \mathbb{P}^n$,
sample F uniformly in $\left\{F \text{ s.t. } F(\zeta) = 0 \text{and } \mathrm{d}_\zeta F = L\right\} \cap \mathbb{S}$.

✔ F and ζ are uniformly distributed.

Solves Smale's problem *with randomization*.
Total average complexity $\mathcal{O}\left(nD^{\frac{3}{2}}(\text{input size})^2\right)$.

Smoothed analysis

Bürgisser, Cucker (2011)

average analysis gives little information on the complexity of solving *one* given system.

worst-case analysis is irrelevant here (unbounded close to a system with a singular root).

smoothed analysis bridges the gap and gives information on a single system F pertubed by a Gaussian noise ε of variance σ^2. This models an input data that is only approximate.

$$\sup_{\text{system } F} \mathbb{E}\left[\text{cost of computing one root of } F+\varepsilon\right] = \mathcal{O}(\sigma^{-1} n D^{\frac{3}{2}} N^2).$$

$\mathbb{E}$: average-case w.r.t. the noise

sup: worst-case

Derandomization

Truncation and noise extraction

x, a random uniformly distributed variable in $[0, 1]$.

$$0.50529019746531591013322667888500001621027$$

noise extraction ↑

$$x = \underline{0.6044025624180895161178081249104686}\overline{50529019746531591013322667888500001621027}$$

truncation ↓

$$0.6044025624180895161178081249104686$$

- The truncation is a random variable that is close to x.
- The noise is an independent from the truncation and uniformly distributed in $[0, 1]$.

Truncation and noise extraction on an odd-dimensional sphere

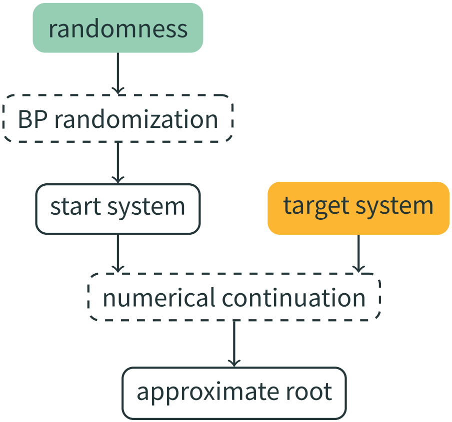
$$\mathbb{S}(\mathscr{H}) \simeq \mathbb{S}^{2n-1} \nearrow \text{mesh on } \mathbb{S}^{2n-1} \qquad \text{(truncation)}$$

$$\mathbb{S}(\mathscr{H}) \simeq \mathbb{S}^{2n-1} \searrow [0,1]^{2n-1} \xrightarrow{S} \mathbb{S}^{2n-1} \qquad \text{(noise)}$$

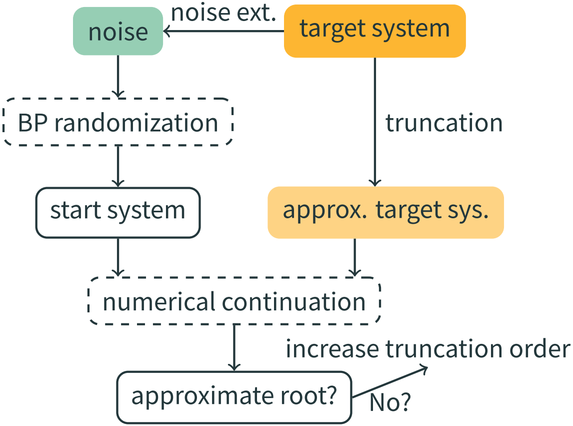
- S is a measure preserving map due to Sibuya (1962).
- The noise is *nearly* uniformly distributed and *nearly* independent from the truncation.

Derandomization

Beltràn and Pardo's randomization



Lairez's derandomization



Solves Smale's problem with a *deterministic algorithm*.

Randomness is in Smale's question from its very formulation asking for an average analysis.

Quasi-optimal complexity

Complexity exponent in Smale's problem

$$\text{total cost} = \mathcal{O}\Big(\underbrace{\text{(input size)}}_{\text{cost of Newton's iteration}} \cdot \#\text{steps} \Big).$$

Beltrán, Pardo (2009) $\mathbb{E}(\#\text{steps}) = (\text{input size})^{1+o(1)}$

Armentano, Beltrán, Bürgisser, Cucker, Shub (2016)

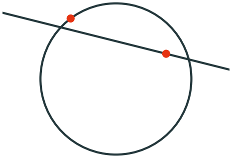
$$\mathbb{E}(\#\text{steps}) = (\text{input size})^{\frac{1}{2}+o(1)}$$

work in progress $\mathbb{E}(\#\text{steps}) = \operatorname{poly}(n, D) = (\text{input size})^{o(1)}$

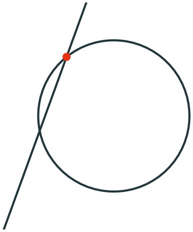
Bigger steps with unitary paths

observation Relatively small pertubation of a typical system F (in the space of all systems) changes everything. Makes it difficult to make bigger steps.

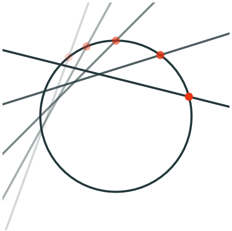
idea Perform the continuation is a lower dimensional parameter space:
We allow only rigid motions of the equations rather than arbitrary deformations.



compute one solution of each equation



move the hypersurfaces to make the solution match



continuously return to the original position

Unitary paths

In more details...

parameter space $U(n+1)\times\cdots\times U(n+1)$, that is n copy of the unitary group. This has dimension $\sim n^3$, compare with $n\cdot\binom{D+n}{n}$.

paths Geodesics in the parameter space.

randomization Same principle as Beltràn and Pardo's randomization.

complexity $\mathbb{E}(\#\text{steps}) = \operatorname{poly}(n, D)$.

Thank you!

Present slides are online at *pierre.lairez.fr* with bibliographic references.

References I

- Armentano, D., C. Beltrán, P. Bürgisser, F. Cucker, M. Shub (2016). "Condition Length and Complexity for the Solution of Polynomial Systems". In: *Found. Comput. Math.*
- Beltrán, C., L. M. Pardo (2009). "Smale's 17th Problem: Average Polynomial Time to Compute Affine and Projective Solutions". In: *J. Amer. Math. Soc.* 22.2, pp. 363–385.
- – (2011). "Fast Linear Homotopy to Find Approximate Zeros of Polynomial Systems". In: *Found. Comput. Math.* 11.1, pp. 95–129.
- Beltrán, C., M. Shub (2009). "Complexity of Bezout's Theorem. VII. Distance Estimates in the Condition Metric". In: *Found. Comput. Math.* 9.2, pp. 179–195.
- Bürgisser, P., F. Cucker (2011). "On a Problem Posed by Steve Smale". In: *Ann. of Math. (2)* 174.3, pp. 1785–1836.
- Hauenstein, J. D., A. C. Liddell (2016). "Certified Predictor–corrector Tracking for Newton Homotopies". In: *Journal of Symbolic Computation* 74, pp. 239–254.

References II

- Hauenstein, J. D., F. Sottile (2012). “Algorithm 921: alphaCertified: Certifying Solutions to Polynomial Systems”. In: *ACM Transactions on Mathematical Software* 38.4, pp. 1–20.
- Lairez, P. (2017). “A Deterministic Algorithm to Compute Approximate Roots of Polynomial Systems in Polynomial Average Time”. In: *Found. Comput. Math.*
- Leykin, A. (2011). “Numerical Algebraic Geometry”. In: *Journal of Software for Algebra and Geometry* 3.1, pp. 5–10.
- Li, T.-Y. (1987). “Solving Polynomial Systems”. In: *The Mathematical Intelligencer* 9.3, pp. 33–39.
- Morgan, A., A. Sommese (1987). “A Homotopy for Solving General Polynomial Systems That Respects M-Homogeneous Structures”. In: *Applied Mathematics and Computation* 24.2, pp. 101–113.
- Shub, M. (1993). “Some Remarks on Bezout’s Theorem and Complexity Theory”. In: *From Topology to Computation: Proceedings of the Smalefest*. Springer, New York, pp. 443–455.
- – (2009). “Complexity of Bezout’s Theorem. VI. Geodesics in the Condition (Number) Metric”. In: *Found. Comput. Math.* 9.2, pp. 171–178.

References III

Shub, M., S. Smale (1993a). "Complexity of Bézout's Theorem. I. Geometric Aspects". In: *J. Amer. Math. Soc.* 6.2, pp. 459–501.

– (1993b). "Complexity of Bezout's Theorem. II. Volumes and Probabilities". In: *Computational Algebraic Geometry (Nice, 1992)*. Vol. 109. Progr. Math. Birkhäuser Boston, Boston, MA, pp. 267–285.

– (1993c). "Complexity of Bezout's Theorem. III. Condition Number and Packing". In: *J. Complexity* 9.1, pp. 4–14.

– (1994). "Complexity of Bezout's Theorem. V. Polynomial Time". In: *Theoret. Comput. Sci.* 133.1. Selected papers of the Workshop on Continuous Algorithms and Complexity (Barcelona, 1993), pp. 141–164.

– (1996). "Complexity of Bezout's Theorem. IV. Probability of Success; Extensions". In: *SIAM J. Numer. Anal.* 33.1, pp. 128–148.

Sibuya, M. (1962). "A Method for Generating Uniformly Distributed Points on N-Dimensional Spheres". In: *Ann. Inst. Statist. Math.* 14, pp. 81–85.

References IV

Smale, S. (1986). “Newton’s Method Estimates from Data at One Point”. In: *The Merging of Disciplines: New Directions in Pure, Applied, and Computational Mathematics (Laramie, Wyo., 1985)*. Springer, New York, pp. 185–196.

– (1998). “Mathematical Problems for the next Century”. In: *The Mathematical Intelligencer* 20.2, pp. 7–15.