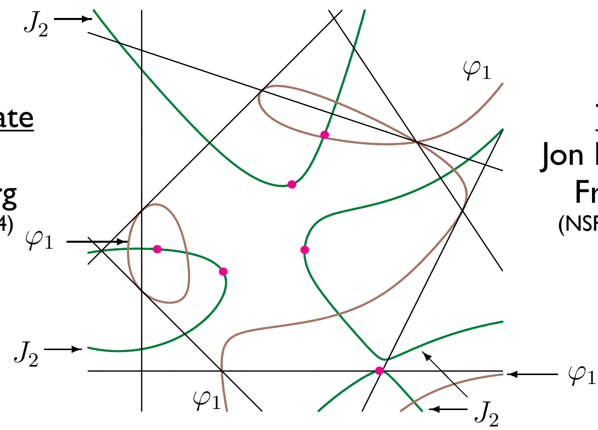
Khovanskii-Rolle continuation for real solutions of polynomial systems

Colorado State
Dan Bates
Matt Niemerg
(NSF DMS-0914674)



Texas A&M
Jon Hauenstein
Frank Sottile
(NSF DMS-0915211)

BIRS   Randomization, Relaxation, & Complexity   March 4, 2010

Motivation

Problem: Approximate all real roots of a zero-dimensional polynomial system.

Solution: There are many options, but all have problems.

Motivation

Problem: Approximate all real roots of a zero-dimensional polynomial system.

Solution: There are many options, but all have problems.

One option: Homotopy continuation.

This is a good method in general, but complexity depends on the number of complex solutions.

Motivation

Problem: Approximate all real roots of a zero-dimensional polynomial system.

Solution: There are many options, but all have problems.

One option: Homotopy continuation.

This is a good method in general, but complexity depends on the number of complex solutions.

Today's method: Numerical (mostly non-homotopy) method with complexity depending on the number of real roots.

Motivation

Question: Given polynomial system

$$f : \mathbb{R}^N \to \mathbb{R}^N$$

with support

$$W = \cup_{i=1}^{N} supp(f_i)$$

having N+L+1 monomials, how many solutions are there?

Motivation

Question: Given polynomial system

$$f : \mathbb{R}^N \to \mathbb{R}^N$$

with support

$$W = \cup_{i=1}^{N} supp(f_i)$$

having N+L+1 monomials, how many solutions are there?

Bézout (1776): no more than the product of the degrees (complex)

Motivation

Question: Given polynomial system

$$f : \mathbb{R}^N \to \mathbb{R}^N$$

with support

$$W = \cup_{i=1}^{N} supp(f_i)$$

having N+L+1 monomials, how many solutions are there?

Bézout (1776): no more than the product of the degrees (complex)

Bernstein-Kouchnirenko-Khovanskii (1976): no more than

$$N!\text{vol}(\text{conv}(W)) \quad \text{(complex)}$$

Motivation

Question: Given polynomial system

$$f : \mathbb{R}^N \to \mathbb{R}^N$$

with support

$$W = \cup_{i=1}^{N} supp(f_i)$$

having N+L+1 monomials, how many solutions are there?

Bézout (1776): no more than the product of the degrees
(complex)

Bernstein-Kouchnirenko-Khovanskii (1976): no more than

$$N!\text{vol}(\text{conv}(W)) \quad \text{(complex)}$$

Remark: Homotopy methods rely on these sorts of bounds.
(stick around for the next two talks)

Bertrand-Bihan-Sottile (2006): 2N+1 (L=2, real, sharp)

Bihan (2007): N+1 (L=2, positive real, sharp)

Bertrand-Bihan-Sottile (2006): 2N+1 (L=2, real, sharp)

Bihan (2007): N+1 (L=2, positive real, sharp)

Bihan-Sottile (2007): no more than

$$\frac{e^2+3}{4} 2^{\binom{L}{2}} N^L \quad \text{(positive real)}$$

Bihan-Rojas-Sottile (2007): That is asymptotically sharp for L fixed and N large.

Bertrand-Bihan-Sottile (2006): 2N+1 (L=2, real, sharp)

Bihan (2007): N+1 (L=2, positive real, sharp)

Bihan-Sottile (2007): no more than

$$\frac{e^2+3}{4}2^{\binom{L}{2}}N^L \quad \text{(positive real)}$$

Bihan-Rojas-Sottile (2007): That is asymptotically sharp for L fixed and N large.

Bates-Bihan-Sottile (2007): no more than

$$\frac{e^4+3}{4}2^{\binom{L}{2}}N^L \quad \text{(real)}$$

Bertrand-Bihan-Sottile (2006): 2N+1 (L=2, real, sharp)

Bihan (2007): N+1 (L=2, positive real, sharp)

Bihan-Sottile (2007): no more than

$$\frac{e^2+3}{4}2^{\binom{L}{2}}N^L \quad \text{(positive real)}$$

Bihan-Rojas-Sottile (2007): That is asymptotically sharp for L fixed and N large.

Bates-Bihan-Sottile (2007): no more than

$$\frac{e^4+3}{4}2^{\binom{L}{2}}N^L \quad \text{(real)}$$

The ratio of this by Bihan-Sottile's bound is constant!
(Come to Corben Rusek's talk at 5:15....)

Motivation

Maurice: We need methods that depend on complexity over the reals. (People who have systems they need to solve feel similarly.)

The proof of the 2007 Bihan-Sottile paper indicates a clear numerical method.

This talk: Khovanskii-Rolle continuation. Features:

- (mostly non-homotopy) numerical method
- finds all solutions on the real torus
- complexity (of some sort) is bounded above by a constant multiple of the number of real solutions
- the actual computational cost is often better than complexity bound

Timings (more motivation)

- Example 1:

$$
\begin{aligned}
cd &= \tfrac{1}{2}be^2 + 2a^{-1}b^{-1}e - 1 & cd^{-1}e^{-1} &= \tfrac{1}{2}(1 + \tfrac{1}{4}be^2 - a^{-1}b^{-1}e) \\
bc^{-1}e^{-2} &= \tfrac{1}{4}(6 - \tfrac{1}{4}be^2 - 3a^{-1}b^{-1}e) & bc^{-2}e &= \tfrac{1}{2}(8 - \tfrac{3}{4}be^2 - 2a^{-1}b^{-1}e) \\
ab^{-1} &= 3 - \tfrac{1}{2}be^2 + a^{-1}b^{-1}e\ ,
\end{aligned}
$$

  102 complex solutions, 10 real solutions
  KhRo took 1.4 seconds, Bertini took 9 sec (on one processor).

Timings (more motivation)

- Example 1:

$$\begin{aligned} cd &= \tfrac{1}{2}be^2 + 2a^{-1}b^{-1}e - 1 & cd^{-1}e^{-1} &= \tfrac{1}{2}(1 + \tfrac{1}{4}be^2 - a^{-1}b^{-1}e) \\ bc^{-1}e^{-2} &= \tfrac{1}{4}(6 - \tfrac{1}{4}be^2 - 3a^{-1}b^{-1}e) & bc^{-2}e &= \tfrac{1}{2}(8 - \tfrac{3}{4}be^2 - 2a^{-1}b^{-1}e) \\ ab^{-1} &= 3 - \tfrac{1}{2}be^2 + a^{-1}b^{-1}e\ , \end{aligned}$$

  102 complex solutions, 10 real solutions
  KhRo took 1.4 seconds, Bertini took 9 sec (on one processor).

- Example 2:

$$\begin{aligned} 10500 - tu^{492} - 3500t^{-1}u^{463}v^5w^5 &= 0 \\ 10500 - t - 3500t^{-1}u^{691}v^5w^5 &= 0 \\ 14000 - 2t + tu^{492} - 3500v &= 0 \\ 14000 + 2t - tu^{492} - 3500w &= 0. \end{aligned}$$

  7663 complex solutions, 6 real solutions
  KhRo took 23 sec, PHCpack took 39.3 min (on one processor). Notice the degrees....

Gameplan

1. **Background on proof of Bihan-Sottile bound**
2. Proof $\longrightarrow$ Algorithm
3. Example (pretty pictures)
4. A word about complexity
5. Further plans

Background: 2 main techniques

Gale Duality

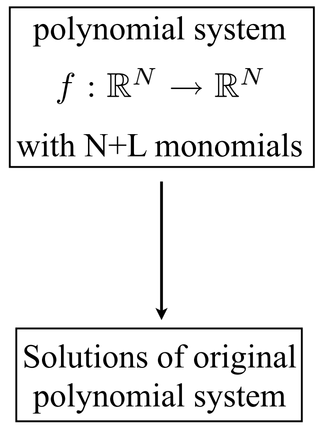
A polynomial system with N+L+1 monomials has a dual system of "master functions" defined in the complement of a hyperplane arrangment AND there is a bijection between the solutions (under a technical condition).
(see Bihan-Sottile).

Khovanskii-Rolle Theorem

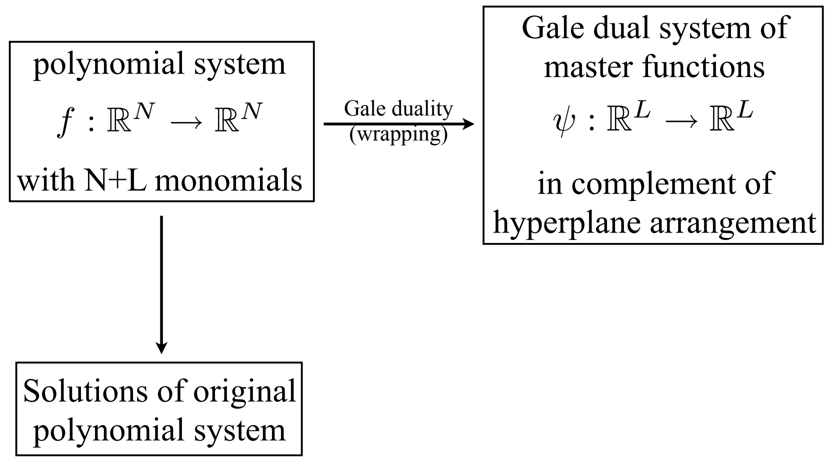
Given a curve C defined by a set of polynomials, the solutions on C of another polynomial are interspersed with solutions of an appropriately defined Jacobian determinant.
(see Khovanskii's *Fewnomials*).

Each idea has an important implication for us.

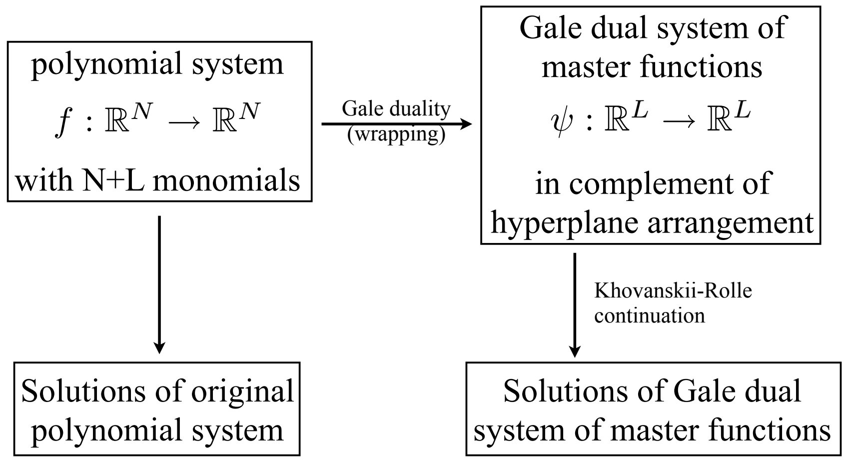
Background: Gale duality (high level)



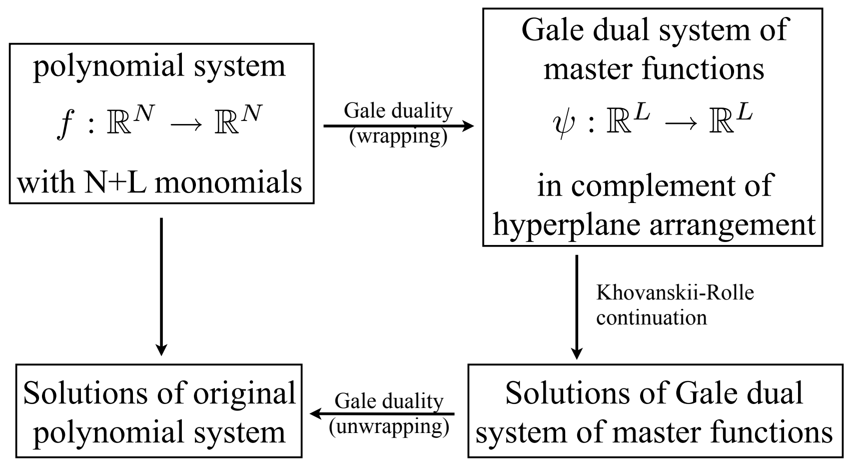
Background: Gale duality (high level)



Background: Gale duality (high level)



Background: Gale duality (high level)



Background: Gale duality (low level)

The details are picky but not impossible...see the paper (or I can show you on paper later).

Bottom line: We want to find the solutions of the master functions defined in the complement of a hyperplane arrangement.

Matt Niemerg and I are nearly done with software for both the wrapping and the unwrapping. We will release the code once we have finished and tested it.

Background: Khovanskii-Rolle theorem

Given master functions

$$\psi : \mathbb{R}^L \to \mathbb{R}^L$$

For j= L, L-1, ... 1, define:

$$J_j \ := \ \mathrm{Jac}(\psi_1, \ldots, \psi_j, \ J_{j+1}, \ldots, J_L)$$

and let

$$C_j \ := \ V(\psi_1, \ldots, \psi_{j-1}, \ J_{j+1}, \ldots, J_L),$$

curves in the complement of a hyperplane arrangement.

Khovanskii-Rolle says that solutions of ψ_j on C_j are separated by solutions of J_j.

Background: Bates-Bihan-Sottile proof

Thanks to Gale duality, to count the positive solutions of a system of polynomials, we can instead count the number of solutions of a system of master functions in the positive chamber.

Consequence of Khovanskii-Rolle:

$$|V(\psi_1,\ldots,\psi_L)| \;\leq\; \flat(C_L)+\cdots+\flat(C_1) \;+\; |V(J_1,\ldots,J_L)|$$

where $\flat(C)$ is the number of unbounded components of C.

Also (from Bihan-Sottile):

1. $|V(J_1,\ldots,J_L)| \leq 2^{\binom{L}{2}}N^L$
2. $\flat(C_j) \;\leq\; \frac{1}{2}2^{\binom{L-j}{2}}n^{N-L}\binom{N+L+1}{j}\cdot 2^j \;\leq\; 2^{\binom{k}{2}}n^k\cdot\frac{2^{2j-1}}{j!}$

Putting this together gives the latest bound.

Gameplan

1. Background on proof of Bihan-Sottile bound
2. **Proof $\longrightarrow$ Algorithm**
3. Example (pretty pictures)
4. A word about complexity
5. Further plans

Proof $\longrightarrow$ Algorithm

Where is the algorithm?

Rather than counting arcs and intersections, we move along them and watch for solutions:

- Solve $J_L, J_{L-1}, \ldots, J_1$ and find all points where the arcs given by vanishing of all J_j except J_1 intersect the boundary of the chamber.

- Traverse each arc twice, looking for solutions of ψ_1 (or the current master function of interest):

  - Move one direction from boundary points
  - Move two directions from interior points
  - Security: Know how many times we reach each point
  - Move on to the next master function and J_j

Gameplan

1. Background on proof of Bihan-Sottile bound
2. Proof $\longrightarrow$ Algorithm
3. **Example (pretty pictures)**
4. A word about complexity
5. Further plans

An example

- The initial (Laurent) polynomials (after Gaussian elimination):

$$cd = \tfrac{1}{2}be^2 + 2a^{-1}b^{-1}e - 1 \qquad cd^{-1}e^{-1} = \tfrac{1}{2}(1 + \tfrac{1}{4}be^2 - a^{-1}b^{-1}e)$$

$$bc^{-1}e^{-2} = \tfrac{1}{4}(6 - \tfrac{1}{4}be^2 - 3a^{-1}b^{-1}e) \qquad bc^{-2}e = \tfrac{1}{2}(8 - \tfrac{3}{4}be^2 - 2a^{-1}b^{-1}e)$$

$$ab^{-1} = 3 - \tfrac{1}{2}be^2 + a^{-1}b^{-1}e \ ,$$

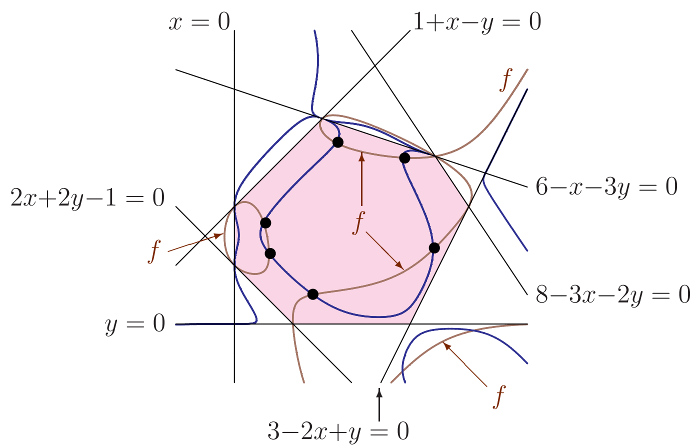
  Notice that N = 5 and L = 2.

- Master functions (just two, in two variables):

$$f := (2x+2y-1)(1+x-y)(8-3x-2y)^2 - 8yx^2(6-x-3y)^2(3-2x+y)\,,$$

$$g := y(2x+2y-1)^6(1+x-y)^6(8-3x-2y)^7(3-2x+y) - 32768\,x^3(6-x-3y)^2\,.$$

Solutions of the master functions:



Thanks to Frank for all the images from now on!

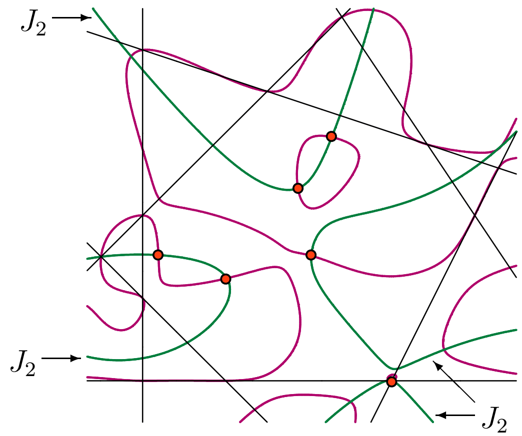
Preparation for KR continuation

- Form the Jacobian determinants
  $J_2 = J(f,g)$,
  $J_1 = J(f,J_2)$.

$$\begin{aligned} J_2 &= -168x^5 - 1376x^4y + 480x^3y^2 - 536x^2y^3 - 1096xy^4 + 456y^5 + 1666x^4 + 2826x^3y \\ &\quad +3098x^2y^2 + 6904xy^3 - 1638y^4 - 3485x^3 - 3721x^2y - 15318xy^2 - 1836y^3 \\ &\quad +1854x^2 + 8442xy + 9486y^2 - 192x - 6540y + 720\,. \\ J_1 &= 10080x^{10} - 168192x^9y - 611328x^8y^2 - \quad \cdots \quad + 27648x + 2825280y\,. \end{aligned}$$

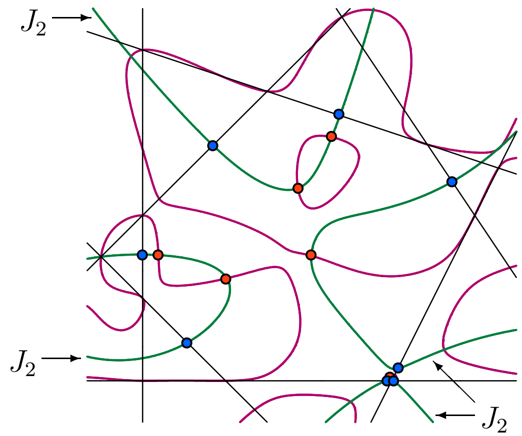
- Key properties:
  1. The solutions of $f = g = 0$ are separated by those of $f = J_2 = 0$ on the curve $f = 0$.
  2. The solutions of $f = J_2 = 0$ are separated by those of $J_1 = J_2 = 0$ on the curve $J_2 = 0$.

The curves for J_1 and J_2:



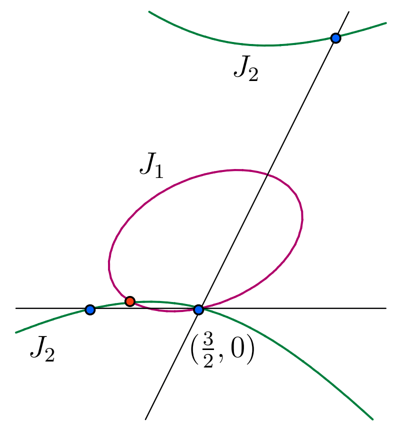
We find the solutions of $J_1 = J_2 = 0$ with continuation.
There are 6 of these.

The curves for J_1 and J_2:



We also find all points where J_2 meets the boundary.
There are 8 of these points.

The bottom right corner:



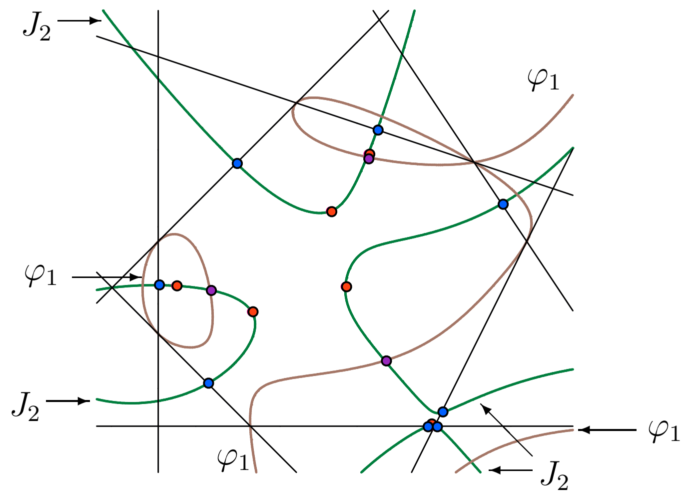
Step 1

Since any two solutions of $f = J_2 = 0$ along the curve $J_2 = 0$ are separated by solutions of $J_1 = J_2 = 0$, we will find all solutions of $f = J_2 = 0$ by tracking

1. each way from the solutions of $J_1 = J_2 = 0$ AND

2. into the polytope from the points at which J_2 reaches the boundary.

Moving from red and blue to purple along green curve
(replacing a Jacobian with a master function):



Safety from the Khovanskii-Rolle theorem

- Since any two solutions of $f = J_2 = 0$ along the curve $J_2 = 0$ are separated by solutions of $J_1 = J_2 = 0$:

  1. we will find all solutions of $f = J_2 = 0$ exactly twice, and

  2. we can watch the behavior along each arc we trace to help make sure that each arc is traced the appropriate number of times.

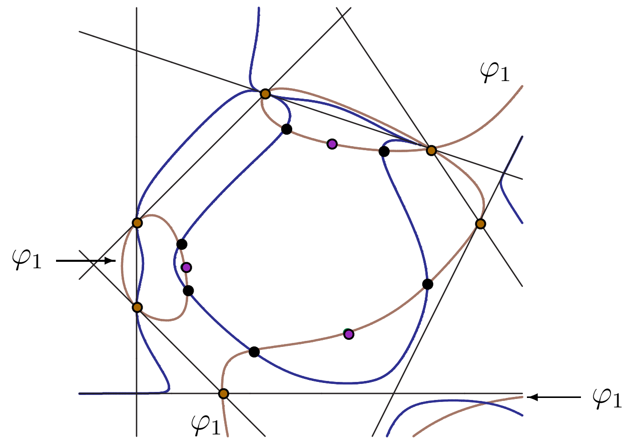
- Who cares? We do, because we don't have the usual guarantees of homotopy continuation.

Step 2 (last step for this example)

Now we move along the curve $f = 0$
- in two directions from each solution of $f = J_2 = 0$ and
- in one direction from each point at which f reaches the boundary

to find all points at which **f = g = 0** (the solutions we wanted in the first place!).

Moving from purple and brown to **black** along brown:



Gameplan

1. Background on proof of Bihan-Sottile bound
2. Proof $\longrightarrow$ Algorithm
3. Example (pretty pictures)
4. **A word about complexity**
5. Further plans

Complexity

Question: What is the complexity of this method?

Complexity

Question: What is the complexity of this method?

Answer: Unknown.

Complexity

Question: What is the complexity of this method?

Answer: Unknown.

However, if you count the following and add:

- Upper bound on # arcs to follow (often fewer),
- # polynomial systems to solve, and
- Bézout number of each,

then the total number of paths/arcs to follow is less than twice the Bihan-Sottile bound!

Complexity

Question: What is the complexity of this method?

Answer: Unknown.

However, if you count the following and add:

- Upper bound on # arcs to follow (often fewer),
- # polynomial systems to solve, and
- Bézout number of each,

then the total number of paths/arcs to follow is less than twice the Bihan-Sottile bound!

(The complexity of curve-tracing/path-following is unknown in general.)

Gameplan

1. Background on proof of Bihan-Sottile bound
2. Proof $\longrightarrow$ Algorithm
3. Example (pretty pictures)
4. A word about complexity
5. **Further plans**

Further plans

- Generalize algorithm to $L > 2$.
- Increase numerical security.
- Extend software (KhRo - see Frank's website) to $L > 2$.
- Add Gale duality pre- and post-processing to KhRo.
- Parallelize.
- Applications.

Thanks!

For more details, please see "Khovanskii-Rolle continuation for real solutions," arXiv:0908.4579

(Ask me about SI(AG)^2 if you don't know about it.)