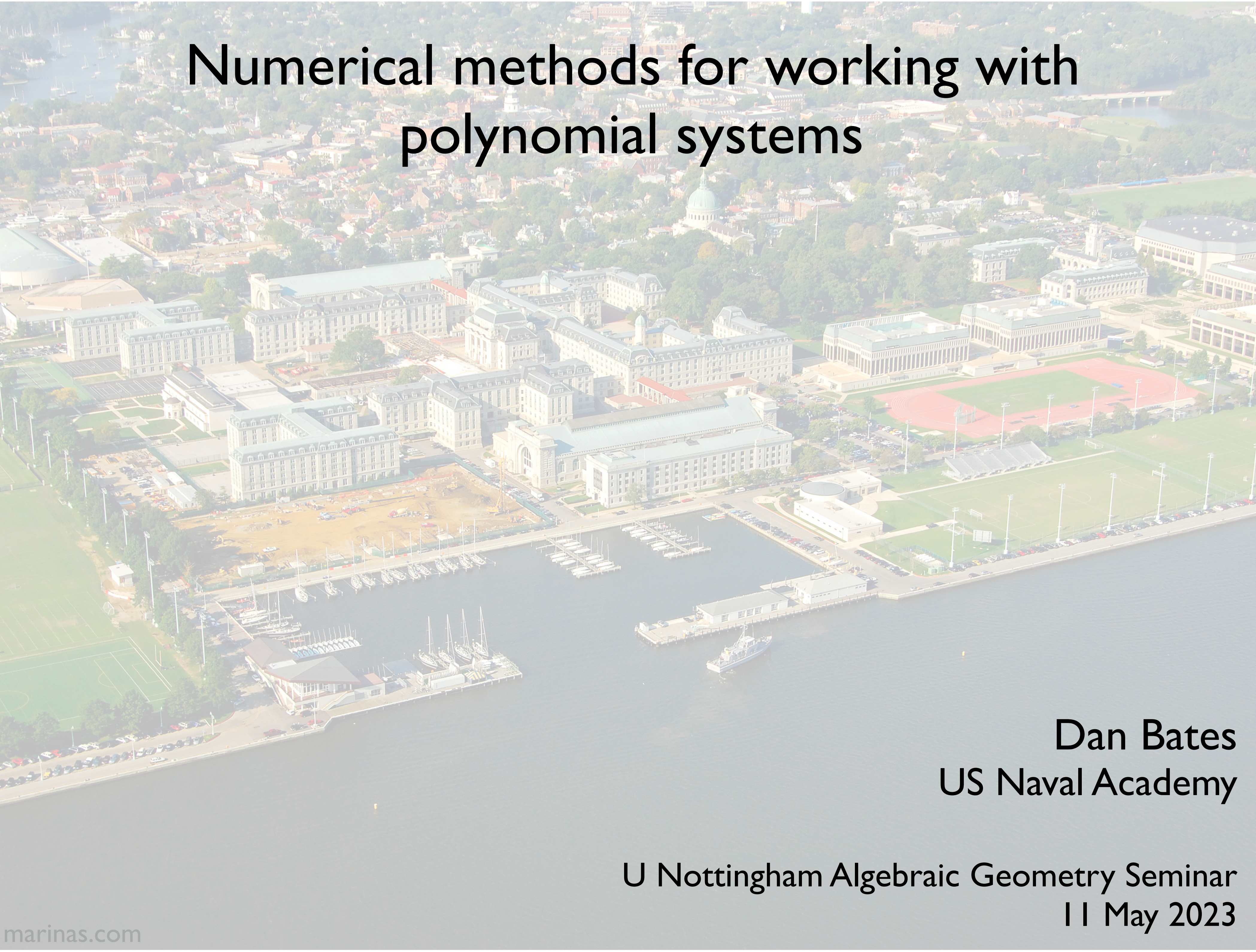
Numerical methods for working with polynomial systems

Dan Bates
US Naval Academy

U Nottingham Algebraic Geometry Seminar
11 May 2023



Motivation

Today's goal:
Introduce the various tools of numerical algebraic geometry (NAG), in case they might be helpful to you.

Motivation

<u>Today's goal</u>:
Introduce the various tools of numerical algebraic geometry (NAG), in case they might be helpful to you.

<u>First example</u>:
What are the solutions of the following polynomial system?

$$f = \begin{bmatrix} (y - x^2)(x^2 + y^2 + z^2 - 1)(x - 2) \\ (z - x^3)(x^2 + y^2 + z^2 - 1)(y - 2) \\ (z - x^3)(y - x^2)(x^2 + y^2 + z^2 - 1)(z - 2) \end{bmatrix}$$

Motivation

<u>First example</u>:
What are the solutions of the following polynomial system?

$$f = \begin{bmatrix} (y-x^2)(x^2+y^2+z^2-1)(x-2) \\ (z-x^3)(x^2+y^2+z^2-1)(y-2) \\ (z-x^3)(y-x^2)(x^2+y^2+z^2-1)(z-2) \end{bmatrix}$$

<u>Some possible answers</u>:
* Six irreducible components: a quadric surface (including a real sphere), a cubic curve, three lines, and a point.

* The varieties corresponding to six ideals in the prime decomposition of the ideal generated by these three polynomials.

Motivation

<u>The numerical algebraic geometry answer</u>

* The dimension and degree of each component, along with numerical approximations (to any accuracy) of points on each component.

Two points on the surface (100 digits):

```
(-0.4469187417023529140259318861084152642522761203775702508795613274333890441038
064784525828205296152151501603 8e0-0.6102710670219605508577582985863782755529962 3
1221157450657817129045917695842779305557510524563563326762211 05e0 i,
0.1301867835911536182463155717071071723530633705431267982912470403342211034604 98
43791220186062839003161843226e1-0.9797631891954616377716585987110315815817548375
4924382806366007477291012750177378608011376259729818723123248e-1 i,
0.1956248225827507852834244008952904191834615133805478985660019153364271707605 54
27892290316223611073692908373e0-0.7421827003933645822959459064358277029388028193
5139515678711527150051616817182335240552295390318031311036028e0 i)

(0.1869104826863862563031381694453612081139532180927845636668447908184320231372 7
7999536113201269428190397089 96e1-0.2264908452044595386090414954097308276341709 22
46720177262052807338520510389147232300490374986170415492740129e1 i,
0.1570509298979101749811443110979917681287422340187152550478924919024137351302 86
98617511572403647299128414405e1+0.1152332076074893022864178090389066888656393959
84765189711132427027040150157537288281433535574791985072789 26e1 i,
-0.1812583721926637539333874070357534929730804688597007838924989984346563572626 4
53791368921621107564625499634 0e1-0.1337098557057378271689106517659753437912289 21
48577601135972604026165978821952275526373960068495495133838546e1 i)
```

```
************* Witness Set Decomposition *************

| dimension | components | classified | unclassified
-----------------------------------------------------
|   2       |   1        |   2        |  0
|   1       |   4        |   6        |  0
|   0       |   1        |   1        |  0
-----------------------------------------------------

************** Decomposition by Degree **************

Dimension 2: 1 classified component
-----------------------------------------------------
   degree 2: 1 component

Dimension 1: 4 classified components
-----------------------------------------------------
   degree 1: 3 components
   degree 3: 1 component

Dimension 0: 1 classified component
-----------------------------------------------------
   degree 1: 1 component

*****************************************************
```

Game Plan

1. Motivation + example
2. Numerical algebraic geometry tools
    A. Homotopy continuation
    B. Parameter homotopies
    C. Numerical irreducible decomposition (NID)
    D. Software options
3. Various applications

Homotopy continuation

$$f(x,y) = \begin{bmatrix} x^2(y+1) \\ (x-3)(y-1) \end{bmatrix}$$ **(target system)**

Homotopy continuation

$$f(x, y) = \begin{bmatrix} x^2(y+1) \\ (x-3)(y-1) \end{bmatrix}$$ (target system)

(start system) $$g(x, y) = \begin{bmatrix} x^3 - 1 \\ y^2 - 1 \end{bmatrix}$$

Homotopy continuation

$$f(x, y) = \begin{bmatrix} x^2(y+1) \\ (x-3)(y-1) \end{bmatrix}$$ (target system)

$\uparrow$ t = 0

$$H(x, y; t) = \begin{bmatrix} x^2(y+1)(1-t) + t(x^3 - 1) \\ (x-3)(y-1)(1-t) + t(y^2 - 1) \end{bmatrix}$$ (homotopy)

$\downarrow$ t = 1

(start system) $$g(x, y) = \begin{bmatrix} x^3 - 1 \\ y^2 - 1 \end{bmatrix}$$

Homotopy continuation

Given polynomial system $f : \mathbb{C}^N \to \mathbb{C}^N$ (the **target system**) homotopy continuation is a 3-step process:

1. Choose and solve a polynomial system $g : \mathbb{C}^N \to \mathbb{C}^N$ (the **start system**) based on characteristics of $f(z)$ but relatively easy to solve.

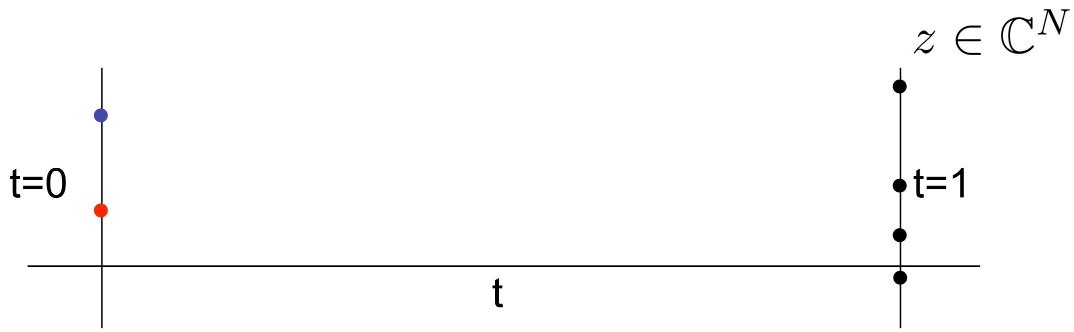
2. Form the **homotopy** $H : \mathbb{C}^N \times \mathbb{C} \to \mathbb{C}^N$ given by

$$H(z,t) = f(z) \cdot (1-t) + g(z) \cdot t$$

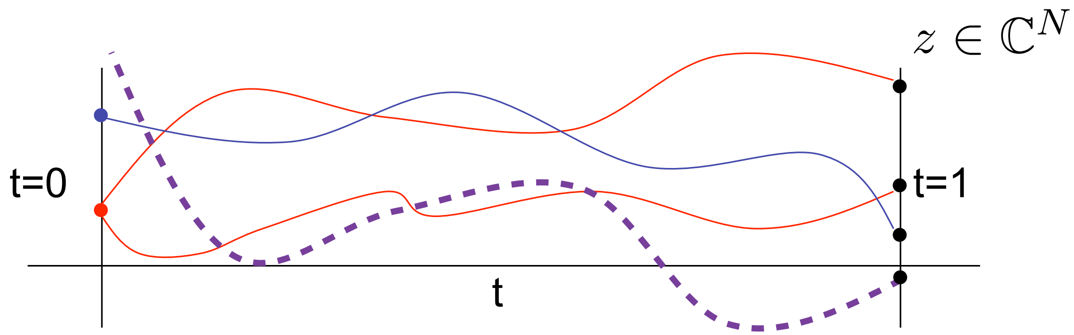
so that $H(z,1) = g(z)$ and $H(z,0) = f(z)$.

3. Use numerical **predictor-corrector methods** to follow the solutions as t marches from 1 to 0, one solution at a time.

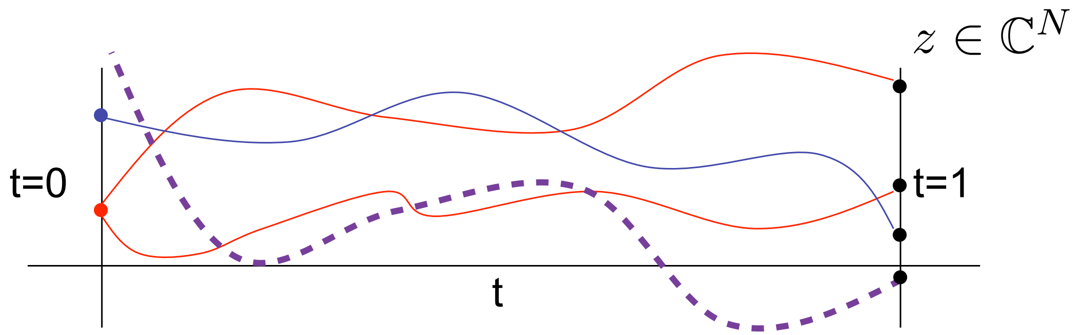
Homotopy continuation

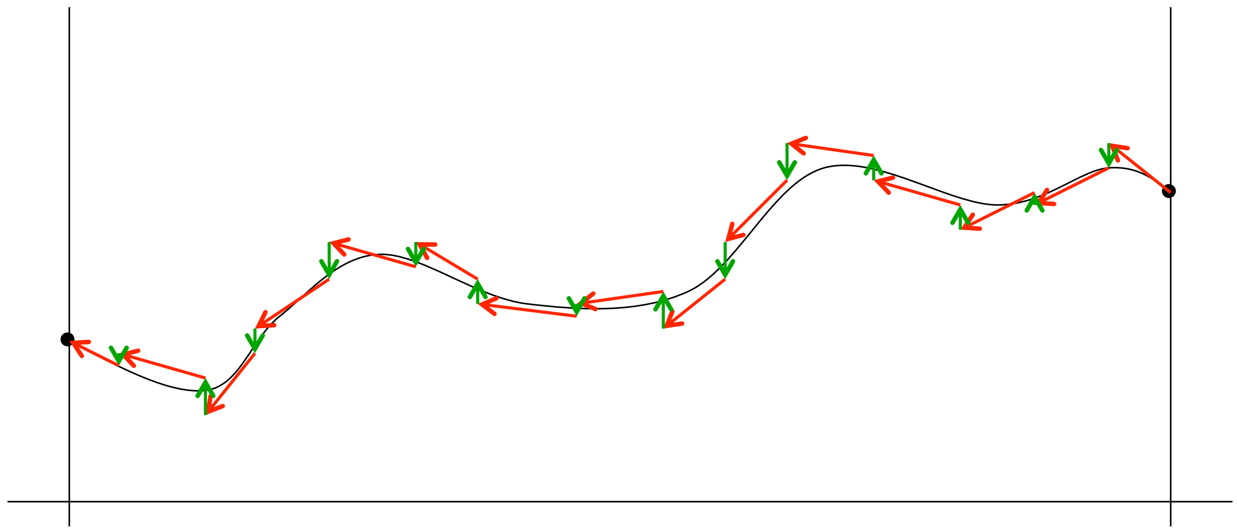


Homotopy continuation



Homotopy continuation





Homotopy continuation

* Skipping many details (predictor/corrector choices, non-square systems, polyhedral methods, adaptive steplength, adaptive precision, safety checks, endgames, certification, etc.)

If you want more details:

DB-Hauenstein-Sommese-Wampler, *Numerically solving polynomial systems with Bertini.* SIAM, 2013.

Homotopy continuation

* Skipping many details (predictor/corrector choices, non-square systems, polyhedral methods, adaptive steplength, adaptive precision, safety checks, endgames, certification, etc.)

If you want more details:
DB-Hauenstein-Sommese-Wampler, *Numerically solving polynomial systems with Bertini.* SIAM, 2013.

* Bottom line:
Guaranteed (with probability one, modulo numerical issues) to find approximations to all solutions that are isolated over the complex numbers.

Homotopy continuation

* Skipping many details (predictor/corrector choices, non-square systems, polyhedral methods, adaptive steplength, adaptive precision, safety checks, endgames, certification, etc.)

If you want more details:
DB-Hauenstein-Sommese-Wampler, *Numerically solving polynomial systems with Bertini.* SIAM, 2013.

* Bottom line:
Guaranteed (with probability one, modulo numerical issues) to find approximations to all solutions that are isolated over the complex numbers.

* Fun tangent: What if you run homotopy continuation on a system with positive-dimensional components?
DB-Eklund-Hauenstein-Peterson, Excess intersections and numerical irreducible decompositions. *23rd International Symposium on Symbolic and Numerical Algorithms for Scientific Computing (SYNASC)*, 2021

Game Plan

1. Motivation + example
2. Numerical algebraic geometry tools
   A. Homotopy continuation
   B. Parameter homotopies
   C. Numerical irreducible decomposition (NID)
   D. Software options
3. Various applications

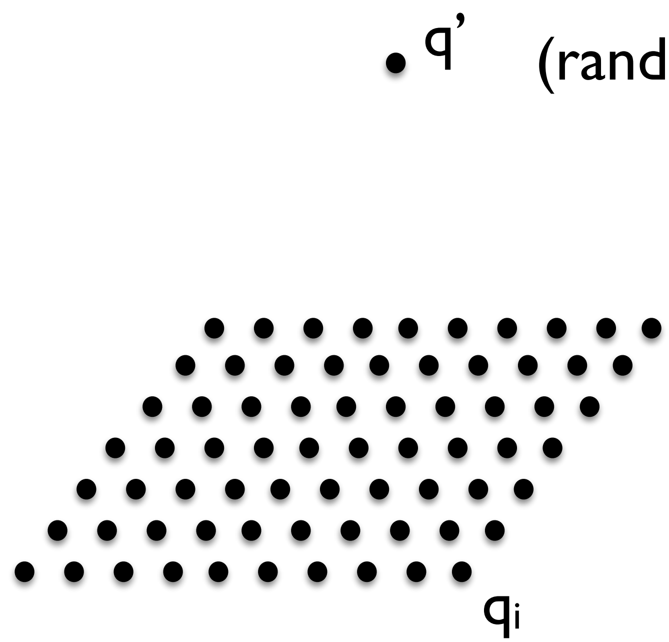
Parameter homotopies

Main point: We can be particularly efficient if we need to solve many nearly identical polynomial systems (same support, different coefficients).

Parameter homotopies

Main point: We can be particularly efficient if we need to solve many nearly identical polynomial systems (same support, different coefficients).

Idea:



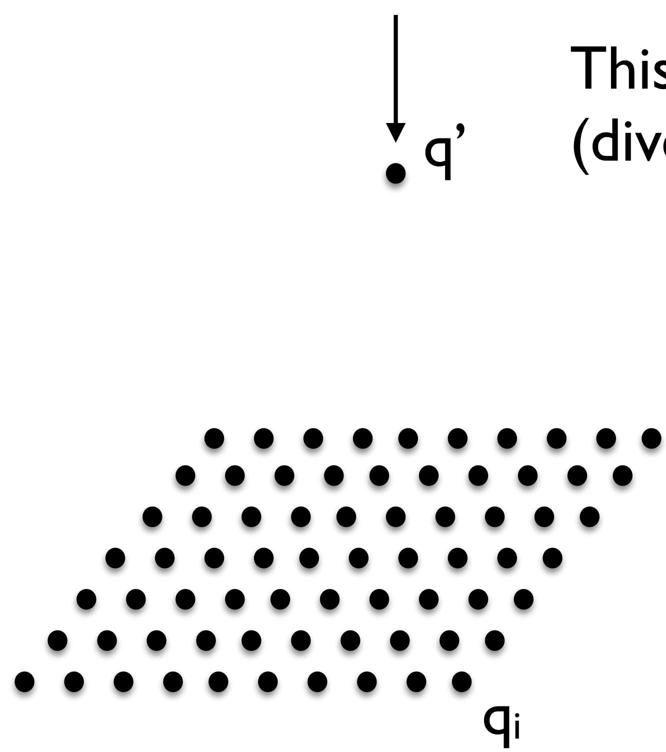
(random complex parameter values)

(many points in some parameter spsace)

Parameter homotopies

Main point: We can be particularly efficient if we need to solve many nearly identical polynomial systems (same support, different coefficients).

Idea:

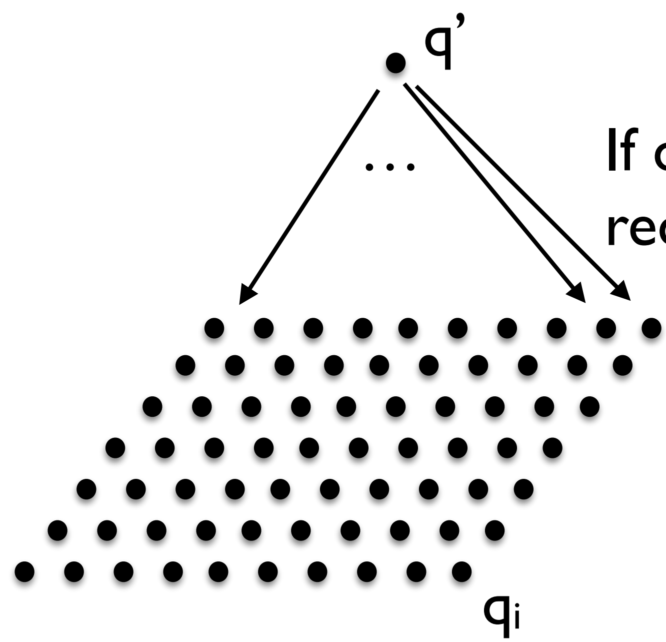


This might require wasted effort (divergent paths).

Parameter homotopies

Main point: We can be particularly efficient if we need to solve many nearly identical polynomial systems (same support, different coefficients).

Idea:

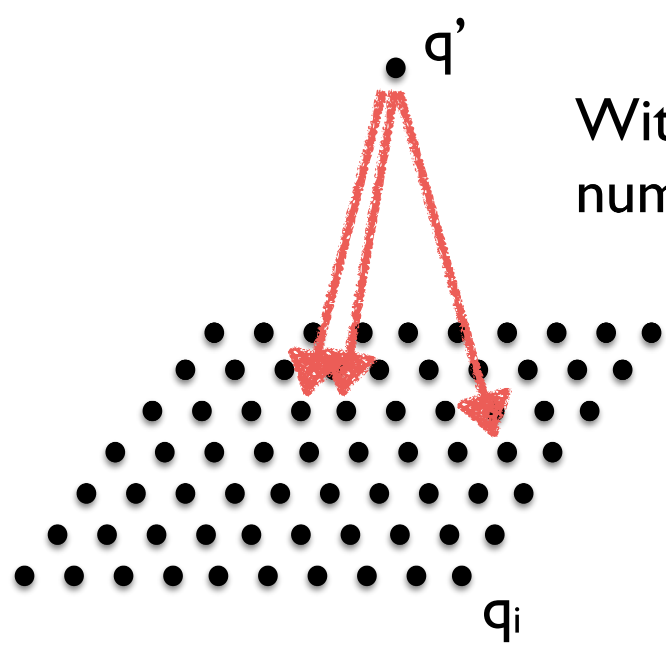


If q’ is generic, these should require few wasted paths.

Parameter homotopies

Main point: We can be particularly efficient if we need to solve many nearly identical polynomial systems (same support, different coefficients).

Idea:

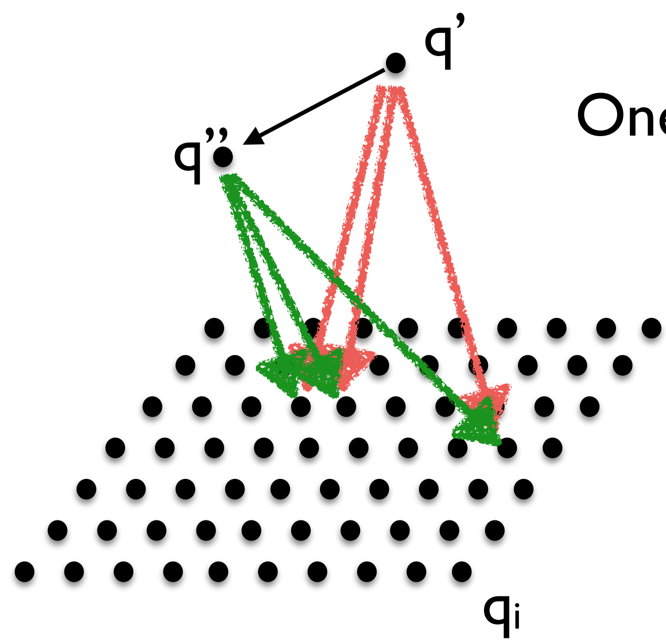


With many homotopies, numerical issues are more likely.

Parameter homotopies

Main point: We can be particularly efficient if we need to solve many nearly identical polynomial systems (same support, different coefficients).

Idea:



One option: try another angle.

Game Plan

1. Motivation + example
2. Numerical algebraic geometry tools
    A. Homotopy continuation
    B. Parameter homotopies
    C. **Numerical irreducible decomposition (NID)**
    D. Software options
3. Various applications

Numerical Irreducible Decomposition

$$\begin{bmatrix} (y-x^2)(x^2+y^2+z^2-1)(x-2) \\ (z-x^3)(x^2+y^2+z^2-1)(y-2) \\ (z-x^3)(y-x^2)(x^2+y^2+z^2-1)(z-2) \end{bmatrix}$$

<u>Solutions</u>:

<u>Dimension 2</u>: One surface
<u>Dimension 1</u>: Three lines and one cubic curve
<u>Dimension 0</u>: One point

We want to find some "witness points" on each of these sets.

Finding positive-dimensional solution sets

<u>Recall:</u> $Z = \mathcal{V}(f) = \bigcup_{i=0}^{D} Z_i = \bigcup_{i=0}^{D} \bigcup_{j \in \Lambda_i} Z_{i,j}$, where:

D is the dimension of Z,
i cycles through possible dimensions of irreducible components,
j is an index within dimension i, and the
$Z_{i,j}$ are the irreducible components.

(This is the irreducible decomposition of Z.)

<u>For each positive-dimensional irreducible component</u>, $Z_{i,j}$, we aim to find numerical approximations to *some number* of generic points on $Z_{i,j}$.

Finding positive-dimensional solution sets

Key fact: Given irreducible component $Z_{i,j}$ of dimension i, for almost every choice of linear space L of codimension i, $Z_{i,j}$ intersects L in a set of a particular number of points. That number is the **degree** of $Z_{i,j}$.

So, to find deg($Z_{i,j}$) points on $Z_{i,j}$, we can append i linears to f. We refer to this operation as **slicing**.

To find points on all components, we can just loop through all reasonable values of i.

Finding positive-dimensional solution sets

Problem 1: We could pick up points on higher-dimensional components.
Problem 2: We could find points on multiple i-dimensional components.

Example: Suppose there are two curves and a surface. When we slice for the curves, we will find points on both curves and also on the surface.

Solution 1: Start at the top dimension and work your way down. Use a **membership test** on points in lower dimensions to see if they sit on the higher-dimensional components already found.

Solution 2: Carry out an equidimensional decomposition, using **monodromy** and the **trace test**.

Finding positive-dimensional solution sets

In fact, there is a clever way to string the homotopies together, called a **cascade** of homotopies. (There are more recent approaches, too.)

All told, the goal is to have deg $Z_{i,j}$ **witness points** on each component $Z_{i,j}$, yielding witness point set

$$W_{i,j} = Z_{i,j} \cap L_i.$$

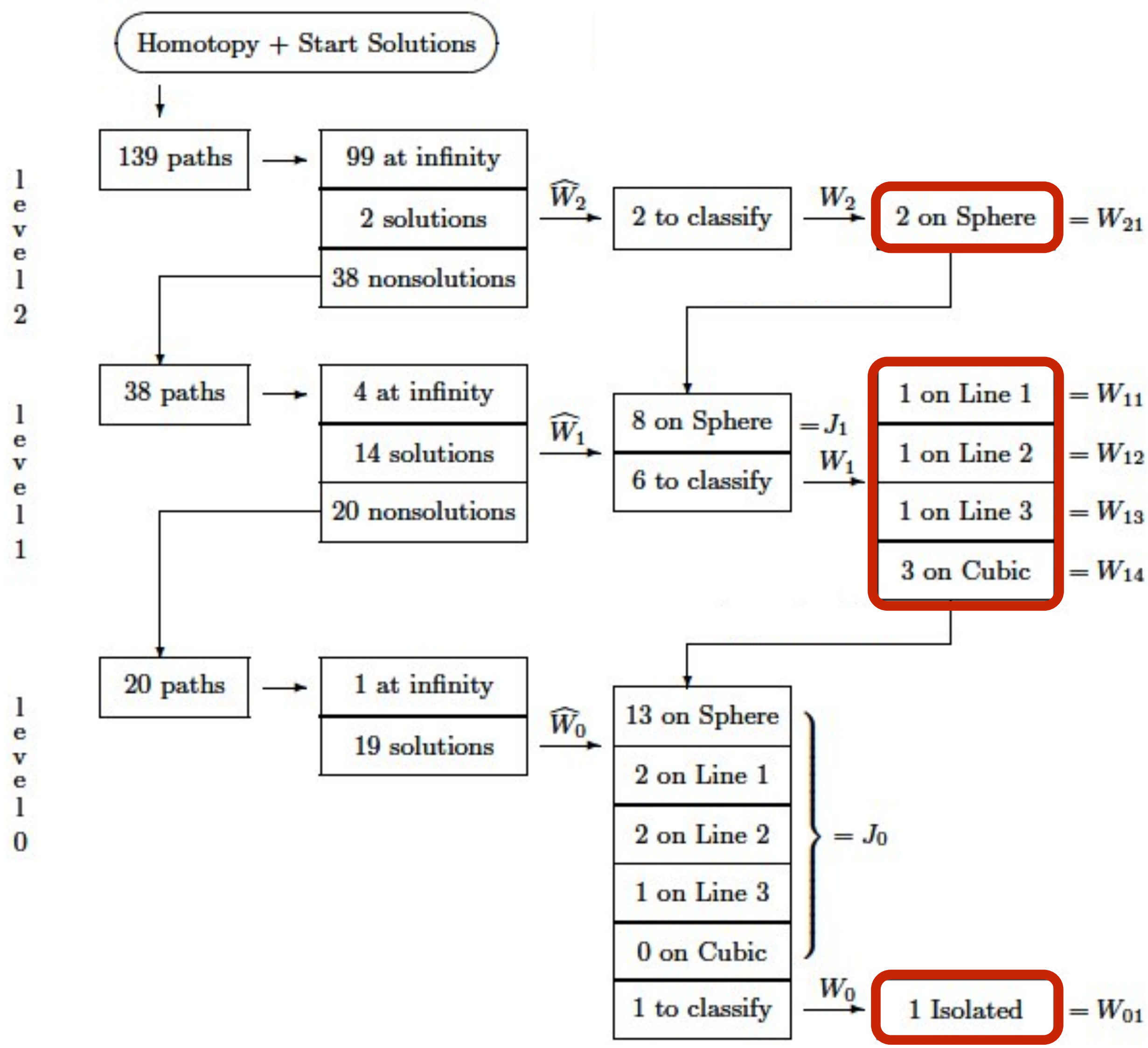
For each component, put the linear functions, the witness points, and the original functions together and you have a **witness set** for the component:

$$\mathcal{W}_{i,j} = (f, L_i, W_{i,j})$$

Then, the **numerical irreducible decomposition** is the union of all such sets for all irreducible components:

$$\mathcal{W} = \bigcup_{i=0}^{D} \mathcal{W}_i = \bigcup_{i=0}^{D} \bigcup_{j \in \Lambda_i} \mathcal{W}_{i,j}.$$

Numerical Irreducible Decomposition



Cascade of homotopies for computing the numerical irreducible decomposition of the illustrative example.

[Omitting many details!]

From Sommese-Verschelde-Wampler, *SIAM J. Num. An.* '01, the original cascade article.
See also Hauenstein-Sommese-Wampler, App. Math and Comp., '11 for regen cascade.

Numerical Irreducible Decomposition

Bertini Classic I/O

(input file)

```
Dans-MacBook-Pro-2:tmp_19may16 bates$ more input
CONFIG

TrackType: 1;

END;

INPUT

variable_group x,y,z;
function f,g,h;

f = (y-x^2)*(x^+y^2+z^2-1)*(x-2);
g = (z-x^3)*(x^+y^2+z^2-1)*(y-2);
h = (z-x^3)*(y-x^2)*(x^+y^2+z^2-1)*(z-2);

END;
```

Numerical Irreducible Decomposition

Bertini Classic I/O

(input file)

```
Dans-MacBook-Pro-2:tmp_19may16 bates$ more input
CONFIG

TrackType: 1;

END;

INPUT

variable_group x,y,z;
function f,g,h;

f = (y-x^2)*(x^+y^2+z^2-1)*(x-2);
g = (z-x^3)*(x^+y^2+z^2-1)*(y-2);
h = (z-x^3)*(y-x^2)*(x^+y^2+z^2-1)*(z-2);

END;
```

(screen output)

```
************* Witness Set Decomposition *************

| dimension | components | classified | unclassified
-----------------------------------------------------
|   2       |   1        |   2        |   0
|   1       |   4        |   6        |   0
|   0       |   1        |   1        |   0
-----------------------------------------------------

************** Decomposition by Degree **************

Dimension 2: 1 classified component
-----------------------------------------------------
   degree 2: 1 component

Dimension 1: 4 classified components
-----------------------------------------------------
   degree 1: 3 components
   degree 3: 1 component

Dimension 0: 1 classified component
-----------------------------------------------------
   degree 1: 1 component

*****************************************************
```

Game Plan

1. Motivation + example
2. Numerical algebraic geometry tools
   A. Homotopy continuation
   B. Parameter homotopies
   C. Numerical irreducible decomposition (NID)
   D. Software options
3. Various applications

Software

Don't worry — this has been implemented:

- **Bertini**: S Amethyst, DB, J Hauenstein, A Sommese, C Wampler
- HOM4PS-2/3: TY Li, TR Chen, et al.
- **HomotopyContinuation.jl**: P Breiding, S Timme
- **NAG4M2**: A Leykin
- Paramotopy: S Amethyst, DB, M Niemerg
- **PHCpack**: J Verschelde
- POLSYS GLP: L Watson, et al
- Others have come and gone, list may not be comprehensive.

Game Plan

1. Motivation + example
2. Numerical algebraic geometry tools
    A. Homotopy continuation / basic solving
    B. Parameter homotopies
    C. Numerical irreducible decomposition (NID)
    D. Software options
3. **Various applications**

Various applications

- Math:
  - Algebraic geometry (e.g., recovering exactness)
  - Dynamical systems
- Engineering:
  - Kinematics (e.g., mechanism design)
  - Optimal control
  - Geolocation
- Science:
  - Systems biology (e.g., chemical reaction networks)
  - String theory
  - Computer vision

(Others, too….)

Various applications

- Math:
  - Algebraic geometry (e.g., **recovering exactness**)
  - Dynamical systems

- Engineering:
  - Kinematics (e.g., mechanism design)
  - Optimal control
  - Geolocation

- Science:
  - Systems biology (e.g., chemical reaction networks)
  - String theory
  - Computer vision

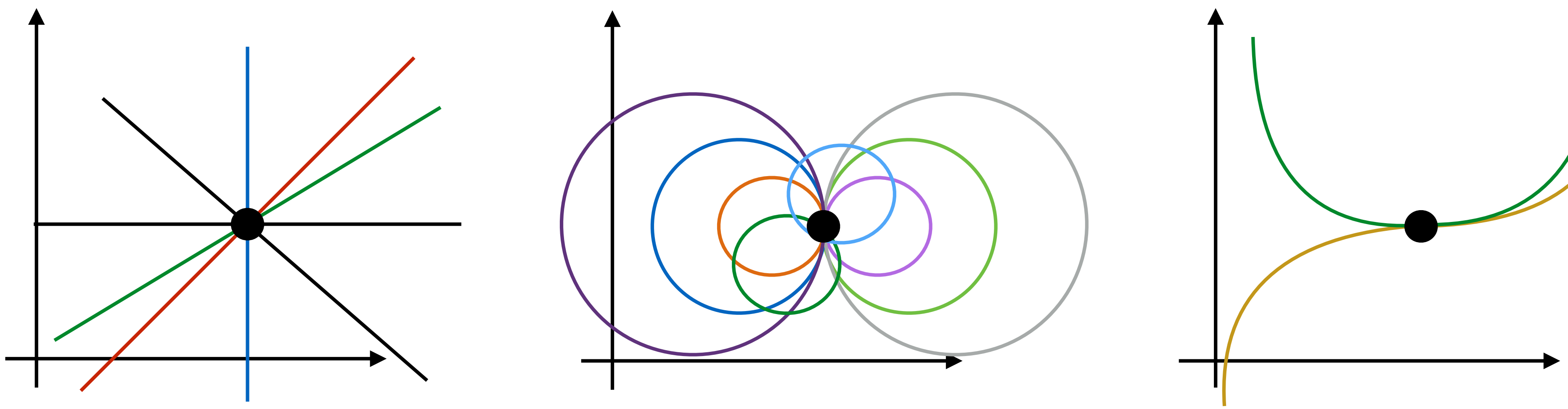
(Others, too….)

Recovering Exactness

Given the point,

$$x = 0.7949333985,$$
$$y = 0.6066967050,$$

what polynomials approximately vanish at this point?



Here's one I can compute: $x^2 + y^2 - 1$

Recovering Exactness

Given some point(s), find some polynomials of fixed degree d with "small" integer coefficients that approximately vanish at the point(s).

For example, if we fix degree 2 and can refine our point(s) to more digits of accuracy, what can we find?

[1] Recovering exact results from inexact numerical data in algebraic geometry. DB, J Hauenstein, T McCoy, C Peterson, A Sommese. *Experimental Mathematics* 22(1), 2013.

[2] Numerical irreducible decomposition over a number field. T. McCoy, C Peterson, A Sommese, *J. Algebra & its Applications* 17(10), 2018.

Recovering Exactness

Main idea from [1]:

Transform (x,y) into $V_2(x,y) = [1, x, y, x^2, xy, y^2]$.
(or homogeneous version)

Goal: Find $\mathbf{c} = [a_1, a_2, a_3, a_4, a_5, a_6] \in \mathbb{Z}^6$ so that
$\mathbf{c} \cdot V_2(x,y) = a_1 + a_2 x + a_3 y + a_4 x^2 + a_5 xy + a_6 y^2 \approx 0.$

Q: How?
A: Lattice basis reduction methods are very good at this, e.g., LLL and its variants.

Recovering Exactness

$x = 0.7949333985$
$y = 0.6066967050$

Degree 1: $[539, 47, -950] \cdot [1, x, y] \approx 0.$

Degree 2: $[-1, 0, 0, 1, 0, 1] \cdot [1, x, y, x^2, xy, y^2] \approx 0.$

Degree 3:
$[0, 0, -1, 0, 0, 0, 0, 1, 0, 1] \cdot [1, x, y, x^2, xy, y^2, x^3, x^2y, xy^2, y^3] \approx 0.$

Not terribly surprising: $-y + x^2y + y^3 = y(x^2 + y^2 - 1)$.

It's easy to *expand* $[-1, 0, 0, 1, 0, 1]$ up to degree 3 and check that $[0, 0, -1, 0, 0, 0, 0, 1, 0, 1]$ is very much in its span.

Recovering Exactness

$$J = < w^3xz^2 - w^3y^2z + 3wx^2yz^2 - 3wxy^3z + 7wxz^4 - 7wy^2z^3 + 2xy^4z - 2y^6,$$
$$w^4xz - w^3yz^2 + 3w^2x^2yz + 7w^2xz^3 + 2wxy^4 - 3wxy^2z^2 - 7wyz^4 - 2y^5z,$$
$$w^5yz - w^4yz^2 - w^4z^3 + 3w^3xy^2z + 7w^3yz^3 + w^3z^4 - 3w^2xy^2z^2 - 3w^2xyz^3$$
$$+ 2w^2y^5 - 7w^2yz^4 - 7w^2z^5 + 3wxyz^4 - 2wy^5z - 2wy^4z^2 + 7wz^6 + 2y^4z^3 >$$

Used Bertini to find a point on each irreducible component.
Going up to degree 7 with 238 digits of precision yielded:

Component 1: $zw^3 + 3zyxw + 7z^3w + 2y^4$

Component 2: $-zx + y^2$
$-xw + zy$
$-yw + z^2$

Component 3: x
y
$-w + z$

Same decomposition as computed via Gröbner bases in the paper where we found this example:

Direct methods for primary decomposition.
D. Eisenbud, C. Huneke, W. Vasconcelos.
Inventiones Mathematicae 110, 1992.

Various applications

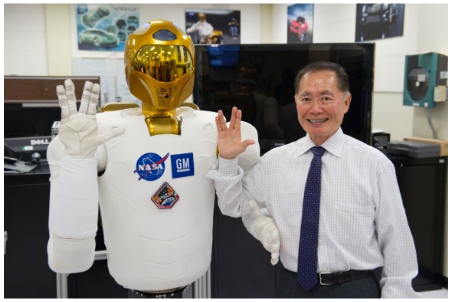
- Math:
  - Algebraic geometry (e.g., recovering exactness)
  - Dynamical systems
- Engineering:
  - Kinematics (e.g., **mechanism design**)
  - Optimal control
  - Geolocation
- Science:
  - Systems biology (e.g., chemical reaction networks)
  - String theory
  - Computer vision

(Others, too….)

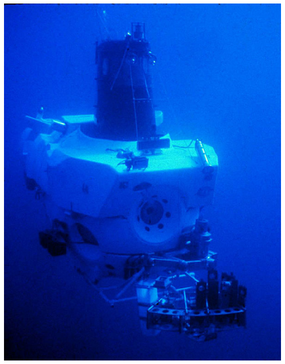
Kinematics

Kinematics: The study of mechanical linkages, ignoring forces.

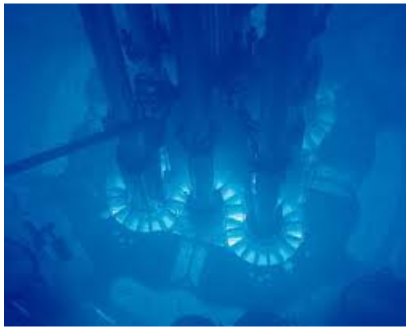
Mars rover



Robonaut 2
(GM/NASA)



Submarine

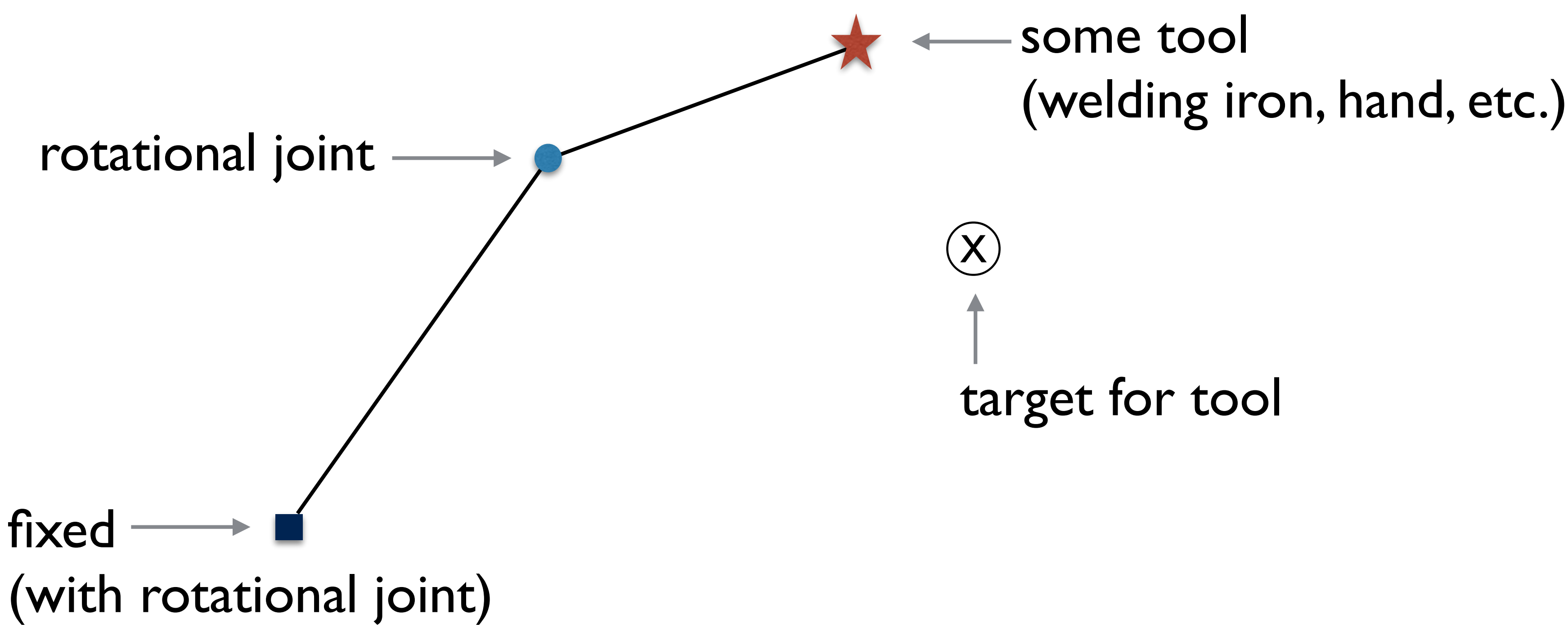


Spent fuel rods

All of these machines involve robotic arms. Other machines don't:

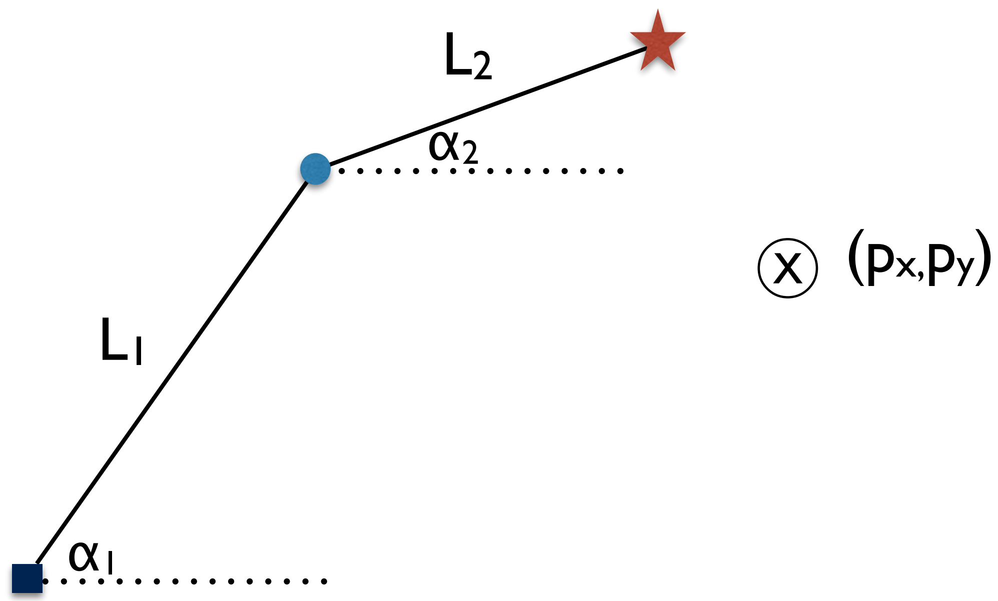
Kinematics

A 2R (2 links with a rotational joint) planar linkage:



Kinematics

Here's some notation:



Kinematics

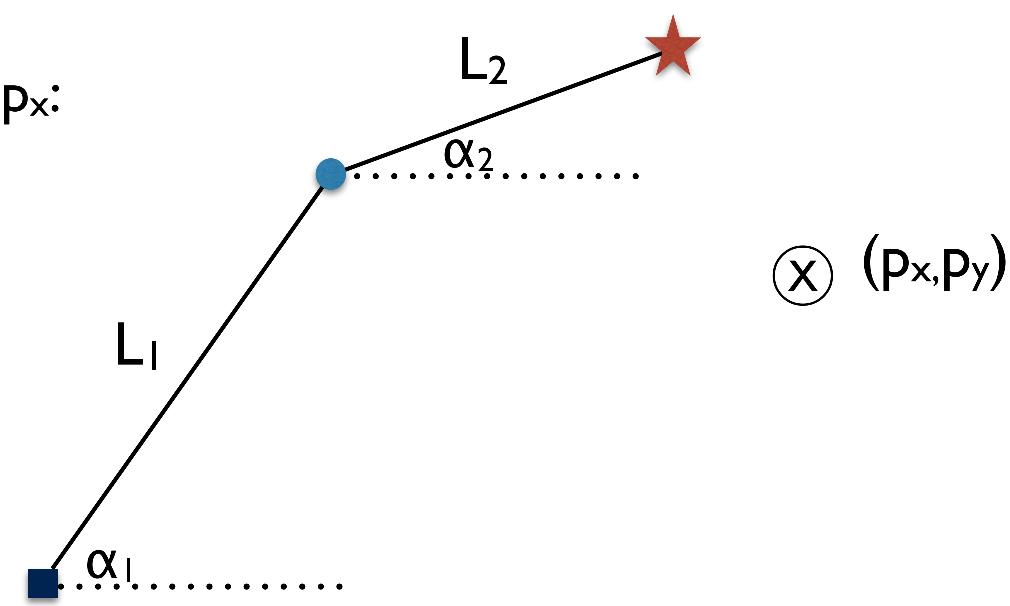
The Pythagorean theorem gives us equations to find the angles, given the target:

To get the x-coordinate of the tool to p_x:

$L_1 \cos(\alpha_1) + L_2 \cos(\alpha_2) = p_x$

Ditto for y:

$L_1 \sin(\alpha_1) + L_2 \sin(\alpha_2) = p_y$



Kinematics

Now we can just solve the 2x2 system (with L_1, L_2, p_x, p_y known numbers and variables α_1, α_2):

$$L_1 \cos(\alpha_1) + L_2 \cos(\alpha_2) = p_x$$
$$L_1 \sin(\alpha_1) + L_2 \sin(\alpha_2) = p_y$$

Problem: This isn't a polynomial system!

Trick: Rename the trig functions as variables and include trig identities to make a 4x4 *polynomial* system:

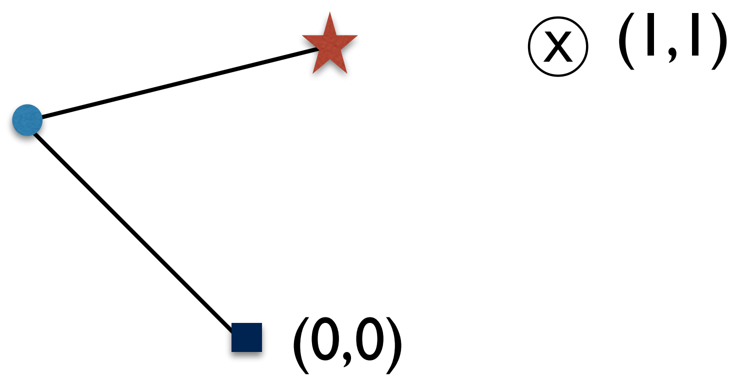
$$L_1 c_1 + L_2 c_2 - p_x = 0$$
$$L_1 s_1 + L_2 s_2 - p_y = 0$$
$$c_1^2 + s_1^2 - 1 = 0$$
$$c_2^2 + s_2^2 - 1 = 0$$

Kinematics

$L_1 c_1 + L_2 c_2 - p_x = 0$
$L_1 s_1 + L_2 s_2 - p_y = 0$
$c_1^2 + s_1^2 - 1 = 0$
$c_2^2 + s_2^2 - 1 = 0$

For example, if both links have length one and we want to reach (1,1), the system becomes:

$c_1 + c_2 - 1 = 0$
$s_1 + s_2 - 1 = 0$
$c_1^2 + s_1^2 - 1 = 0$
$c_2^2 + s_2^2 - 1 = 0$



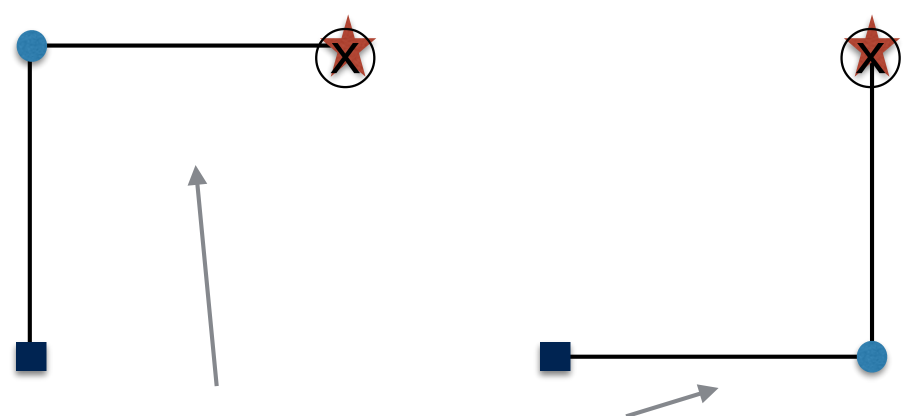
for which the solutions are $(c_1, s_1, c_2, s_2) = (0,1,1,0)$ and $(1,0,0,1)$.

Kinematics

$L_1\ c_1 + L_2\ c_2 - p_x = 0$
$L_1\ s_1 + L_2\ s_2 - p_y = 0$
$c_1^2 + s_1^2 - 1 = 0$
$c_2^2 + s_2^2 - 1 = 0$

For example, if both links have length one and we want to reach (1,1), the system becomes:

$c_1 + c_2 - 1 = 0$
$s_1 + s_2 - 1 = 0$
$c_1^2 + s_1^2 - 1 = 0$
$c_2^2 + s_2^2 - 1 = 0$



for which the solutions are $(c_1, s_1, c_2, s_2) = (0,1,1,0)$ and $(1,0,0,1)$.

Mt. Everest

9-point path synthesis problem: How many mechanisms of a particular type (two planar 2R linkages connected to form a triangle) pass through nine specified points?

"Alt's problem" from 1923: Became known as Mt. Everest of Kinematics.

Partial solutions in 1963 (Roth & Freudenstein), 1989 (Tsai & Lu).

First complete solution: 1992 (Wampler), using homotopy continuation.

Latest part of the story:
Brake, Hauenstein, Murray, Myszka, Wampler. *J. Mechanisms Robotics*, 2016.

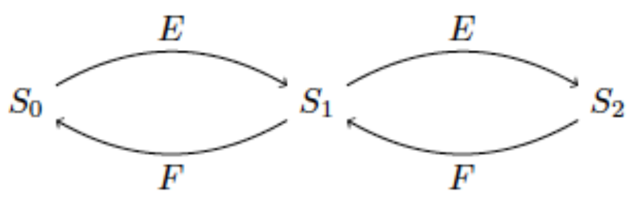
See also Sommese-Wampler, Algebraic Kinematics, *Acta Numerica* 20, 2011.

Various applications

- Math:
  - Algebraic geometry (e.g., recovering exactness)
  - Dynamical systems
- Engineering:
  - Kinematics (e.g., mechanism design)
  - Optimal control
  - Geolocation
- Science:
  - Systems biology (e.g., **chemical reaction networks**)
  - String theory
  - Computer vision

(Others, too….)

Chemical reaction networks



$$
\begin{aligned}
S_0 + E &\underset{k_2}{\overset{k_1}{\rightleftharpoons}} Y_1 \xrightarrow{k_3} Y_2 \underset{k_5}{\overset{k_4}{\rightleftharpoons}} S_1 + E \\
S_1 + E &\underset{k_7}{\overset{k_6}{\rightleftharpoons}} Y_3 \xrightarrow{k_8} Y_4 \underset{k_{10}}{\overset{k_9}{\rightleftharpoons}} S_2 + E \\
S_2 + F &\underset{k_{12}}{\overset{k_{11}}{\rightleftharpoons}} Z_4 \xrightarrow{k_{13}} Z_3 \underset{k_{15}}{\overset{k_{14}}{\rightleftharpoons}} S_1 + F \\
S_1 + F &\underset{k_{17}}{\overset{k_{16}}{\rightleftharpoons}} Z_2 \xrightarrow{k_{18}} Z_1 \underset{k_{20}}{\overset{k_{19}}{\rightleftharpoons}} S_0 + F.
\end{aligned}
$$

$$
\begin{aligned}
0 = {} & \alpha v^2 + \zeta u v + \beta \zeta^2 u^2 + (\alpha \epsilon_0 - \alpha \epsilon_0 \sigma - \alpha + \phi_1 \zeta)\, u v^2 + \left(\epsilon_1 \zeta - \epsilon_1 \zeta \sigma - \zeta + \beta \phi_2 \zeta^2\right) u^2 v \\
& + \left(\beta \epsilon_2 \zeta^2 - \beta \epsilon_2 \zeta^2 \sigma - \beta \zeta^2\right) u^3 + \alpha \phi_0 v^3 - (\alpha \epsilon_0 + \phi_1 \zeta)\, u^2 v^2 - \left(\epsilon_1 \zeta + \beta \phi_2 \zeta^2\right) u^3 v \\
& - \beta \epsilon_2 \zeta^2 u^4 - \alpha \phi_0 u v^3
\end{aligned}
$$

$$
\begin{aligned}
0 = {} & \alpha v^2 + \zeta u v + \beta \zeta^2 u^2 + (\alpha \phi_0 - \alpha \phi_0 \lambda - \alpha)\, v^3 + (\phi_1 \zeta - \phi_1 \zeta \lambda - \zeta + \alpha \epsilon_0)\, u v^2 \\
& + \left(\beta \phi_2 \zeta^2 - \beta \phi_2 \zeta^2 \lambda - \beta \zeta^2 + \epsilon_1 \zeta\right) u^2 v + \beta \epsilon_2 \zeta^2 u^3 - (\alpha \epsilon_0 + \phi_1 \zeta)\, u v^3 - \left(\epsilon_1 \zeta + \beta \phi_2 \zeta^2\right) u^2 v^2 \\
& - \beta \epsilon_2 \zeta^2 u^3 v - \alpha \phi_0 v^4.
\end{aligned}
$$

(project with C Nam, B Gyori, S Amethyst, J Gunawardena, started at AIM)

Chemical reaction networks

Basic stats

2 polynomials in 2 variables

8 non-dimensionalized parameters

(down from 13, thanks to Thomson-Gunawardena, 2009)

Generic root count: 7

Number of real solutions: 1 (monostable) or 3 (multistable)

Chemical reaction networks

<u>**Basic stats**</u>

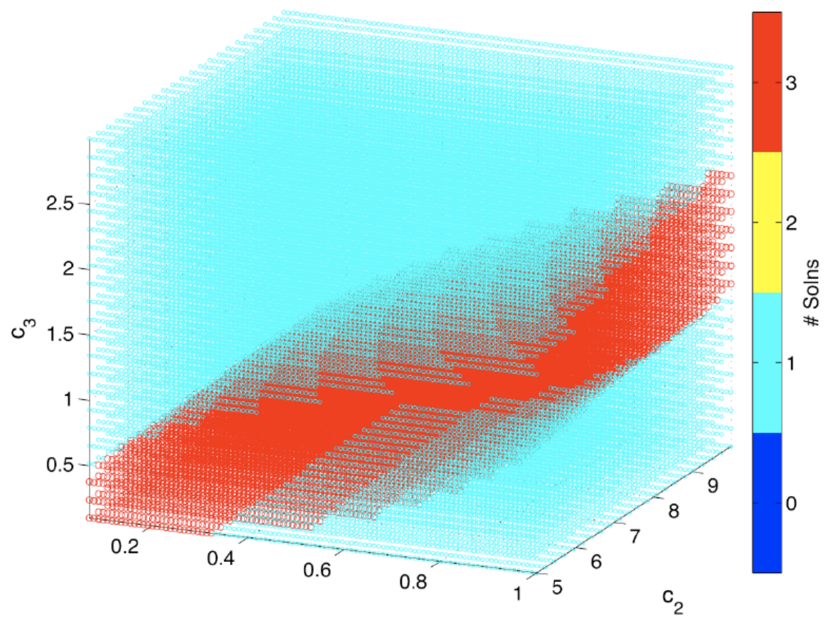
2 polynomials in 2 variables

8 non-dimensionalized parameters

(down from 13, thanks to Thomson-Gunawardena, 2009)

<u>Generic root count</u>: 7

<u>Number of real solutions</u>: 1 (monostable) or 3 (multistable)



Chemical reaction networks

Basic stats
2 polynomials in 2 variables
8 non-dimensionalized parameters
    (down from 13, thanks to Thomson-Gunawardena, 2009)
Generic root count: 7
Number of real solutions: 1 (monostable) or 3 (multistable)

Goal: Study the region of multistability:
- isolated points?
- measure zero?
- volume?
- change in volume as parameters change?
- connected?
- convex?

Chemical reaction networks

Goal: Study the region of multistability:
- isolated points: No! (tracked along a line segment)
- measure zero: No! (checked in nbhd of a point)
- volume: Depends
- change in volume as parameters change: Yes (see below)
- connected: No
- convex: No

> 100 million runs of Bertini →

σ	Number of uniform random samples	Number of multistable samples	Volume (%)
1	10^7	0	0
1.5	10^7	20	0.0002
2	10^7	2315	0.02315
2.5	$6 \cdot 10^6$	5330	0.08883
3	$4 \cdot 10^6$	6568	0.1642
4	$4 \cdot 10^6$	11373	0.2843
5	$4 \cdot 10^6$	14588	0.3647
7	$4 \cdot 10^6$	18451	0.4613
10	$4 \cdot 10^6$	21401	0.5350
20	10^5	627	0.627
50	10^5	672	0.672
100	10^5	683	0.683
200	10^5		
500	10^5		

> 1 billion in paper

Final thoughts

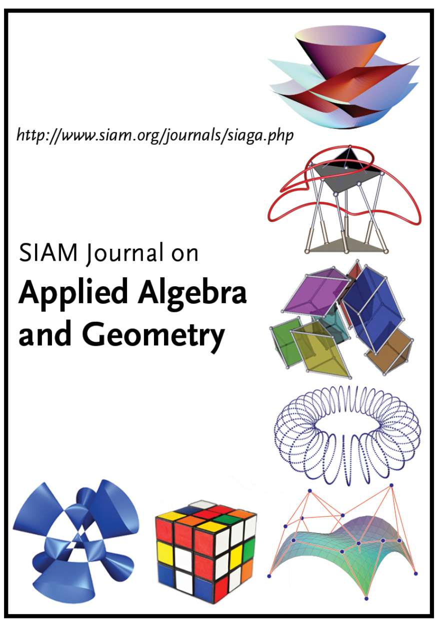
Compared to symbolic methods, numerical algebraic geometry scales better with dimension, worse with degree. For more, see:

DB, W Decker, J Hauenstein, C Peterson, G Pfister, FO Schreyer, A Sommese, C Wampler. Comparison of probabilistic algorithms for analyzing the components of an affine algebraic variety. *Applied Math and Computation* 231(C), 2014.

Numerical algebraic geometry might be useful for you at some point. I would be happy to help.

Final thoughts

- Open access is free for submissions NLT 31 Dec 23
- Focus is on computation and applications of algebraic or discrete structures, preferably with novel mathematics and novel algorithms…papers in numerical algebraic geometry could fit
- Open to special issues



- More established (first issue in 2017)
- Q1 journal
- Seeks papers that contribute both mathematically and within some application(s)

THANK YOU!