Recent work in Numerical Algebraic Geometry

Andrew Sommese

University of Notre Dame

www.nd.edu/~sommese

Numer. Alg. Geometry Collaborators

- Daniel Bates* (CSU)
- Jonathan Hauenstein* (Fields/Texas A&M)
- Chris Peterson (CSU)
- Charles Wampler* (GM R & D)

*Bertini Team

Biological Modeling Collaborators

- Wenrui Hao (Notre Dame)
- Jonathan Hauenstein (Fields/Texas A&M)
- Bei Hu (Notre Dame)
- Yuan Liu (Notre Dame)
- Timothy McCoy (Notre Dame)
- Yong-Tao Zhang (Notre Dame)

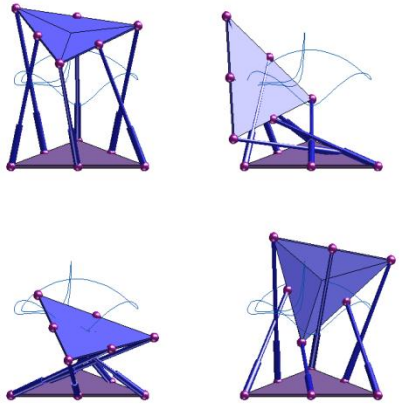
Overview

- Numerical Algebraic Geometry
  - Adaptive Multiprecision
  - Local Dimension Test
  - Regeneration
- Bertini
- Zebra Fish
- Tumor Growth

Numerical Algebraic Geometry

- **Goal:** To numerically manipulate algebraic sets
- **Technical Challenge:** To combine high performance numerics with algebraic geometry
- **Applications:**
  - Robotics and Mechanism Theory
  - Chemical Reactions including combustion
  - Computation of algebraic-geometric invariants
  - Solution of discretizations of nonlinear differential equations

Robotics/Mechanism Theory

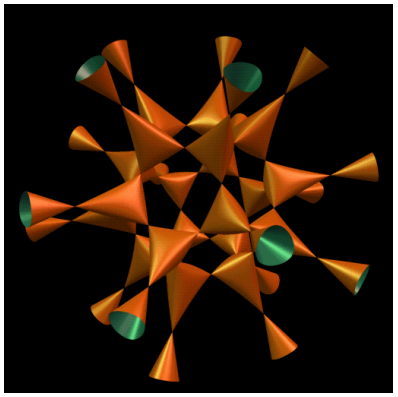


Combustion

$$
\begin{aligned}
O_2 &\rightleftharpoons 2O & k_1 X_{O_2} &= X_O^2 \\
H_2 &\rightleftharpoons 2H & k_2 X_{H_2} &= X_H^2 \\
N_2 &\rightleftharpoons 2N & k_3 X_{N_2} &= X_N^2 \\
CO_2 &\rightleftharpoons O + CO & k_4 X_{CO_2} &= X_O X_{CO} \\
OH &\rightleftharpoons O + H & k_5 X_{OH} &= X_O X_H \\
H_2O &\rightleftharpoons O + 2H & k_6 X_{H_2O} &= X_O X_H^2 \\
NO &\rightleftharpoons O + N & k_7 X_{NO} &= X_O X_N .
\end{aligned}
$$

There are four conservation equations:

$$
\begin{aligned}
T_H &= X_H + 2X_{H_2} + X_{OH} + 2X_{H_2O} \\
T_C &= X_{CO} + X_{CO_2} \\
T_O &= X_O + X_{CO} + 2X_{O_2} + 2X_{CO_2} + X_{OH} + X_{H_2O} + X_{NO} \\
T_N &= X_N + 2X_{N_2} + X_{NO}
\end{aligned}
$$



graphics on right from Sommese-Wampler Book

General References up to end of 2004

- A.J. Sommese and C.W. Wampler, *Numerical solution of systems of polynomials arising in engineering and science*, (2005), World Scientific Press.
- T.Y. Li, Numerical solution of polynomial systems by homotopy continuation methods, in *Handbook of Numerical Analysis*, Volume XI, 209-304, North-Holland, 2003.

Three Recent Articles

- D.J. Bates, J.D. Hauenstein, A.J. Sommese, and C.W. Wampler, Adaptive multiprecision path tracking, SIAM Journal on Numerical Analysis 46 (2008) 722-746.

- D.J. Bates, J.D. Hauenstein, C. Peterson, and A.J. Sommese, A numerical local dimension test for points on the solution set of a system of polynomial equations, SIAM Journal on Numerical Analysis, 47 (2009), 3608-3623.

- J.D. Hauenstein, A.J. Sommese, and C.W. Wampler, Regeneration homotopies for solving systems of polynomials, to appear Math. Of Computation.

Adaptive Multiprecision

Is Costly!

double (52 bits)	64 bits	96 bits	128 bits	256 bits	512 bits	1024 bits
2.447	32.616	35.456	35.829	50.330	73.009	124.401

GMP is convenient, but because of its relative machine independence it takes almost no advantage of the built in hardware floating point

Adaptive multiprecision

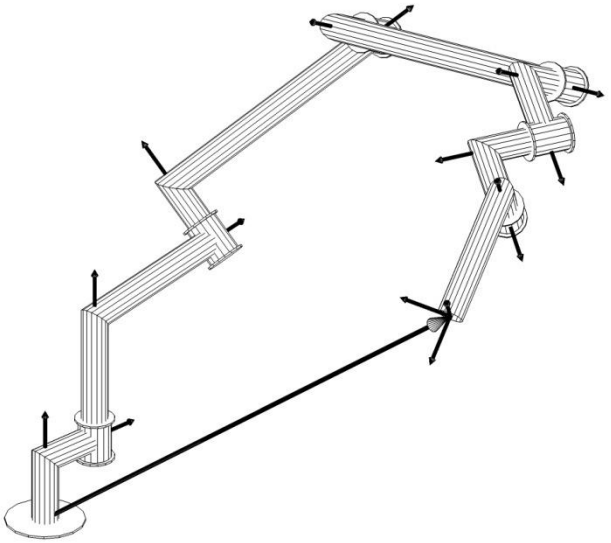
	96-bit	AMP	AMP2
time (sec)	196.143	47.393	39.854
paths/sec	1.305	5.402	6.423

Average time, in seconds, needed to solve the *Inverse Kinematic Problem of General 6R Serial Robot* using Bertini with tracking tolerance of 10^{-6} and

final tolerance of 10^{-12}

2.4 GHz, Opteron 250 processor

Six revolute serial link robot



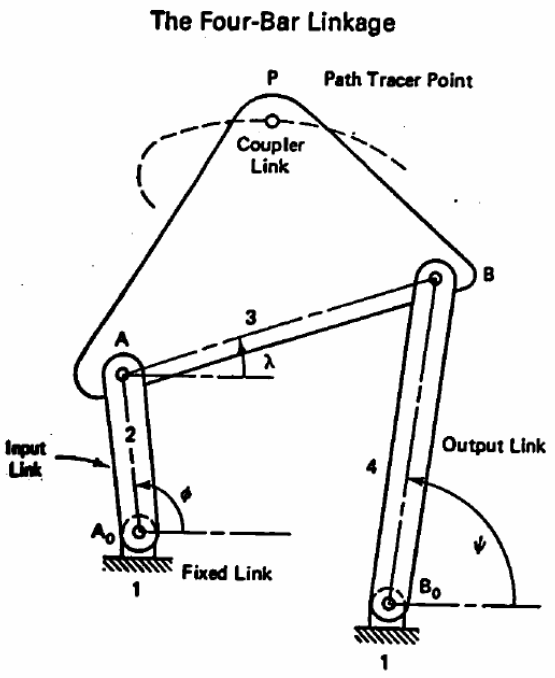
Adaptive multiprecision

Near-singular conditions actually arise.

Out of 143,360 paths (for the 9-point problem):

1184 paths (0.826%) used higher precision and then dropped back to double precision before starting the endgame

680 paths (0.474%) used at least 96-bit precision and then dropped back to double precision before starting the endgame



Continuation's Core Computation

- Given a system f(x) = 0 of N polynomials in N unknowns, continuation computes a finite set S of solutions such that:
  - any isolated root of f(x) = 0 is contained in S;
  - any isolated root "occurs" a number of times equal to its multiplicity as a solution of f(x) = 0;
  - S is often larger than the set of isolated solutions.

Local Dimension Test

- The essential case: check if p is isolated
- Homotopy continuation yields a number which bounds the multiplicity if the point was isolated.
- If not isolated, the space of truncated Taylor series around p of functions on the solution space is strictly increasing in dimension
- The Macaulay matrix (as presented by Dayton-Zeng) computes this dimension

Implementation Considerations

- Computation of the rank of the Macaulay matrix requires
  - Different levels of precision
  - Reliable multiple precision endgame to compute point p to needed accuracy

Some Comparisons Using Bertini

	is_isolated		membership test	
m	slicing	cascade	slicing	cascade
3	0.12	0.15	0.12	0.17
4	0.71	1.12	1.15	1.32
5	4.96	7.30	11.86	10.68
6	29.26	71.51	149.59	92.28
7	183.14	288.70	2036.73	854.33
8	1157.74	1714.35	17362.71	8720.14
9	7296.78	9533.50	219509.84	83060.43

Comparison for computing a numerical irreducible decomposition for G_m, in seconds

2xm adjacent minors of 3xm matrix

Parallel Version (64 cores)

- 8 dual quad-core Xeon 5410s (2.33 GH)

	is_isolated		membership test	
m	slicing	cascade	slicing	cascade
7	15.83	16.36	82.59	30.03
8	35.87	49.88	350.96	168.46
9	138.91	213.23	3320.04	1399.43

Equation-by-Equation Methods

- Potential to solve systems with relatively few solutions that are completely outside of the beyond the pale of standard continuation methods
  - Intersection Method by Sommese, Verschelde, and Wampler
  - Regeneration Method by Hauenstein, Sommese, and Wampler

Basic Idea

It is a natural and ancient approach to work *equation by equation.*

First Solve $f_1(x) = 0$;

then solve $\begin{bmatrix} f_1(z) \\ f_2(z) \end{bmatrix} = 0$;

then solve $\begin{bmatrix} f_1(z) \\ f_2(z) \\ f_3(z) \end{bmatrix} = 0$;**. . . and so on.**

Both methods generate a witness set for $f_1,..,f_{k+1}$

Regeneration $\longrightarrow$ $$H(z,t) = \begin{bmatrix} f_1(z) \\ \vdots \\ f_k(z) \\ (1-t)f_{k+1}(z) + \gamma t \Pi_{i=1}^{\deg f_{k+1}} L_{k+1,i}(z) \end{bmatrix}$$

$$H(w,t) = \begin{bmatrix} f_1 \circ \pi_1 \\ \vdots \\ f_k \circ \pi_1 \\ f_{k+1} \circ \pi_2 \end{bmatrix} ((1-t)\Psi(w) + \gamma t \Phi(w)) \longleftarrow \text{Intersection}$$

$$H(z,t) = \begin{bmatrix} f_1(z) \\ \vdots \\ f_k(z) \\ (1-t)f_{k+1}(z) + \gamma t \Pi_{i=1}^{\deg f_{k+1}} L_{k+1,i}(z) \end{bmatrix}$$

Hardware

- Continuation is computationally intensive. On average:
  - in 1985: 3 minutes/path on largest mainframes.

Hardware

- Continuation is computationally intensive.
  On average:
  - in 1985: 3 minutes/path on largest mainframes.
  - in 1991: over 8 seconds/path on an IBM 3081;
    2.5 seconds/path on a top-of-the-line IBM 3090.

Hardware

- Continuation is computationally intensive. On average:
  - in 1985: 3 minutes/path on largest mainframes.
  - in 1991: over 8 seconds/path on an IBM 3081; 2.5 seconds/path on a top-of-the-line IBM 3090.
  - 2008: 10+ paths a second on an single processor desktop CPU; 1000's of paths/second on moderately sized clusters.

Bertini

- Developed by Daniel Bates, Jonathan Hauenstein, Charles Wampler, and myself

- Binaries for Linux (including clusters and multiple core workstations), Macs, Windows are freely available at

www.nd.edu/~sommese/bertini

Bertini

- Bertini is designed to
  - Be efficient and robust, e.g., straightline evaluation, numerics with careful error control
  - With data structures reflecting the underlying geometry
  - Take advantage of parallel hardware
  - To dynamically adjust the precision to achieve a solution with a prespecified error.

Major Ingredients in Bertini

- Adaptive Multiprecision
- Straightline evaluation
- Special Homotopies
- Genericity
- Endgames & ODE Methods
- Intersections
- Deflation
- Multiplicity & Local Dimension Testing
- Regeneration

Solving Differential Equations

- E.L. Allgower, D.J. Bates, A.J. Sommese, and C.W. Wampler, Solution of Polynomial systems derived from differential equations, Computing, 76 (2006), 1-10.
- Direct solution and refinement.

Predator-prey system (Hauenstein, Hu, & S.)

Let $n \in \mathbb{N}$. For $1 \leq i \leq n$ and $1 \leq j \leq 4$, define

$$f_{ij} = \frac{1}{25}\left(u_{i+1,j} - 2u_{i,j} + u_{i-1,j}\right) + \frac{1}{(n+1)^2}\left(u_{i,j+1} - 2u_{i,j} + u_{i,j-1}\right) + \frac{1}{25(n+1)^2} u_{i,j}\left(1 - v_{i,j}\right)$$

$$g_{ij} = \frac{1}{25}\left(v_{i+1,j} - 2v_{i,j} + v_{i-1,j}\right) + \frac{1}{(n+1)^2}\left(v_{i,j+1} - 2v_{i,j} + v_{i,j-1}\right) + \frac{1}{25(n+1)^2} v_{i,j}\left(u_{i,j} - 1\right)$$

with $u_{0,j} = v_{0,j} = u_{n+1,j} = v_{n+1,j} = u_{i,0} = v_{i,0} = u_{i,5} = v_{i,5} = 0$.

- 8n quadratics with 8n variables
  - Total degree 2^{8n}
  - Actually has 2^{4n} nonsingular isolated solutions

	total degree	2-homogeneous	polyhedral	regeneration	
n	paths	paths	paths	paths	slices moved
1	256	70	16	60	42
2	65,536	12,870	256	1020	762
3	16,777,216	2,704,156	4096	16,380	12,282
4	4,294,967,296	601,080,390	65,536	262,140	196,602
5	1,099,511,627,776	137,846,528,820	1,048,576	4,194,300	3,145,722

	PHC	HOM4PS-2.0	Bertini	
n	polyhedral	polyhedral	regeneration	parallel regeneration
1	0.6s	0.1s	0.3s	
2	4m57s	7.3s	15.6s	
3	18d10h18m56s	9m32s	9m43s	
4	-	3d8h28m30s	5h22m15s	7m32s
5	-	-	6d16h27m3s	3h41m24s

n = 5 (40 equations & 40 variables): < 80 min.
with 200 cores (25 dual Xeon 5410 nodes)

Zebra Fish

- Why do the stripes on a zebra fish or the spots on a tiger form the patterns they do?
  - Alan Turing (1952), The chemical basis of morphogenesis: nonlinear diffusion equations.
- A good reference for this story is Mathematical Biology by J.D. Murray

- Based on the model developed in
  - Y.–T. Zhang, A. Lander, and Q. Nie, Computational analysis of BMP gradients in dorsal–ventral patterning of the zebrafish embryo, *Journal of Theoretical Biology*, 248(4) : 579 – 589, 2007.
- Our work
  - W. Hao, Y. Liu, J. Hauenstein, B. Hu, A. Sommese, and Y.-T. Zhang, Multiple stable steady states of a reaction-diffusion model on zebrafish dorsal-ventral patterning, *to appear Discrete and Continuous Dynamical Systems - Series S.*

The differential equation system

$$\frac{\partial[L]}{\partial t} = D_L \frac{\partial^2[L]}{\partial x^2} - k_{on}[L](R_0 - [LR]) + k_{off}[LR] - j_{on}[L][C] + (j_{off} + \tau)[LC] + V_L;$$

$$\frac{\partial[LR]}{\partial t} = k_{on}[L](R_0 - [LR]) - (k_{off} + k_{deg})[LR];$$

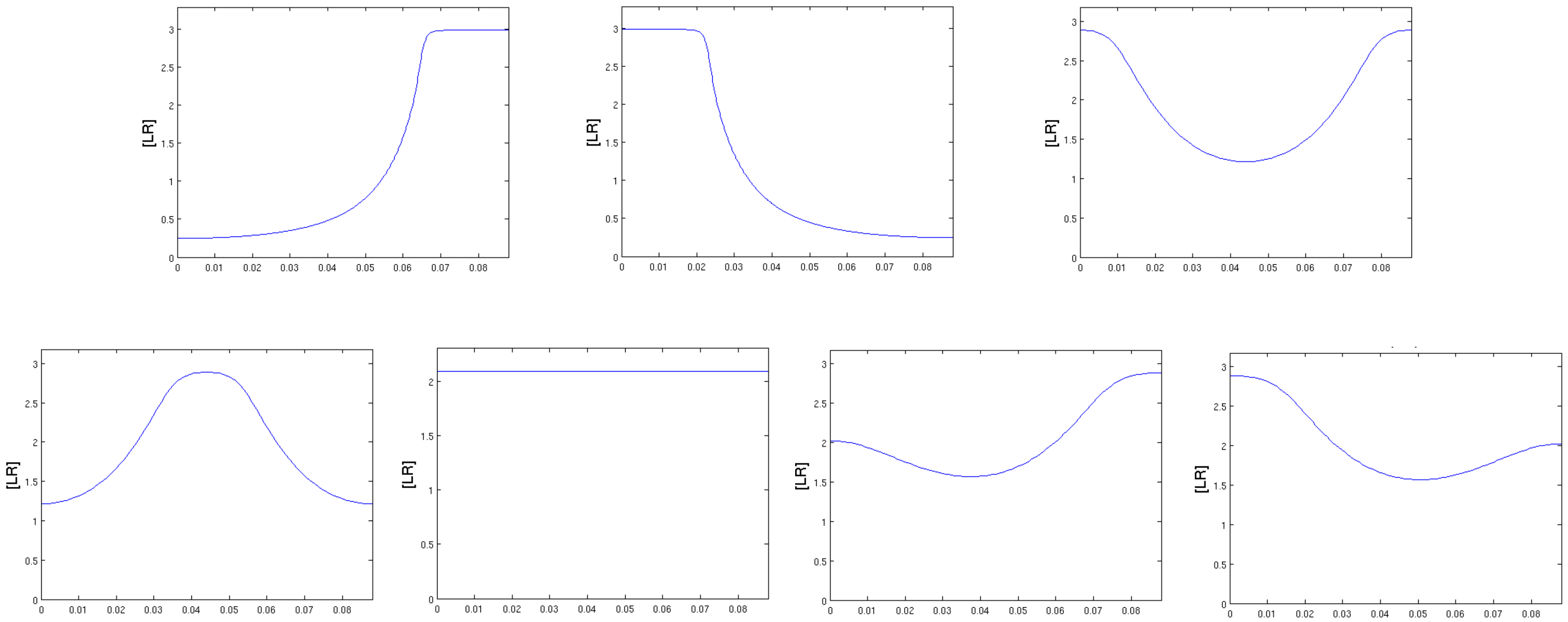
$$\frac{\partial[LC]}{\partial t} = D_{LS} \frac{\partial^2[LC]}{\partial x^2} + j_{on}[L][C] - (j_{off} + \tau)[LC];$$

$$\frac{\partial[C]}{\partial t} = D_C \frac{\partial^2[C]}{\partial x^2} - j_{on}[L][C] + j_{off}[LC] + V_C,$$

$$V_C = V_{Cmin} + \frac{V_{Cmax} - V_{Cmin}}{1 + \gamma_C[LR]} + \begin{cases} V_{Corg} e^{-at}, & \text{if } x \geq \frac{7}{8} x_{max}; \\ 0, & \text{otherwise.} \end{cases}$$

$$V_L = V_{Lmin} + \frac{V_{Lmax} - V_{Lmin}}{1 + \gamma_L [LR]^{-1}} + V_{Lmat} e^{-bt}.$$

Solutions



Some timings

- Total degree 16^{N-1} (which = 4,294,967,296 When N = 9).

N	lin. prod. bound	solutions over $\mathbb{C}$	solutions over $\mathbb{R}$	computing nodes	time
3	25	16	6	serial	2.7s
4	125	98	16	serial	14.4s
5	625	544	28	1	21.1s
6	3,125	2,882	184	5	51.6s
7	15,625	14,896	930	25	2m43s
8	78,125	75,938	3,720	25	35m2s
9	390,625	384,064	17,974	25	11h3m

Table 2.1: Summary of solving the discretized system for $3 \leq N \leq 9$

Tumor growth

$$
\begin{aligned}
\sigma_t - \Delta\sigma &= -\sigma && \text{in } \Omega(t) \\
-\Delta p &= \mu(\sigma - \widetilde{\sigma}) && \text{in } \Omega(t)
\end{aligned}
$$

$$
\begin{aligned}
\sigma &= 1 && \text{on } \partial\Omega(t) \\
p &= \kappa && \text{on } \partial\Omega(t) \\
\frac{\partial p}{\partial n} &= -V_n && \text{on } \partial\Omega(t).
\end{aligned}
$$

Assumptions

- In vitro

$\Omega(t)$ denotes the tumor region, σ denote the concentration of nutrients, p denote the pressure, $\widetilde{\sigma}$ denote the concentration of nutrients needed for sustainability, and μ denote the aggressiveness of the tumor. Let κ denote the mean curvature, n denote the outward normal direction, and V_n denote the velocity of $\partial\Omega(t)$ in the outward normal direction n.

Governing equations:

- Diffusion of the nutrients:

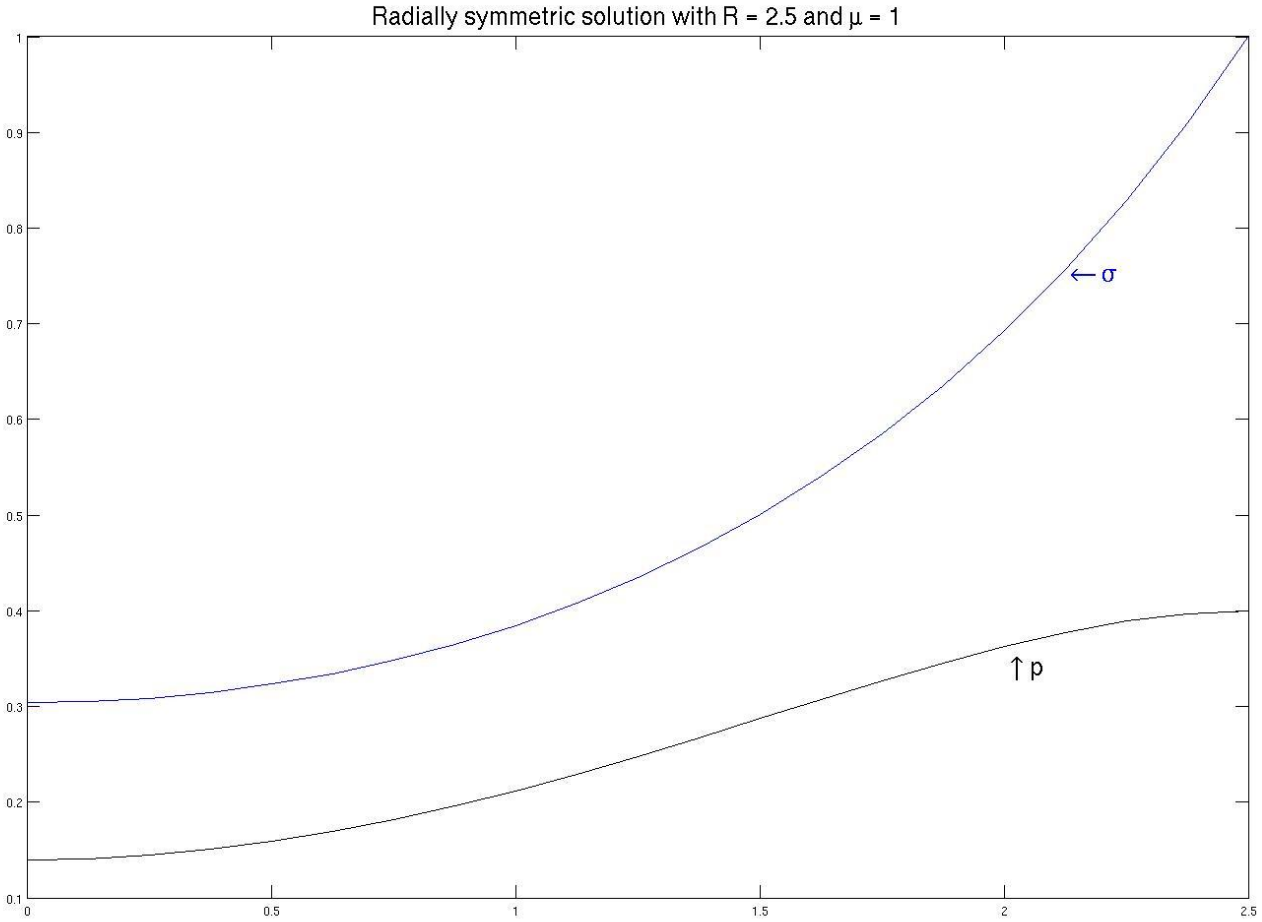
$$\sigma_t - \Delta\sigma + \sigma = 0 \quad \text{in } \Omega(t).$$

- Conservation of mass: $\operatorname{div}\vec{V} = S$, $S =$ proliferation rate. Assuming linear dependence on σ: $S = \mu(\sigma - \tilde{\sigma})$, (here $\tilde{\sigma} > 0$ is the death rate)
- Porous medium in tumor region: Darcy's law: $\vec{V} = -\nabla p$. Thus

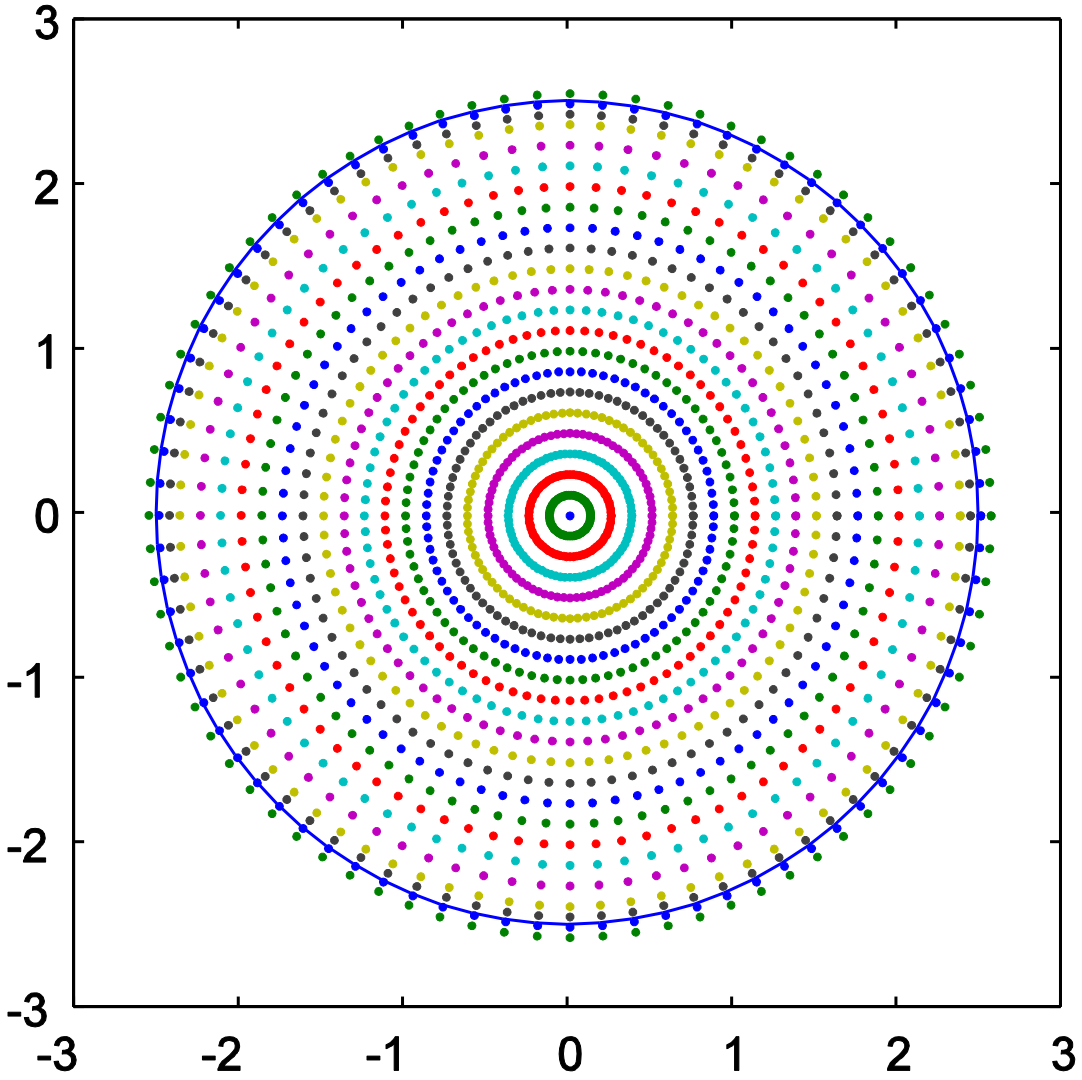
$$\Delta p = -\mu(\sigma - \tilde{\sigma}) \quad \text{in } \Omega(t).$$

- Continuity: $V_n = -\dfrac{\partial p}{\partial n}$ on $\partial\Omega(t)$
  where V_n= velocity in the normal n direction.

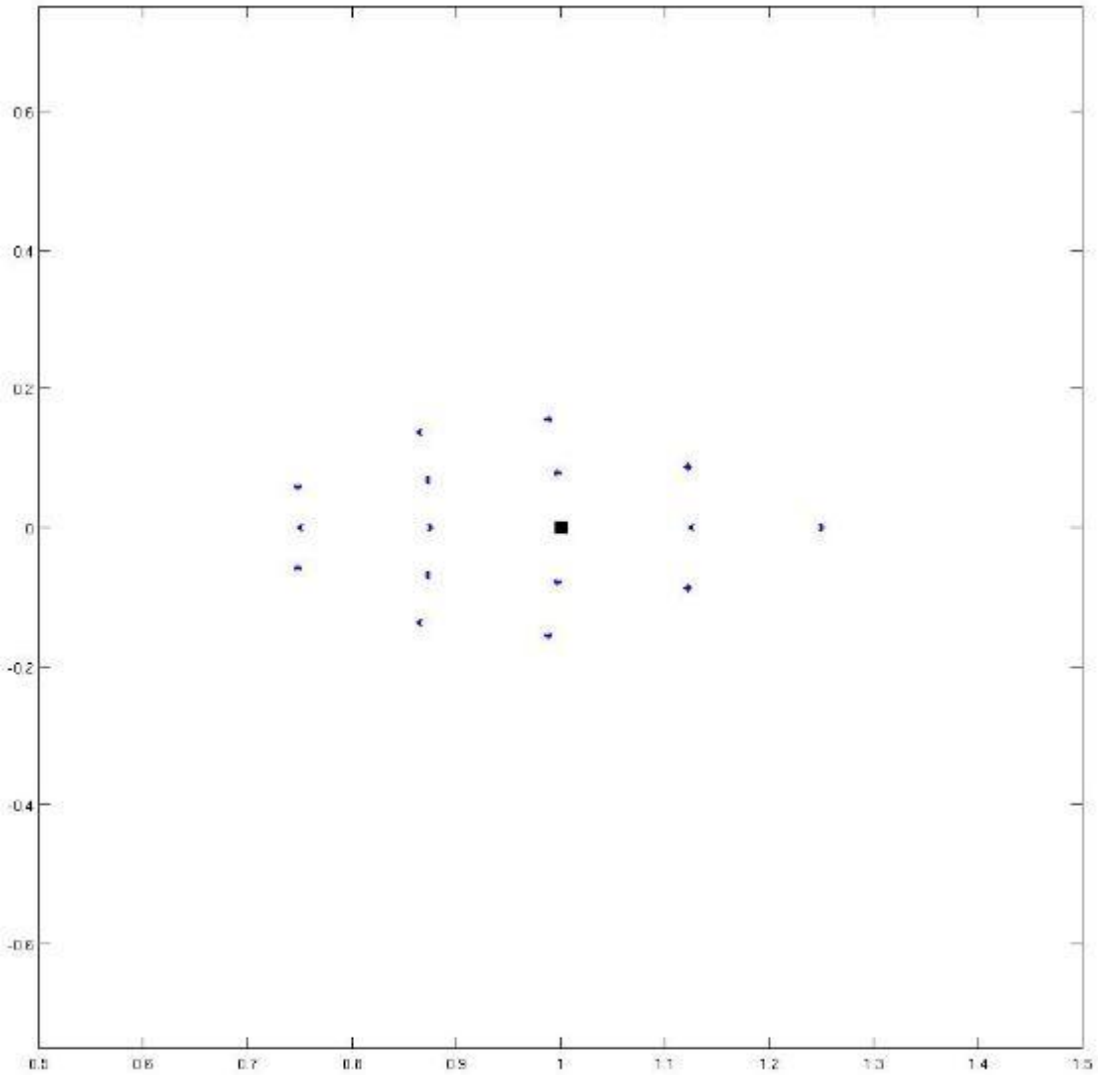
Radial solution is quite cheap: < 1 sec. (one core)



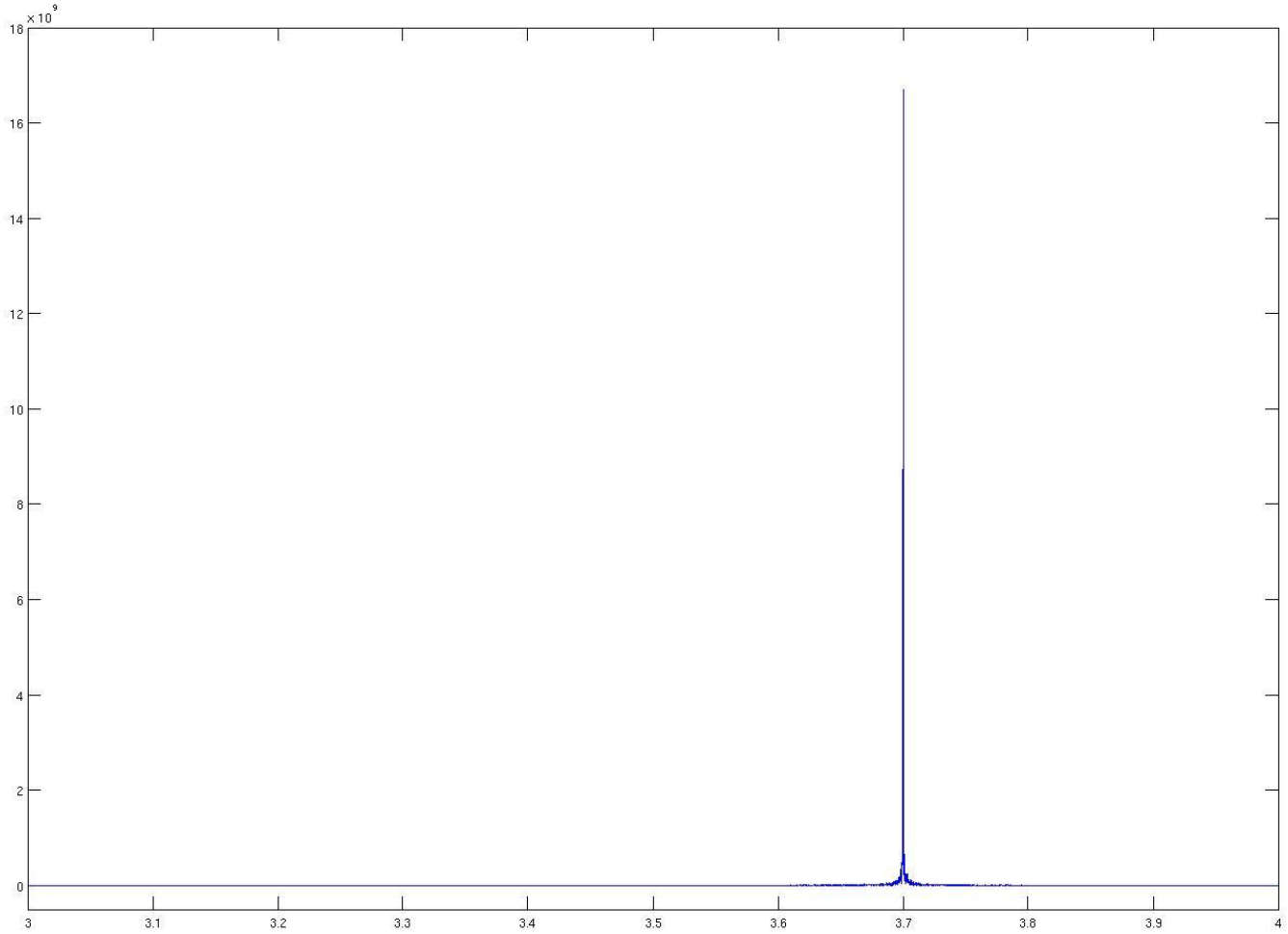
Moving Grid



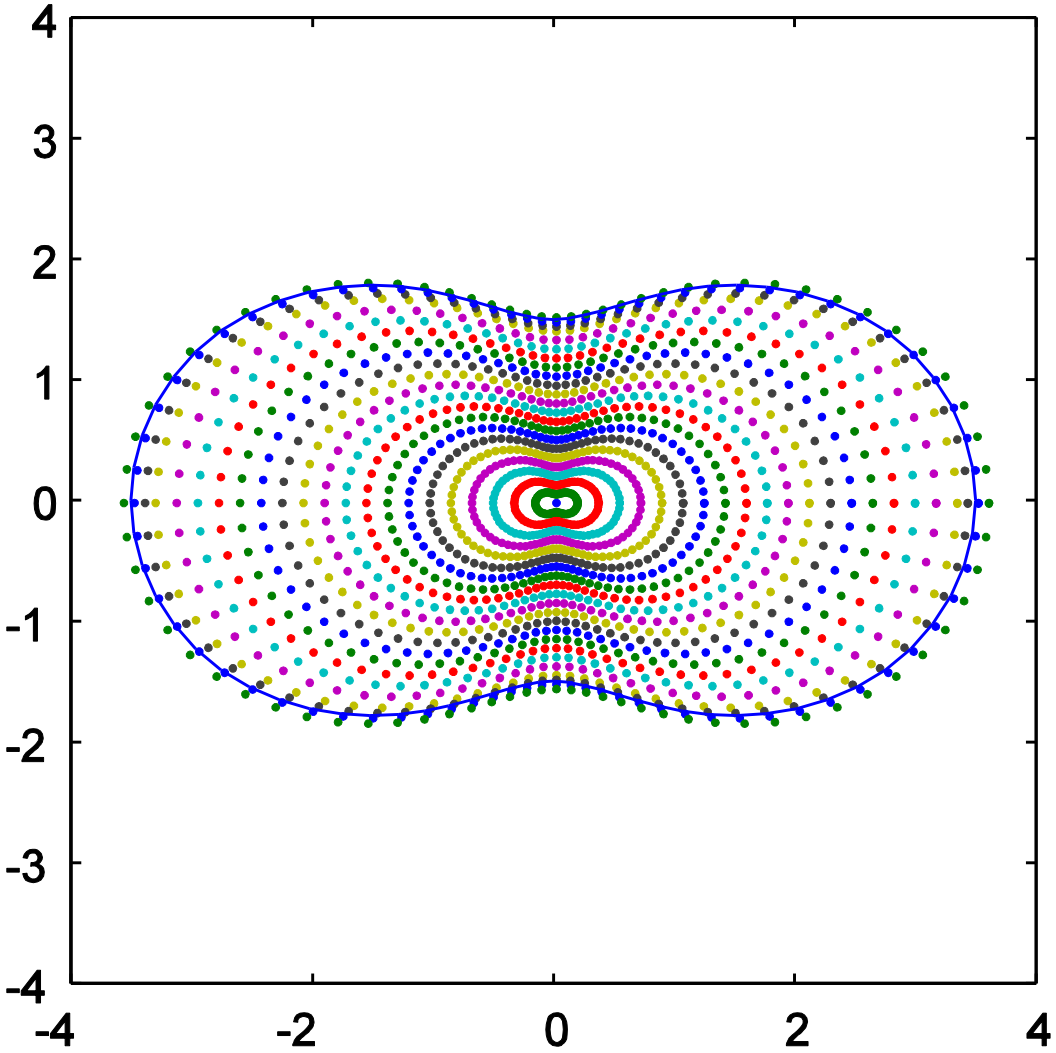
3rd Order Stencil



Critical Points 3 minutes with 200 cores



Far Along the Branch



Further work

- Stability
- More realistic models
  - Three Dimensional Models
  - Necrotic Core Models (disconnected free boundaries)
  - Model presented in Friedman & Hu, Bifurcation for a free boundary problem modeling tumor growth by Stokes equation, SIAM J. Math. Anal., 39, 174-194.

Stationary Problem

$$\sigma_t - \Delta\sigma + \sigma = 0, \quad x \in \Omega(t),\ t > 0,$$
$$\sigma = 1, \quad x \in \Omega(t),\ t > 0,$$
$$-\Delta\vec{v} + \nabla p = \frac{\mu}{3}\nabla(\sigma - \tilde{\sigma}), \quad x \in \Omega(t),\ t > 0,$$
$$\mathrm{div}\vec{v} = \mu(\sigma - \tilde{\sigma}), \quad x \in \Omega(t),\ t > 0 \quad (\tilde{\sigma} < 1),$$
$$T(\vec{v}, p)\vec{n} = \Big(-\gamma\kappa + \frac{2}{3}\mu(1 - \tilde{\sigma})\Big)\vec{n}, \quad x \in \Gamma(t),\ t > 0,$$
$$T(\vec{v}, p) = (\nabla\vec{v})^T + \nabla\vec{v} - p\, I, \qquad I = (\delta_{ij})_{i,j=1}^3,$$
$$V_n = \vec{v} \cdot \vec{n} \quad \text{on } \Gamma(t),$$

subject to the constraints

$$\int_{\Omega(t)} \vec{v}\, dx = 0, \qquad \int_{\Omega(t)} \vec{v} \times \vec{x}\, dx = 0.$$

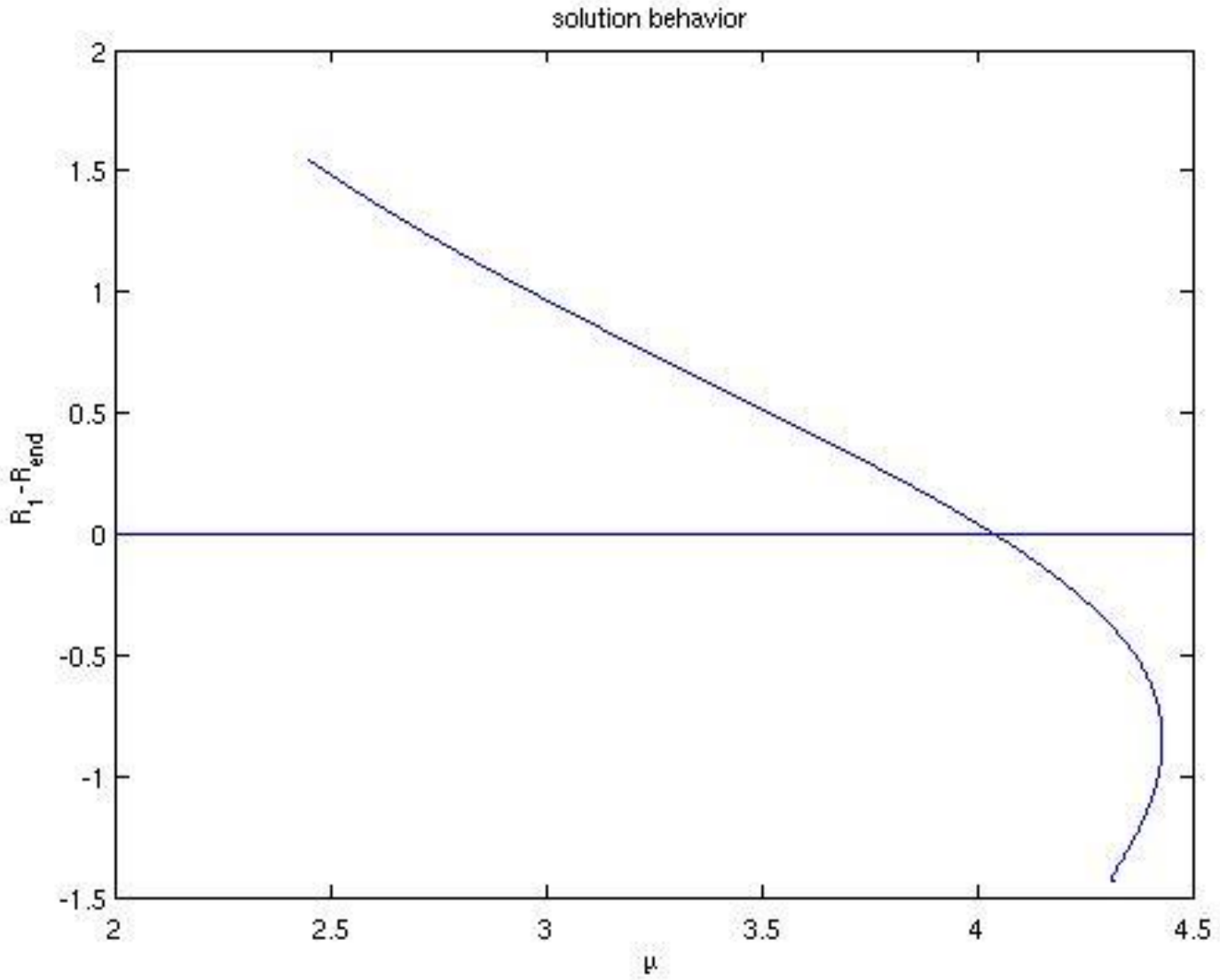
Governing equations:

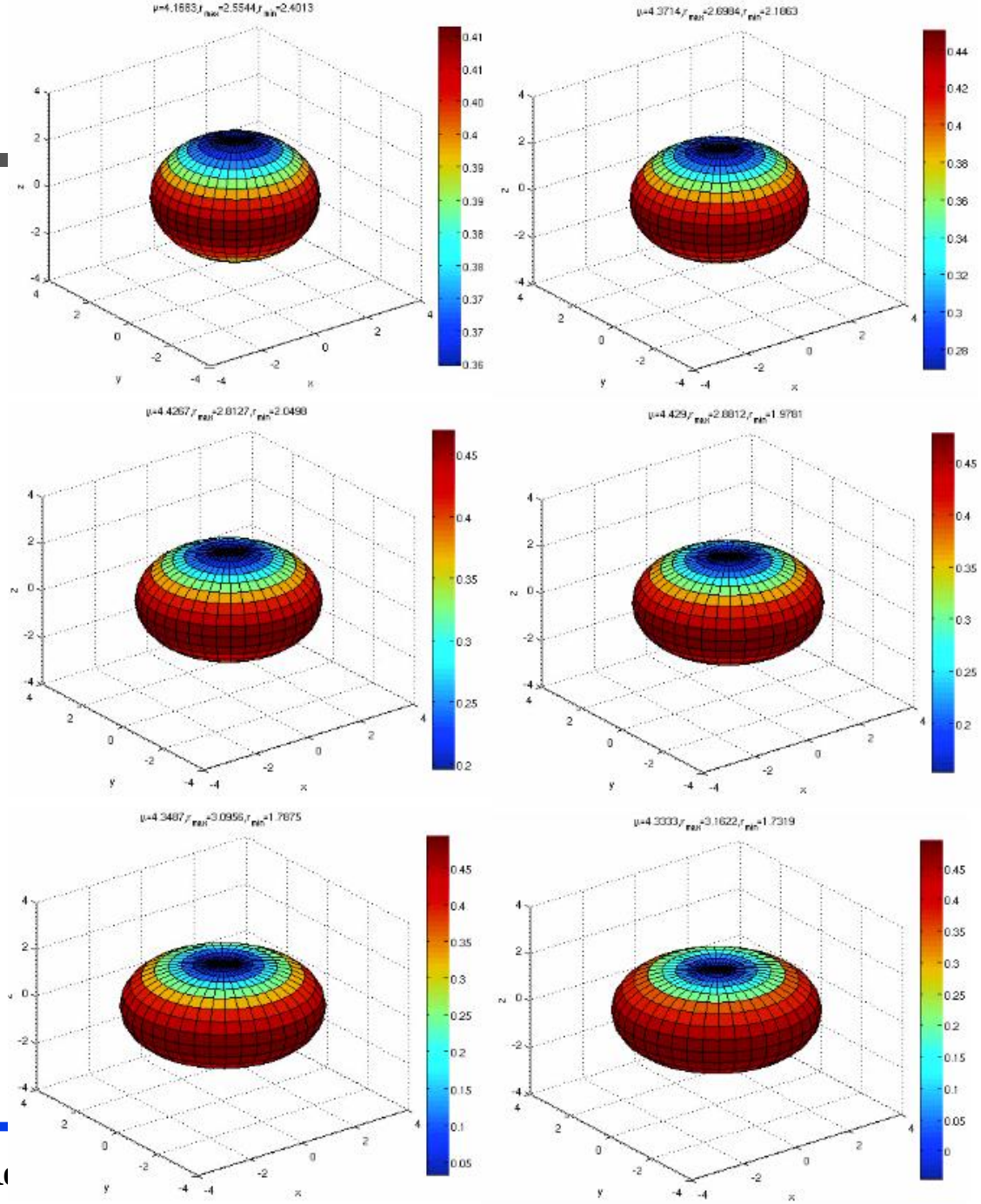
- Diffusion of the nutrients: $\sigma_t - \Delta\sigma + \sigma = 0 \quad$ in $\Omega(t)$.
- Conservation of mass: $\mathrm{div}\vec{V} = S$, $S =$ proliferation rate.
  Assume linear dependence on σ: $S = \mu(\sigma - \tilde{\sigma})$, (here $\tilde{\sigma} > 0$ is the death rate)
- Instead of Darcy's law, Stoke's equation is used: $-\nu\Delta\vec{v} + \nabla p - \frac{1}{3}\nu\nabla\mathrm{div}\vec{v} = 0 \quad$ in $\Omega(t)$.
- Introducing the stress tensor $Q = \nu(\nabla\vec{v} + (\nabla\vec{v})^T) - (p + \frac{2}{3}\nu\mathrm{div}\vec{v})I$ with components
  $Q_{ij} = \nu\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right) - \delta_{ij}\left(p + \frac{2\nu}{3}\mathrm{div}\vec{v}\right)$, we then have

$$Q\vec{n} = -\gamma\kappa\vec{n} \quad \text{on } \Gamma(t), \quad t > 0,$$

  here the cell-to-cell adhesion equal to a constant γ, κ is the mean curvature.
- Continuity: $V_n = \vec{v}\cdot\vec{n} \quad$ on $\partial\Omega(t)$
  where V_n= velocity in the normal n direction.

Since $\vec{v}$ is determined up to $\vec{b}\times\vec{x}$, some additional constraints are needed.


solution behavior
2
1.5
1
0.5
0
-0.5
-1
-1.5
R_1-R_{end}
2
2.5
3
3.5
4
4.5
μ



Algebraic Geometry

- Infinite Dimensional Algebraic Sets = Solutions of Differential Equations?

- Coupled Towers of Finite Dimensional Algebraic Sets?

Summary

- Basic but difficult questions about Scientific Models lead to algebraic sets defined by highly structured, sparse systems of polynomials that are extremely large by classical standards.
- Numerical Algebraic Geometry can make contributions when coupled with moderate amounts of computer power and appropriate numerical software.