Zebra Fish, Tumor Growth, and Algebraic Geometry

Andrew Sommese

University of Notre Dame

www.nd.edu/~sommese

Thank you for permission to use slides:

- Slides 28 - 30: Charles Wampler
- Slides 34 – 40: Jonathan Hauenstein
- Slide 48: Wenrui Hao
- Slides 52 & 65: Bei Hu

Numer. Alg. Geometry Collaborators

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

*Bertini Team

Biological Modeling Collaborators

- Wenrui Hao
- Jonathan Hauenstein
- Bei Hu
- Yuan Liu
- Yong-Tao Zhang

General References

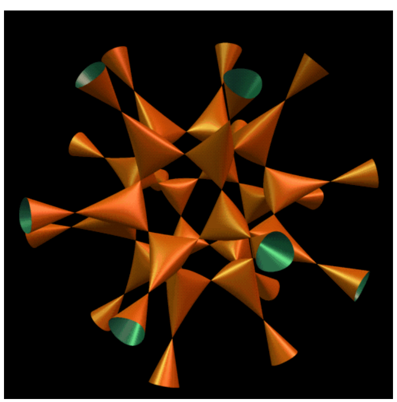
- Reference on the area up to 2005:
  - A.J. Sommese and C.W. Wampler, *Numerical solution of systems of polynomials arising in engineering and science*, (2005), World Scientific Press.
- Survey covering other topics
  - T.Y. Li, Numerical solution of polynomial systems by homotopy continuation methods, in *Handbook of Numerical Analysis*, Volume XI, 209-304, North-Holland, 2003.

Overview

- Numerical Algebraic Geometry
  - Solution Sets
  - Homotopy Continuation
  - Bertini
- Zebra Fish
- Tumor Growth
- Algebraic Geometry

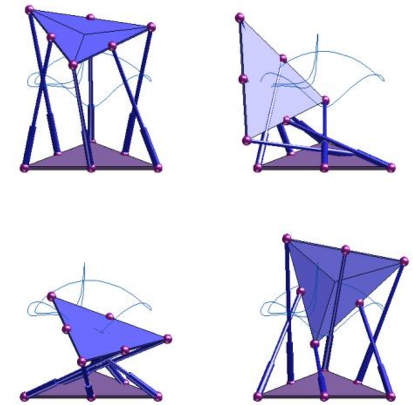
Numerical Algebraic Geometry

- **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**



graphics on right from Sommese-Wampler Book

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:

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

The Core Computation – In the Past!

- 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.

Computing Isolated Solutions

- Find all isolated solutions in C^N of a system on n polynomials:

$$\begin{bmatrix} \mathrm{f}_1(x_1,\ldots,x_N) \\ \vdots \\ \mathrm{f}_n(x_1,\ldots,x_N) \end{bmatrix} = 0$$

Solving a system

- Homotopy continuation is our main tool:
  - Start with known solutions of a known start system and then track those solutions as we deform the start system into the system that we wish to solve.

Path Tracking

This method takes a system g(x) = 0, whose solutions we know, and makes use of a homotopy, e.g.,

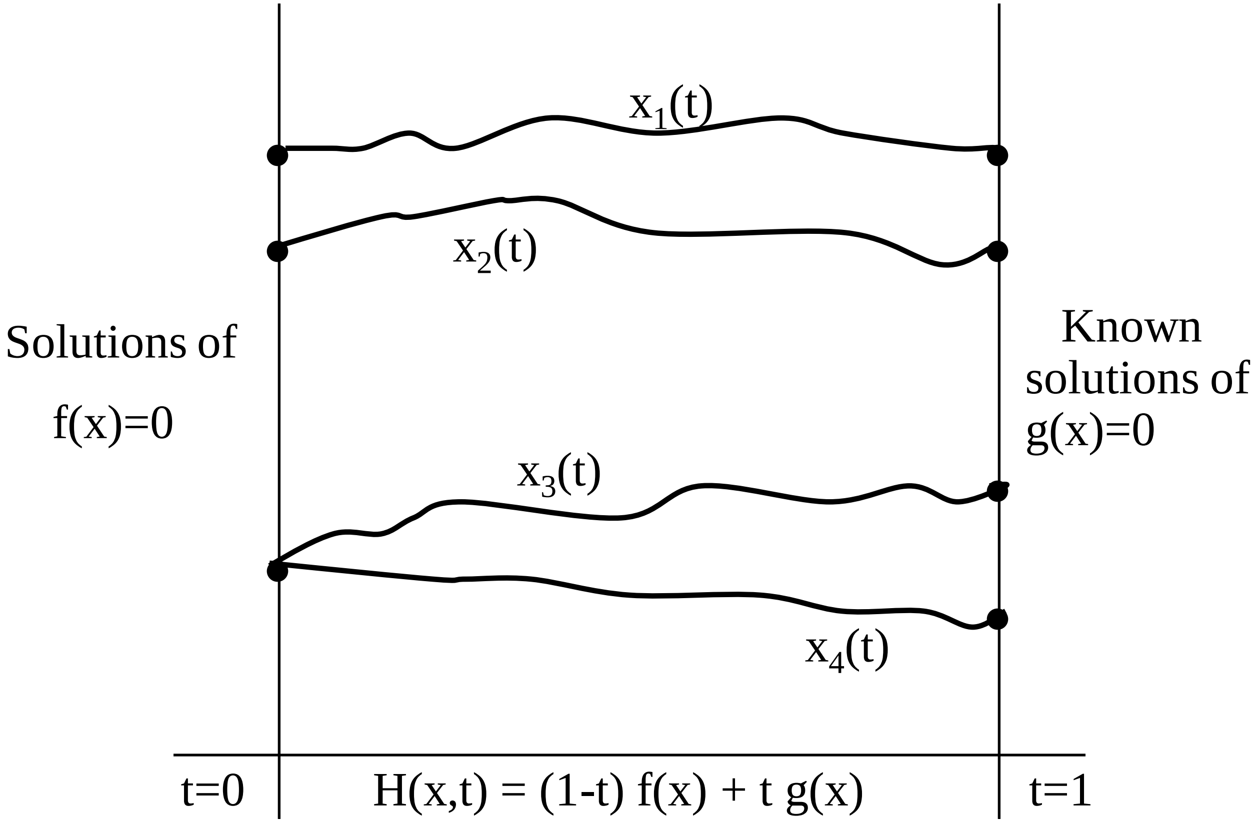
$$H(x, t) = (1 - t)f(x) + tg(x).$$

Hopefully, H(x,t) defines "paths" x(t) as t runs from 1 to 0. They start at known solutions of g(x) = 0 and end at the solutions of f(x) at t = 0.

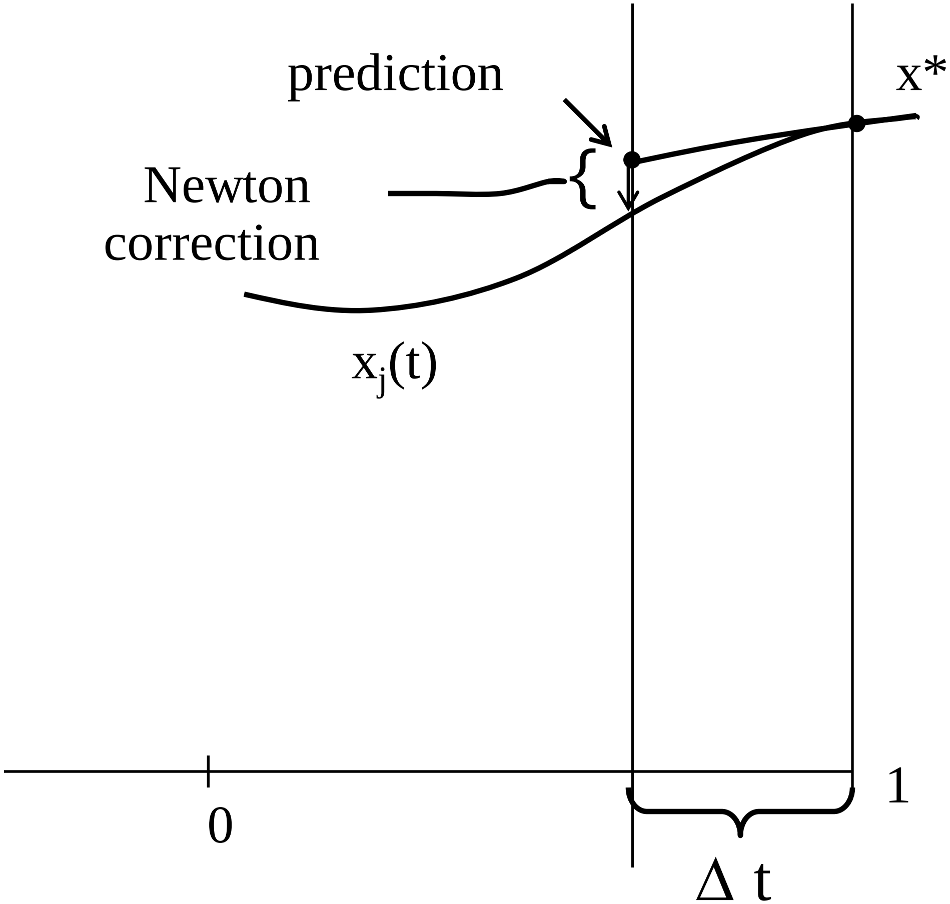
- The paths satisfy the Davidenko equation

$$0 = \frac{dH(x(t), t)}{dt} = \sum_{i=1}^{N} \frac{\partial H}{\partial x_i} \frac{dx_i}{dt} + \frac{\partial H}{\partial t}$$

- To compute the paths: use ODE methods to predict and Newton's method to correct.



$x_1(t)$
$x_2(t)$
Solutions of
f(x)=0
Known
solutions of
g(x)=0
$x_3(t)$
$x_4(t)$
t=0
H(x,t) = (1-t) f(x) + t g(x)
t=1

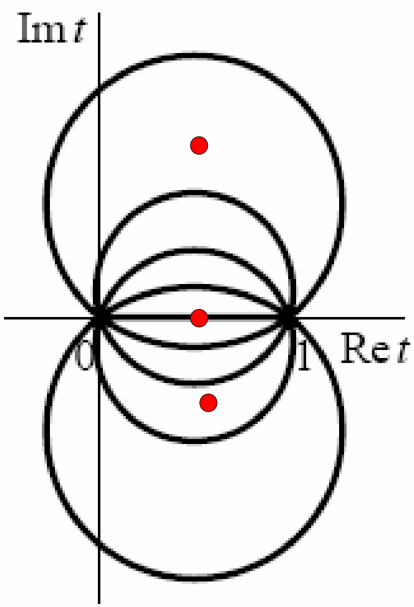

prediction
x*
Newton
correction
$x_j(t)$
0
1
Δ t

Uses of algebraic geometry

Simple but extremely useful consequence of algebraicity

- Instead of the homotopy $H(x,t) = (1-t)f(x) + tg(x)$

use $H(x,t) = (1-t)f(x) + \gamma t g(x)$



Major Ingredients

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

Major Computational Events

- Parallelization
  - No longer a niche tool requiring specialized hardware and nonstandard coding
- Multiprecision
  - No longer an option of last resort, highly nontrivial to design and dependent on hardware

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.
  - 2007: over 20 paths a second on an single processor desktop CPU;1000's of paths/second on moderately sized clusters; millions of paths on top-of-the-line 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.

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.
- 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, to appear SIAM Journal on Numerical Analysis.
- J.D. Hauenstein, A.J. Sommese, and C.W. Wampler, Regeneration homotopies for solving systems of polynomials.

Bertini and the need for adaptive precision

- Why use Multiprecision?
  - to ensure that the region where an endgame works is not contained the region where the numerics break down;

Bertini and the need for adaptive precision

- Why use Multiprecision?
  - to ensure that the region where an endgame works is not contained the region where the numerics break down;
  - to ensure that a polynomial is zero at a point is the same as the polynomial numerically being approximately zero at the point;

Bertini and the need for adaptive precision

- Why use Multiprecision?
  - to ensure that the region where an endgame works is not contained the region where the numerics break down;
  - to ensure that a polynomial is zero at a point is the same as the polynomial numerically being approximately zero at the point;
  - to prevent the linear algebra in continuation from falling apart.

Evaluation

$$p(z) = z^{10} - 28z^{9} + 1$$

- To 15 digits of accuracy one of the roots of this polynomial is a = 27.9999999999999. Evaluating p(a) to 15 digits, we find that

  p(a) = -0.578

- Even with 17 digit accuracy, the approximate root a is a = 27.999999999999905 and we still only have p(a) = -0.005.

Regeneration

- Basic step

$$V_0\left(\begin{bmatrix} f_1(x) \\ \vdots \\ f_{k-1}(x) \\ L_k(x) \\ L_{k+1}(x) \\ \vdots \\ L_N(x) \end{bmatrix}\right) \Rightarrow V_0\left(\begin{bmatrix} f_1(x) \\ \vdots \\ f_{k-1}(x) \\ f_k(x) \\ L_{k+1}(x) \\ \vdots \\ L_N(x) \end{bmatrix}\right)$$

Regeneration: Step 1

$$V_0\left(\begin{bmatrix} f_1(x) \\ \vdots \\ f_{k-1}(x) \\ L_k(x) \\ L_{k+1}(x) \\ \vdots \\ L_N(x) \end{bmatrix}\right)$$

move
linear fcn
d_k times

$$V_0\left(\begin{bmatrix} f_1(x) \\ \vdots \\ f_{k-1}(x) \\ L_{k,1}(x) \\ L_{k+1}(x) \\ \vdots \\ L_N(x) \end{bmatrix}\right) \cdot \cdot \cdot \; V_0\left(\begin{bmatrix} f_1(x) \\ \vdots \\ f_{k-1}(x) \\ L_{k,d_k}(x) \\ L_{k+1}(x) \\ \vdots \\ L_N(x) \end{bmatrix}\right)$$

Union of
sets

$$V_0\left(\begin{bmatrix} f_1(x) \\ \vdots \\ f_{k-1}(x) \\ L_{k,1}(x)\cdots L_{k,d_k}(x) \\ L_{k+1}(x) \\ \vdots \\ L_N(x) \end{bmatrix}\right)$$

Regeneration: Step 2

$$V_0\left(\begin{bmatrix} f_1(x) \\ \vdots \\ f_{k-1}(x) \\ L_{k,1}(x)\cdots L_{k,d_k}(x) \\ L_{k+1}(x) \\ \vdots \\ L_N(x) \end{bmatrix}\right) \xrightarrow{\text{Linear homotopy}} V_0\left(\begin{bmatrix} f_1(x) \\ \vdots \\ f_{k-1}(x) \\ f_k(x) \\ L_{k+1}(x) \\ \vdots \\ L_N(x) \end{bmatrix}\right)$$

Repeat for k+1,k+2,…,N

A Bottleneck & Local Dim. Testing

- Given a solution, i.e., a point p with f(p) = 0, what is the dimension at p of the solution component through p.

The problem becomes worse as dimension increases

Idea

- 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 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

Regenerative cascade

Adjacent minor system:

Determinants of 2×2 adjacent minors of a $3 \times m$ matrix with variable entries

For example: $m = 3$

$$\begin{bmatrix} x_1 & x_2 & x_3 \\ x_4 & x_5 & x_6 \\ x_7 & x_8 & x_9 \end{bmatrix}$$

UNIVERSITY OF NOTRE DAME

Regenerative cascade

Adjacent minor system:

Determinants of *2 x 2* adjacent minors of a *3 x m* matrix with variable entries

For example: $m = 3$ $\begin{bmatrix} x_1 & x_2 & x_3 \\ x_4 & x_5 & x_6 \\ x_7 & x_8 & x_9 \end{bmatrix}$

$$f_1 = x_1 x_5 - x_2 x_4$$

Regenerative cascade

Adjacent minor system:

Determinants of 2×2 adjacent minors of a $3 \times m$ matrix with variable entries

For example: $m = 3$

$$\begin{bmatrix} x_1 & x_2 & x_3 \\ x_4 & x_5 & x_6 \\ x_7 & x_8 & x_9 \end{bmatrix}$$

$$f_1 = x_1 x_5 - x_2 x_4$$

$$f_2 = x_2 x_6 - x_3 x_5$$

Regenerative cascade

Adjacent minor system:

Determinants of *2 x 2* adjacent minors of a *3 x m* matrix with variable entries

For example: $m = 3$

$$\begin{bmatrix} x_1 & x_2 & x_3 \\ x_4 & x_5 & x_6 \\ x_7 & x_8 & x_9 \end{bmatrix}$$

$$f_1 = x_1 x_5 - x_2 x_4 \qquad f_3 = x_4 x_8 - x_5 x_7$$

$$f_2 = x_2 x_6 - x_3 x_5$$

Regenerative cascade

Adjacent minor system:

Determinants of *2 x 2* adjacent minors of a *3 x m* matrix with variable entries

For example: $m = 3$

$$\begin{bmatrix} x_1 & x_2 & x_3 \\ x_4 & x_5 & x_6 \\ x_7 & x_8 & x_9 \end{bmatrix}$$

$$f_1 = x_1 x_5 - x_2 x_4 \qquad f_3 = x_4 x_8 - x_5 x_7$$

$$f_2 = x_2 x_6 - x_3 x_5 \qquad f_4 = x_5 x_9 - x_6 x_8$$

Numerical irreducible decomposition

Adjacent minor:

n	Decomposition
3	0.04s
4	0.18s
5	0.83s
6	2.67s
7	13.5s
8	31.8s

Numerical irreducible decomposition

Adjacent minor system:

n	Decomposition
3	0.04s
4	0.18s
5	0.83s
6	2.67s
7	13.5s
8	31.8s

	Membership test			Local dimension test		
n	Regen cascade	Dim-by-dim	Cascade	Regen cascade	Dim-by-dim	Cascade
3	0.1s	0.1s	0.2s	0.1s	0.1s	0.2s
4	0.8s	1.1s	1.3s	0.6s	0.8s	1.1s
5	6.2s	11.9s	11.2s	3.1s	4.6s	7.4s
6	1m1s	2m14s	1m34s	15.6s	29.0s	48.4s
7	10m36s	25m39s	14m54s	1m16s	3m8s	5m23s
8	2h12m54s	5h21m48s	2h33m5s	6m33s	19m45s	29m22s

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 $1 \le i \le n$ and $1 \le j \le 4$, 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 variable
  - 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 Xeon 5410)

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
  - Y. Liu, W. Hao, J. Hauenstein, B. Hu, A. Sommese, and Y.-T. Zhang, Multiple stable steady states of a reaction-diffusion model on zebrafish dorsal-ventral patterning

The differential equation system

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

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

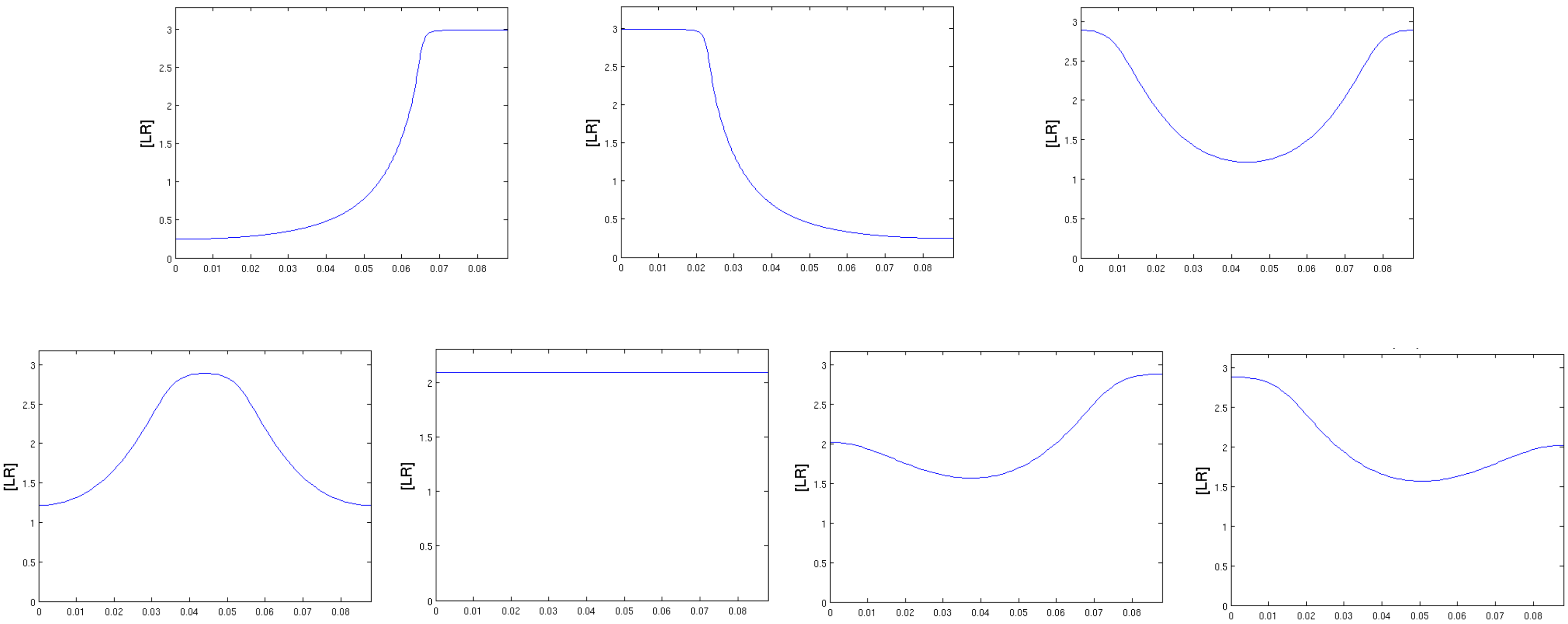
$$\frac{\partial[LC]}{\partial t} = D_{LS}\frac{\partial^2[LC]}{\partial x^2} + j_{on}[L][C] - (j_{off} + \tau)[LC];$$

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

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

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

Solutions



[LR]
0 0.5 1 1.5 2 2.5 3
0 0.01 0.02 0.03 0.04 0.05 0.06 0.07 0.08
[LR]
0 0.5 1 1.5 2 2.5 3
0 0.01 0.02 0.03 0.04 0.05 0.06 0.07 0.08
[LR]
0 0.5 1 1.5 2 2.5 3
0 0.01 0.02 0.03 0.04 0.05 0.06 0.07 0.08
[LR]
0 0.5 1 1.5 2 2.5 3
0 0.01 0.02 0.03 0.04 0.05 0.06 0.07 0.08
[LR]
0 0.5 1 1.5 2
0 0.01 0.02 0.03 0.04 0.05 0.06 0.07 0.08
[LR]
0 0.5 1 1.5 2 2.5 3
0 0.01 0.02 0.03 0.04 0.05 0.06 0.07 0.08
[LR]
0 0.5 1 1.5 2 2.5 3
0 0.01 0.02 0.03 0.04 0.05 0.06 0.07 0.08

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

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

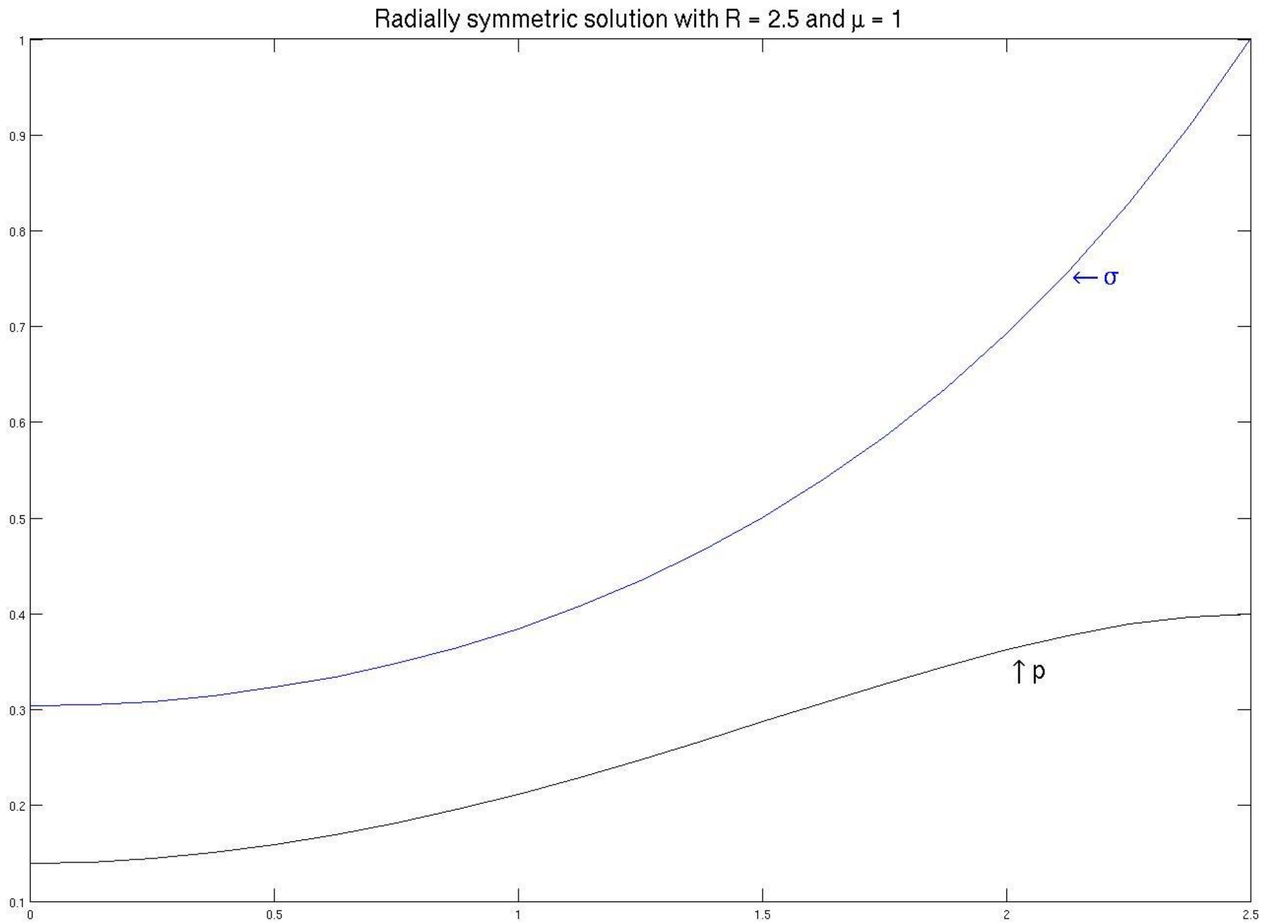
- Continuity: $V_n = -\dfrac{\partial p}{\partial n} \quad \text{on } \partial\Omega(t)$
  where V_n= velocity in the normal n direction.

Adding dead cells

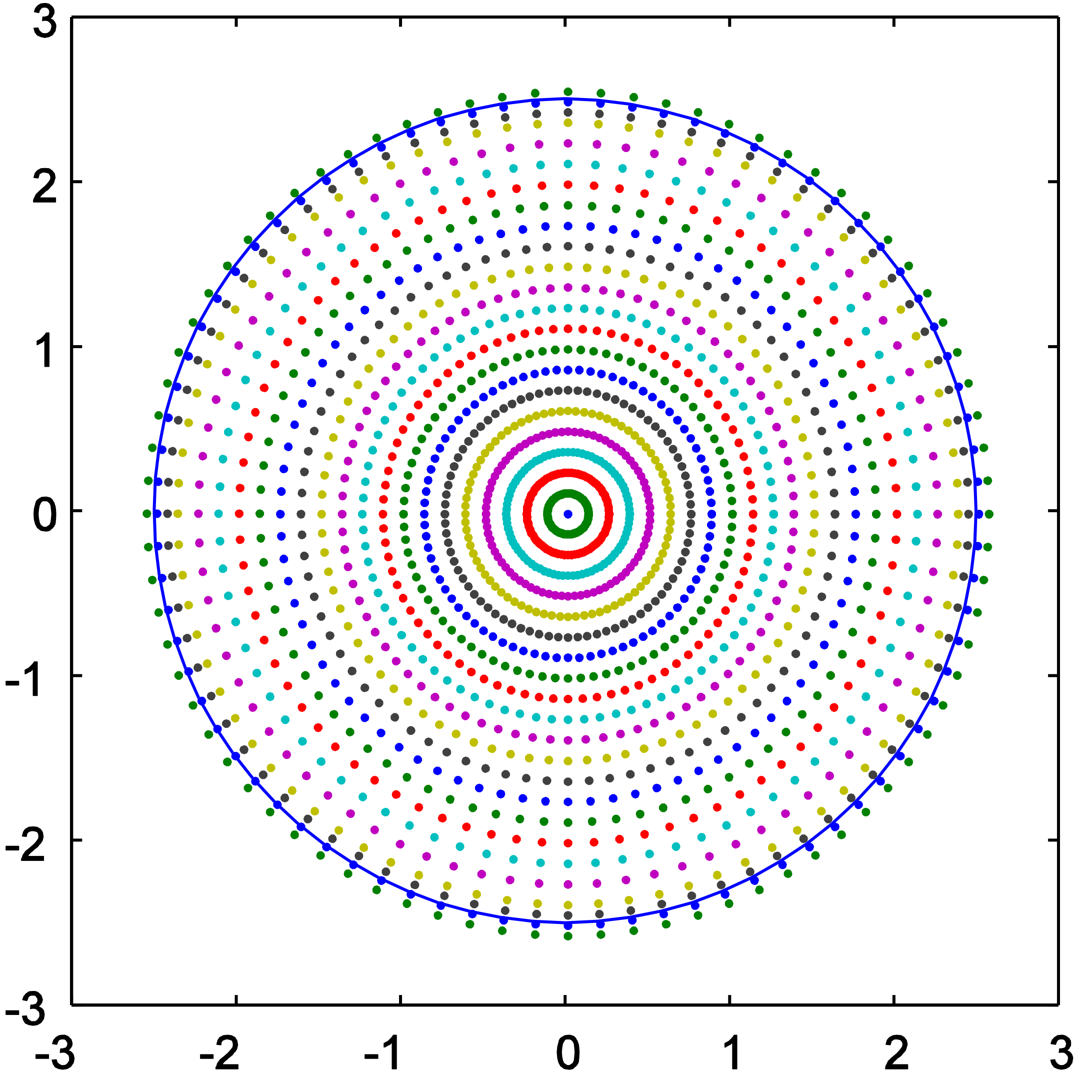
The steady-state tumor model is given by

$$\begin{cases} \Delta\sigma = \sigma\chi(x) & \text{in } \Omega \\ -\Delta p = \mu(\sigma - \tilde{\sigma})\chi(x) & \text{in } \Omega \\ \\ \sigma = \sigma_0 & \text{on } \partial D \\ \sigma = 1 & \text{on } \partial\Omega \\ p = \kappa & \text{on } \partial\Omega \\ \frac{\partial p}{\partial n} = 0 & \text{on } \partial\Omega. \end{cases}$$

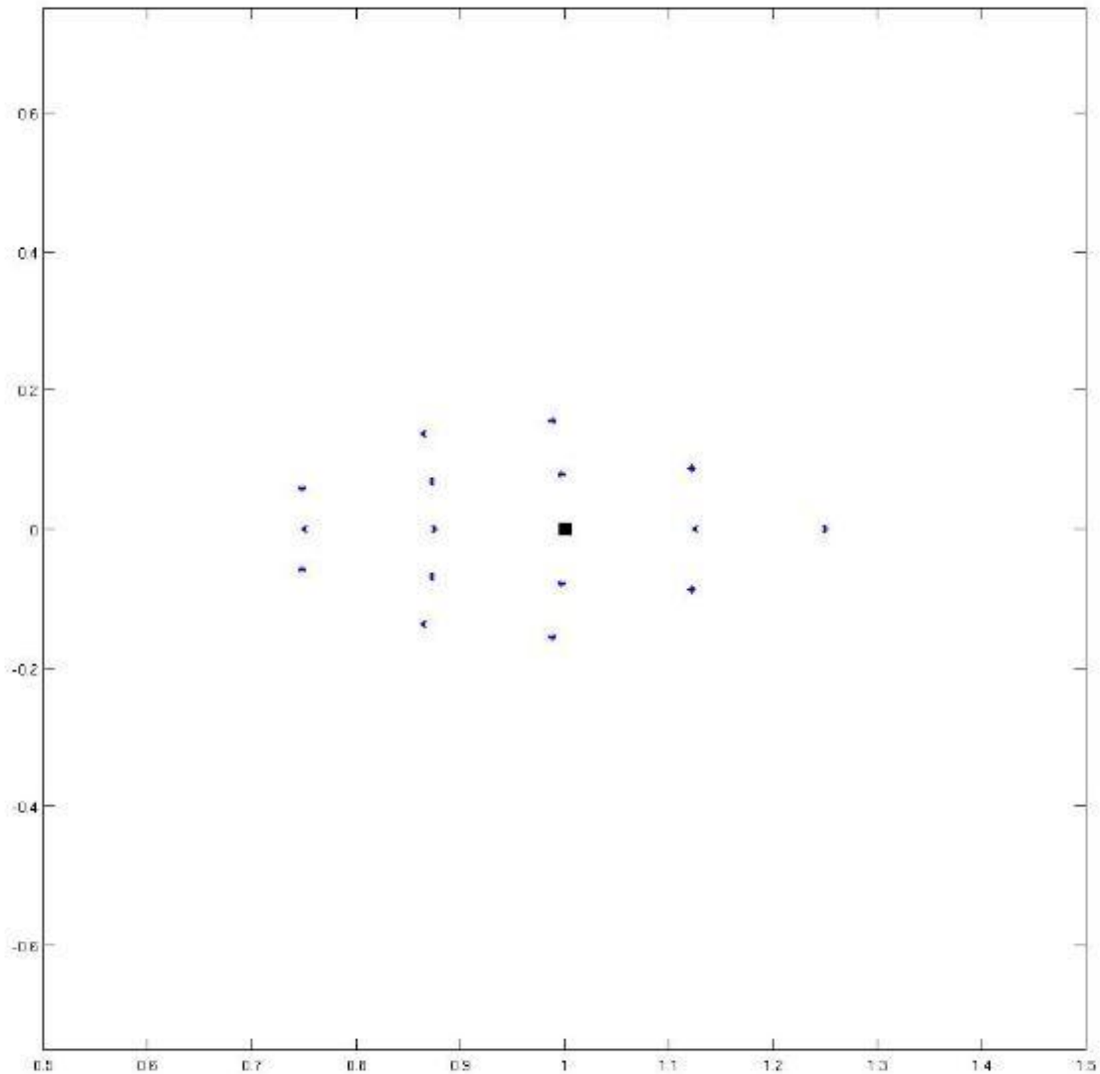
Radial solution is quite cheap: < 1 sec. (one core)



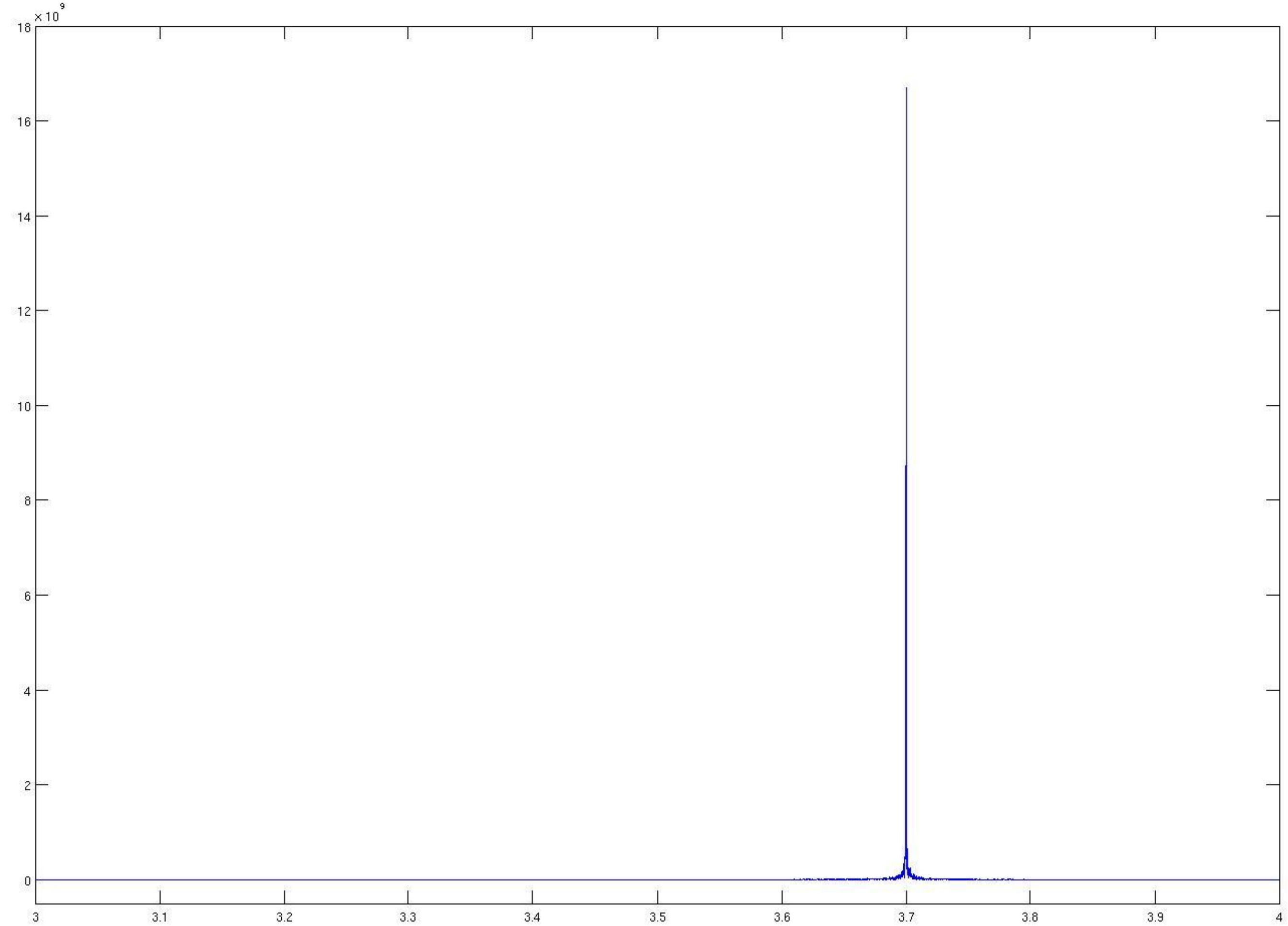
Moving Grid



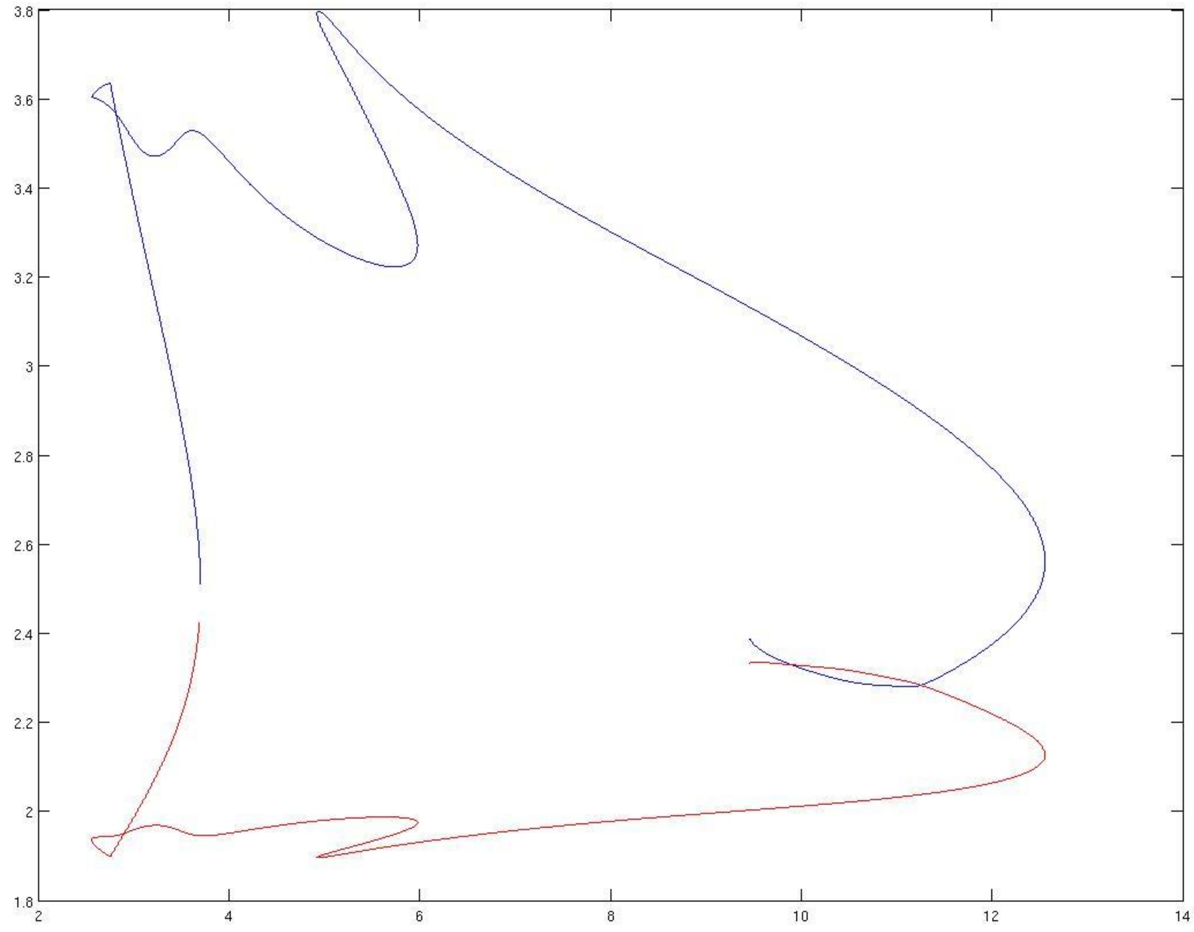
3^{rd} Order Stencil



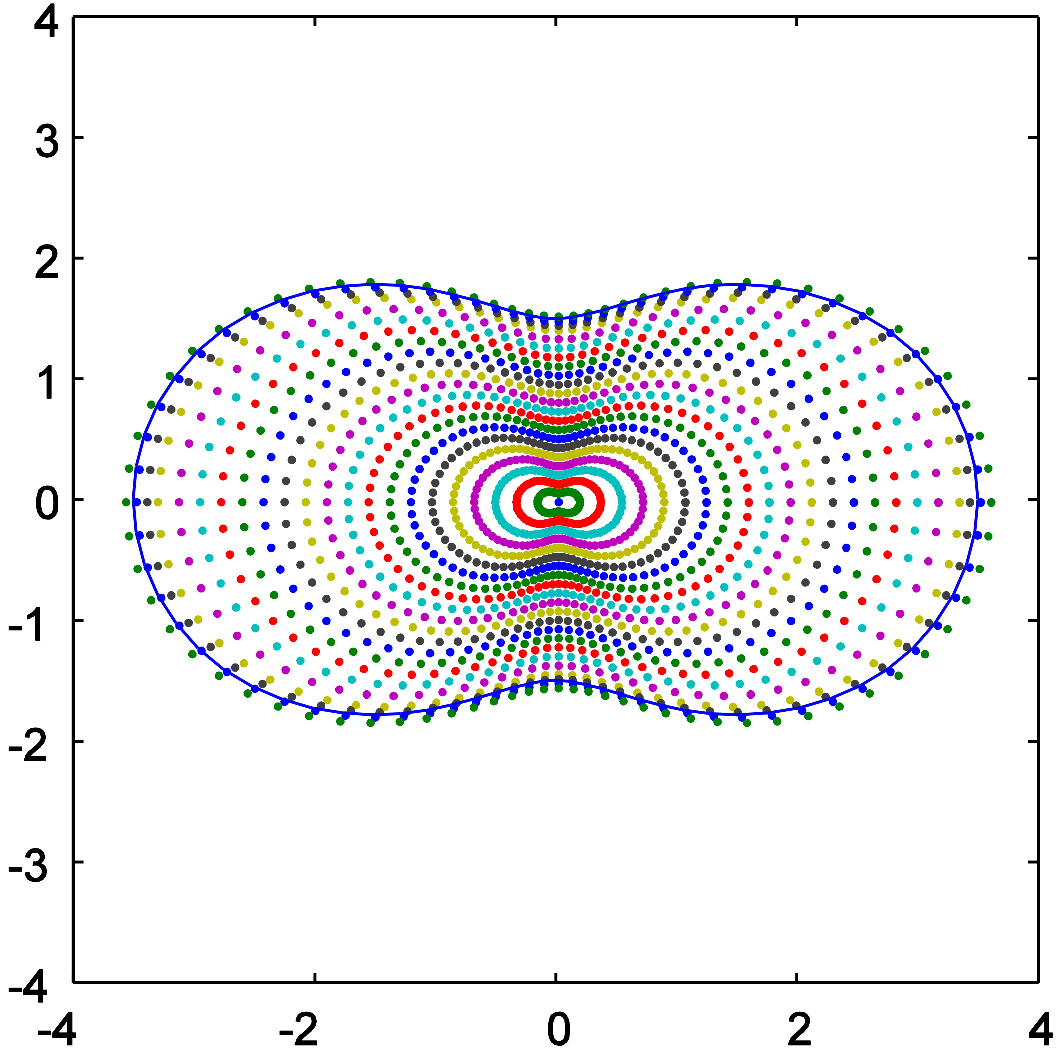
Critical Points 3 minutes with 200 cores



Tangent Cone and Jumping off Crit Point



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

$$(1.9)\qquad -\Delta\sigma + \sigma = 0 \quad \text{in } \Omega, \qquad \sigma = 1 \quad \text{on } \partial\Omega,$$

$$(1.10)\qquad -\Delta\vec{v} + \nabla p = (\mu/3)\nabla(\sigma - \tilde{\sigma}) \quad \text{in } \Omega,$$

$$(1.11)\qquad \operatorname{div}\vec{v} = \mu(\sigma - \tilde{\sigma}) \quad \text{in } \Omega \quad (\tilde{\sigma} < 1),$$

$$(1.12)\qquad T(\vec{v}, p)\vec{n} = \Big(-\gamma\kappa + \frac{2\nu}{3}\mu(1 - \tilde{\sigma})\Big)\vec{n} \quad \text{on } \partial\Omega,$$

$$(1.13)\qquad \vec{v}\cdot\vec{n} = 0 \quad \text{on } \partial\Omega,$$

$$(1.14)\qquad \int_\Omega \vec{v}\,dx = 0, \qquad \int_\Omega \vec{v}\times\vec{x}\,dx = 0,$$

where $T(\vec{v}, p) = (\nabla\vec{v})^T + \nabla\vec{v} - p\,I$, $I = (\delta_{ij})_{i,j=1}^3$.

Governing equations:

- Diffusion of the nutrients: $\sigma_t - \Delta\sigma + \sigma = 0 \quad \text{in } \Omega(t)$.
- Conservation of mass: $\text{div}\vec{V} = S$, $S =$ 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: $-\nu\Delta\vec{v} + \nabla p - \frac{1}{3}\nu\nabla\text{div}\vec{v} = 0 \quad \text{in } \Omega(t)$.
- Introducing the stress tensor $Q = \nu(\nabla\vec{v} + (\nabla\vec{v})^T) - (p + \frac{2}{3}\nu\text{div}\vec{v})I$ with components $Q_{ij} = \nu\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right) - \delta_{ij}\left(p + \frac{2\nu}{3}\text{div}\vec{v}\right)$, we then have

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

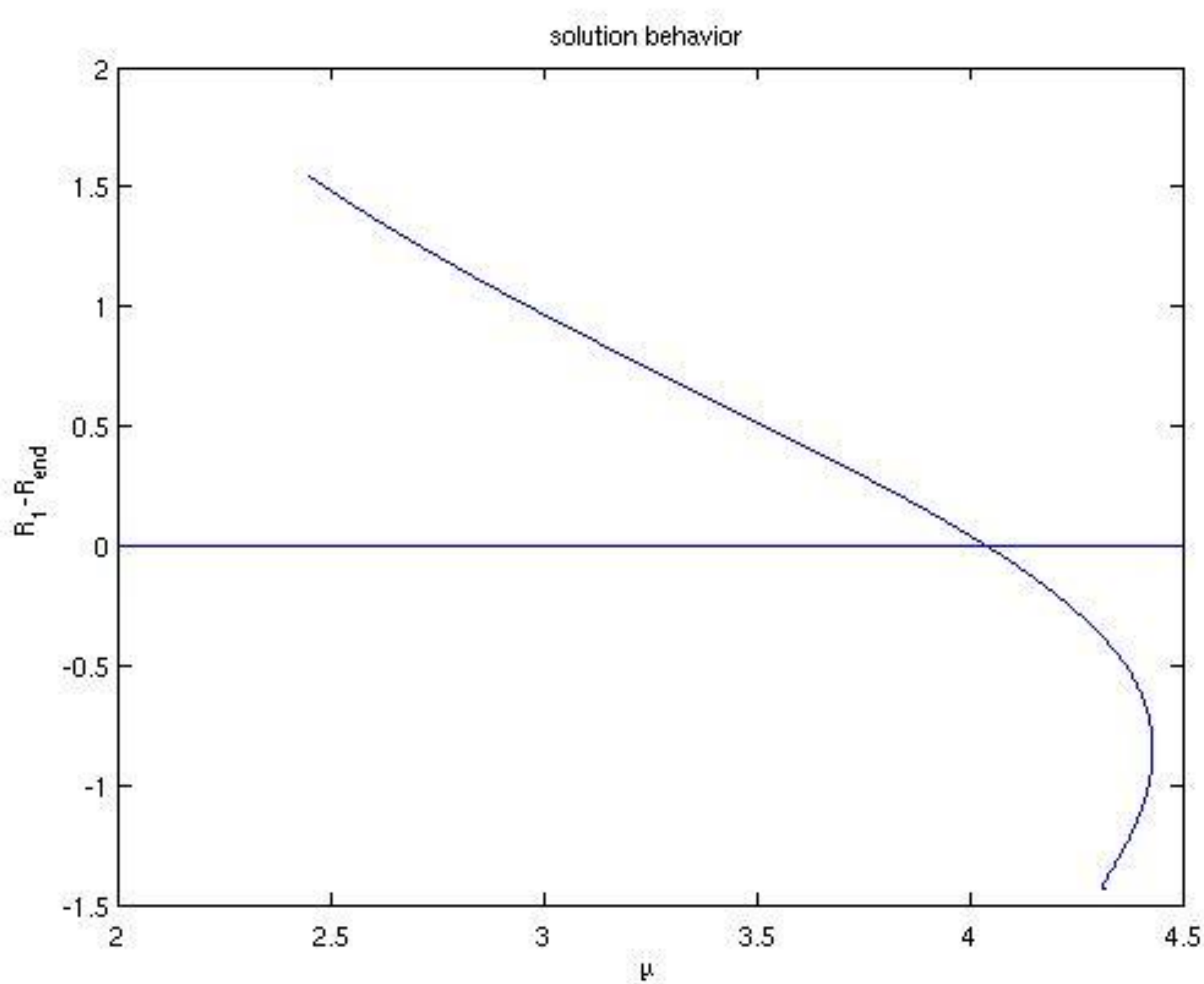
  here the cell-to-cell adhesion equal to a constant γ, κ is the mean curvature.
- Continuity: $V_n = \vec{v}\cdot\vec{n} \quad \text{on } \partial\Omega(t)$
  where V_n= velocity in the normal n direction.

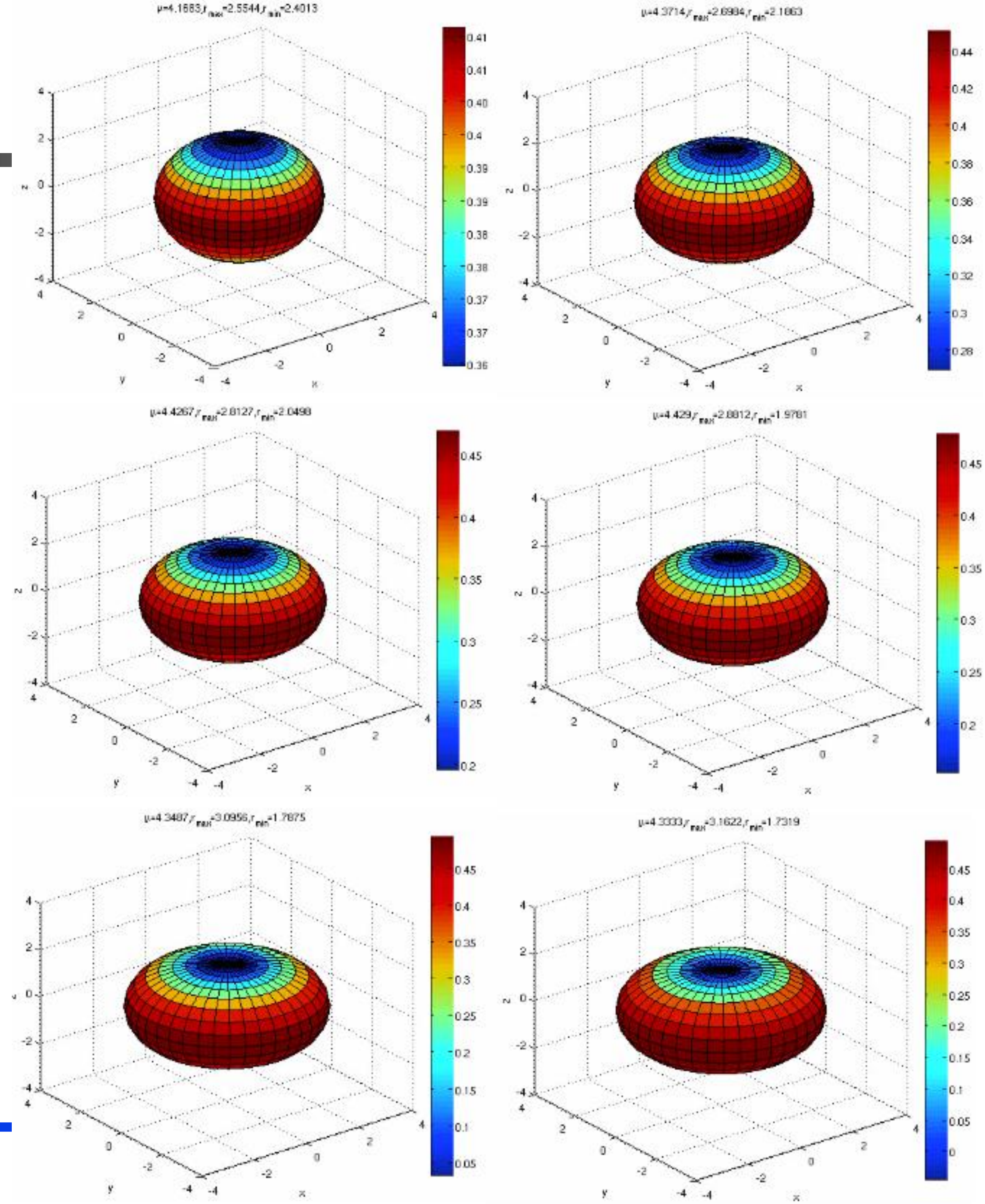
Since $\vec{v}$ is determined up to $\vec{b}\times\vec{x}$, some additional constraints are needed.

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

subject to the constraints

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


solution behavior
2
1.5
1
0.5
0
-0.5
-1
-1.5
R1-Rend
2
2.5
3
3.5
4
4.5
μ



μ=4.1683,r_{max}=2.5544,r_{min}=2.4013
μ=4.3714,r_{max}=2.6984,r_{min}=2.1863
μ=4.4267,r_{max}=2.8127,r_{min}=2.0498
μ=4.429,r_{max}=2.8812,r_{min}=1.9781
μ=4.3487,r_{max}=3.0956,r_{min}=1.7875
μ=4.3333,r_{max}=3.1622,r_{min}=1.7319

- Two-dimensional tumor movie
- Three-dimensional tumor movie
- Thre-dimensional tumor movie with dead core

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.