# Sweeping for singular solutions of polynomial systems with parameters

Kathy Piret    Jan Verschelde

University of Illinois at Chicago
Department of Mathematics, Statistics, and Computer Science
`http://www.math.uic.edu/~kathywang`   `http://www.math.uic.edu/~jan`
`kathywang@math.uic.edu`   `jan@math.uic.edu`

ACA 2009 Session on Algorithms for Parametric Systems and their Applications. ÉTS, Montréal, Canada, 25-28 June 2009.

# Outline

1. Introduction and Problem Statement
   - solving polynomial systems with homotopy continuation
   - reconditioning singularities with deflation
   - global and local problems
   - detection and location of quadratic turning points
2. Detection of Singularities
   - a neural network model with straight solution paths
   - Puiseux series and the determinant criterion
   - parabolic interpolation of determinants
3. Applications
   - three polynomial systems from the literature

# Solving Polynomial Systems

## numerical algebraic geometry: numerical analysis and algebraic geometry

Polynomial systems are nonlinear systems with algebraic structure.
This algebraic structure enables to compute

- not only **all** isolated solutions,
- but also **a numerical irreducible decomposition**
  $\longrightarrow$ degrees and dimensions of all irreducible components.

Two key references:


1. **Tien-Yien Li.** Numerical solution of polynomial systems by homotopy continuation methods. In Volume XI of *Handbook of Numerical Analysis*, pages 209–304, 2003.
2. **Andrew J. Sommese and Charles W. Wampler.** *The Numerical Solution of Systems of Polynomials Arising in Engineering and Science.* World Scientific, 2005.

# Homotopy Continuation Methods

## natural and artificial parameter homotopies

A **homotopy** $h$ is a family of systems, depending on a parameter.
With **continuation** methods we track solution paths defined by $h$.
We distinguish between two types of parameters:

1. a natural parameter $\lambda$, for example:

$$h(\lambda, x) = \lambda^2 + x^2 - 1 = 0.$$

   As $\lambda$ varies we track the unit circle: $(\lambda, x(\lambda)) \in h^{-1}(0)$.

2. an artificial parameter $t$, for example:

$$h(t, \lambda, x) = \left\{ \begin{array}{r} \lambda^2 + x^2 - 1 = 0 \\ (\lambda - 2)t + (\lambda + 2)(1 - t) = 0. \end{array} \right.$$

   As $t$ moves from 0 to 1, $\lambda$ goes from $-2$ to $+2$
   and we ***sweep*** points $(\lambda(t), x(\lambda(t)))$ on the unit circle.

# Reconditioning Singularities via Deflation

## restoring the quadratic convergence of Newton's method

A solution $\mathbf{z}$ to $f(\mathbf{x}) = \mathbf{0}$, $f = (f_1, f_2, \ldots, f_N)$, $\mathbf{x} = (x_1, x_2, \ldots, x_n)$, $N > n$, is **singular** if the Jacobian matrix $A(\mathbf{x}) = \left[ \frac{\partial f_i}{\partial x_j} \right]$ has rank $R < n$ at $\mathbf{z}$.

Choose $\mathbf{c} \in \mathbb{C}^{R+1}$ and $\mathbf{B} \in \mathbb{C}^{n \times (R+1)}$ at random.
Introduce $R + 1$ new multiplier variables $\boldsymbol{\mu} = (\mu_1, \mu_2, \ldots, \mu_{R+1})$.
Apply the Gauss-Newton method to

$$\left\{ \begin{array}{rcl} f(\mathbf{x}) & = & \mathbf{0} \\ A(\mathbf{x})\mathbf{B}\boldsymbol{\mu} & = & \mathbf{0} \\ \mathbf{c}\boldsymbol{\mu} & = & 1 \end{array} \right. \qquad \begin{array}{c} \mathrm{Rank}(A(\mathbf{x})) = R \\ \Downarrow \\ \mathrm{coRank}(A(\mathbf{x})\mathbf{B}) = 1 \end{array}$$

Recurse if necessary, $\#$*deflations* $<$ multiplicity.
An efficient implementation uses algorithmic differentiation.

# Problems and Applications

## some hard motiviating questions

General problem statement:

> Given a polynomial system $f(\boldsymbol{\lambda}, \mathbf{x}) = 0$, $\boldsymbol{\lambda} \in \mathbb{C}^m$, $\mathbf{x} \in \mathbb{C}^n$, find values $\boldsymbol{\lambda}$ for which solutions $\mathbf{x}$ are singular.

Two motivating questions:

- from real algebraic geometry:
  $\rightarrow$ *can all complex solutions turn* ***real****?*
- from numerical algebraic geometry:
  $\rightarrow$ *what are the* ***real*** *irreducible solution components?*

# Complexity Issues

## of local and global solutions

Solving the global problem

> Given a polynomial system $f(\boldsymbol{\lambda}, \mathbf{x}) = 0$, $\boldsymbol{\lambda} \in \mathbb{C}^m$, $\mathbf{x} \in \mathbb{C}^n$,
> find values $\boldsymbol{\lambda}$ for which solutions $\mathbf{x}$ are singular.

involves a description of **the discriminant variety**
and the solution of more difficult polynomial systems.

Instead we consider a **local** problem, for *one* parameter $\lambda$:

| | |
|---|---|
| Given | a polynomial system $f(\lambda, \mathbf{x}) = 0$, $\boldsymbol{\lambda} \in \mathbb{C}^m$, $\mathbf{x} \in \mathbb{C}^n$, |
| | a solution $\mathbf{z}$ for $\lambda = \lambda_0$ and target value $\lambda_1$, |
| find | either the solution $\mathbf{z}$ for $\lambda = \lambda_1$ |
| | if no singularities for all $\lambda(t) = (1-t)\lambda_0 + t\lambda_1$, |
| | or the first $(t, \lambda(t), \mathbf{x}(t))$ for which $\mathbf{z} = (\lambda(t), \mathbf{x}(t))$ is singular. |

# References

## numerical methods


1. **W.J.F. Govaerts.** *Numerical Methods for Bifurcations of Dynamical Equilibria*. SIAM, 2000.
2. **Z. Mei.** *Numerical Bifurcation Analysis for Reaction-Diffusion Equations*. Springer, 2000.
3. **P. Kunkel.** A tree-based analysis of a family of augmented systems for the computation of singular points. *IMA J. Numer. Anal.* 1996.
4. **T.Y. Li and Z. Zeng.** Homotopy continuation algorithm for the real nonsymmetric eigenproblem: Further development and implementation. *SIAM J. Sci. Comput.* 1999.
5. **A. Leykin, J. Verschelde, and A. Zhao.** Newton's method with deflation for isolated singularities of polynomial systems. *Theoretical CS* 2006.
6. **Y. Lu, D.J. Bates, A.J. Sommese, and C.W. Wampler.** Finding all real points of a complex curve. *Contemporary Mathematics* 448: 183–206, 2007.

# Quadratic Turning Points

most common type of singularity

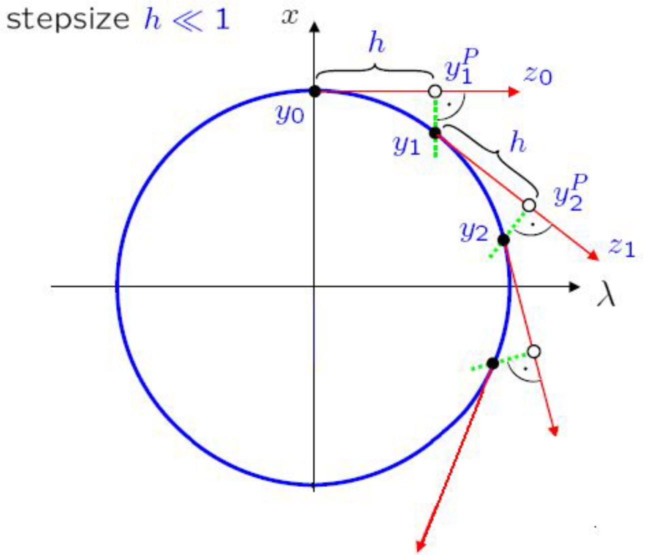
1. **Definition:** solution paths turn back
   when the parameter increases past a quadratic turning point.

   Properties: a double solution, corank of Jacobian equals one,
   transition point: complex $\leftrightarrow$ real.

2. **Detection:** monitor orientation of tangent vectors.
   Two consecutive tangent vectors $\mathbf{v}(t_1)$ and $\mathbf{v}(t_2)$.
   Criterion: $\langle \mathbf{v}(t_1), \mathbf{v}(t_2) \rangle < 0 \Rightarrow \mathbf{v}(t) \perp t-\mathrm{axis}\ \ \mathrm{for}\ t \in [t_1, t_2]$.
   Tangents are simple byproduct of predictor-corrector path tracker.

3. **Location:** shooting method for step size.
   Consider $\mathbf{x}(t) = \mathbf{x}(t_1) + h\,\mathbf{v}(t_1)$, find $h$ and $t$: $\mathbf{v}(t) \perp t-\mathrm{axis}$.
   Overshot turning point for $h = h_2$, at $\mathbf{x}(t_2)$ path has turned back.

# Sweeping a Circle



# Difficulties to Extend Approach

## for any type of isolated singularity along a path

Detecting and locating quadratic turning points goes well.

Extending to any type of singularity has two difficulties:

1. detection: flip of tangent orientation no longer suffices
   $\rightarrow$ the path tracker glides over the singularity
2. location: higher order singularities may have corank $> 1$
   $\rightarrow$ the path tracker fails to converge

Solutions for these difficulties:

1. use a Jacobian criterion for detection, and
2. algebraic higher order predictor for location.

Common tool: Puiseux series expansion at a point along the path.

# Neural Network Model

## a family of polynomial systems for any dimension $n$

**V.W. Noonburg.** A neural network modeled by an adaptive Lotka-Volterra system. *SIAM J. Appl. Math.* 1989.

- Applying a sweep to the polynomial systems:

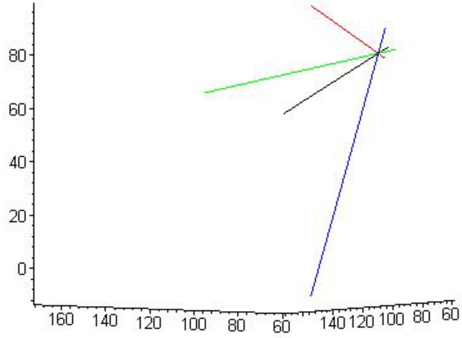
$$f(x,\lambda) = \begin{cases} x_1 x_2^2 + x_1 x_3^2 - \lambda x_1 + 1 = 0 \\ x_2 x_1^2 + x_2 x_3^2 - \lambda x_2 + 1 = 0 \\ x_3 x_1^2 + x_3 x_2^2 - \lambda x_3 + 1 = 0 \\ (\lambda + 1)(1 - t) + (\lambda - 1)t = 0 \end{cases}$$

- As t goes from 0 to 1, $\lambda$ goes from $-1$ to $+1$.

- The tangent does not flip at the origin.
  The path tracker does not detect the quadruple point for $\lambda = 0$.

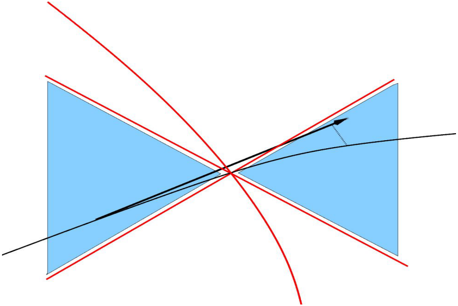
# The Plot of Solution Paths for Neural Networks

the solution paths are really straight



# Jumping Over Singularities

## in case of jumping over a bifurcation point [Z. Mei]



The shaded blue part is the region where Newton's method converges.
On straight curves, the path tracker will never cut back its step size.

# Puiseux or Fractional Power Series

## expanding an algebraic curve at a point

The homotopy $h(\mathbf{x}, t) = \mathbf{0}$ defines solution paths $\mathbf{x}(t)$: $h(\mathbf{x}(t), t) \equiv \mathbf{0}$.

Because $\mathbf{x}(t)$ is an algebraic curve, at any point $t_*$ the corresponding solution $\mathbf{x}(t_*) = \mathbf{z} = (z_1, z_2, \ldots, z_n)$ admits the expansion:

$$\left\{ \begin{array}{ll} x_k(s) = z_k s^{v_k}(1 + O(s)) & k = 1, 2, \ldots, n, v_k \in \mathbb{Z} \\ s^\omega = t - t_* & \text{as } t \rightarrow t_*, s \rightarrow 0 \end{array} \right.$$

Special case: $t_* = 0$: $s^\omega = t$ or $s = t^{1/\omega}$ and $x_k \rightarrow z_k t^{v_k/\omega}$ as $t \rightarrow 0$.

The ***winding number*** $\omega$ determines how hard the path curves.

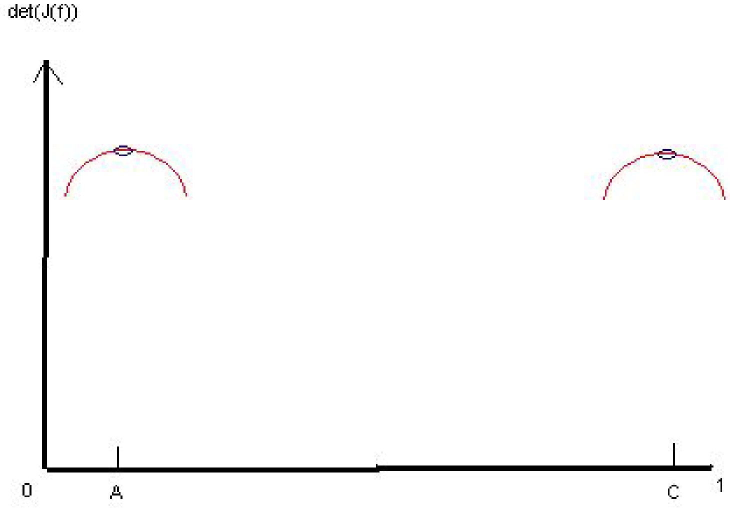
Determinant criterion for singularity along path $\mathbf{x}(t)$:

$$\text{singularity at } t_* \Leftrightarrow \det(A(\mathbf{x}(t_*))) = 0.$$

Via Puiseux series, determinant of Jacobian matrix is function of $t$.

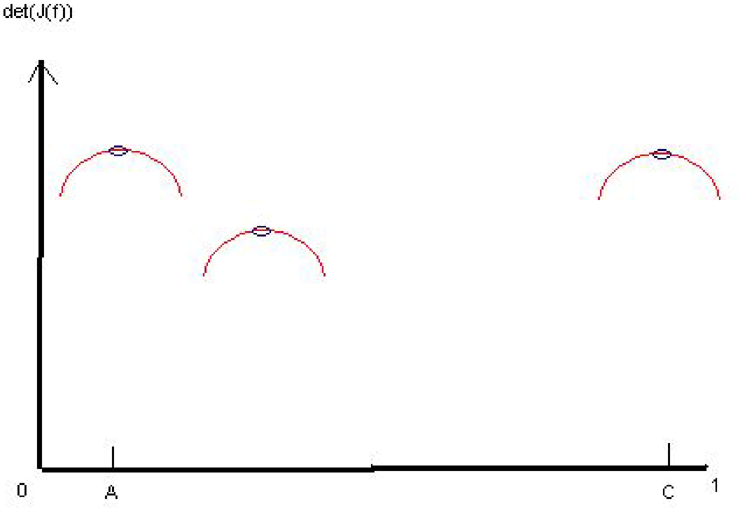
# Parabolic Interpolation of Determinants

monitor concavity of determinant as function of $t$



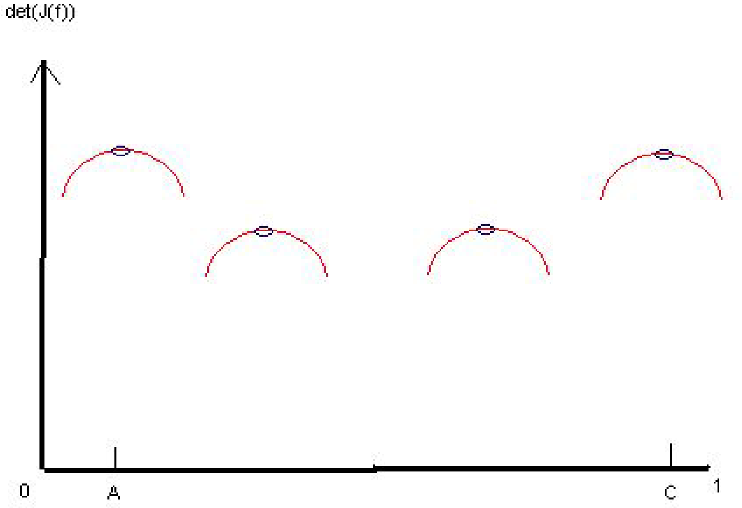
# Parabolic Interpolation of Determinants

monitor concavity of determinant as function of $t$



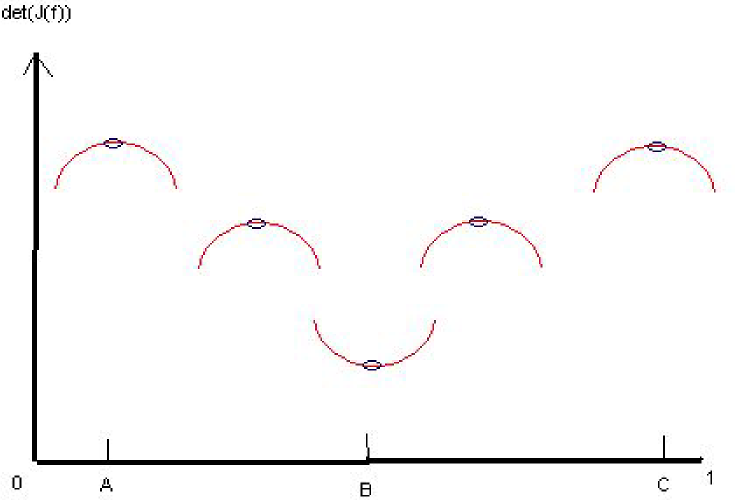
# Parabolic Interpolation of Determinants

monitor concavity of determinant as function of $t$



# Parabolic Interpolation of Determinants

monitor concavity of determinant as function of $t$



# Detection Algorithm Specification

| | |
|---|---|
| Input: $h(\mathbf{x}, t) = \mathbf{0}$; | *a homotopy* |
| $(t_1, t_2, t_3)$, $t_1 < t_2 < t_3$; | *consecutive samples* |
| $(\mathbf{z}_1, \mathbf{z}_2, \mathbf{z}_3)$: $h(\mathbf{z}_i, t_i) = \mathbf{0}$, $i = 1, 2, 3$; | *with solutions* |
| $(d_1, d_2, d_3)$: $d_i = \det(\partial_{\mathbf{x}} h(\mathbf{z}_i, t_i))$, $i = 1, 2, 3$; | *and determinants* |
| $\delta > 0$; | *tolerance on* $t_3 - t_1$ |
| $\epsilon > 0$. | *tolerance on* $\det()$ |
| Output: $(t^*, \mathbf{z}^*, d^*)$, $h(\mathbf{z}^*, t^*) = \mathbf{0}$; | *a solution* |
| $d^* = \det(\partial_{\mathbf{x}} h(\mathbf{z}^*, t^*))$, $\|d^*\| < \epsilon$; | *that is singular* |
| or $\emptyset$, updated $(t_i, \mathbf{z}_i, d_i)$, $i = 1, 2, 3$. | *no singular solution* |

# Detection Algorithm Implementation

| | |
|---|---|
| while $(\lvert d_1 \rvert > \lvert d_2 \rvert < \lvert d_3 \rvert)$ and $(t_3 - t_1 > \delta)$ do | *loop invariants* |
|     $t^* := \min \mathcal{P}((t_1, t_2, t_3), (d_1, d_2, d_3))$; | *parabolic minimum* |
|     $(z^*, d^*) := \mathrm{Newton}(h, t^*, \mathbf{z}_2)$; | *correct solution* |
|     if $\lvert d^* \rvert < \epsilon$ then | *first stop test* |
|         return $(t^*, \mathbf{z}^*, d^*)$; | *found singularity* |
|     else if $\lvert d^* \rvert \geq \lvert d_2 \rvert$ then | *second stop test* |
|         return $\emptyset$; | *no singularity found* |
|     else | *continue loop* |
|         if $t^* < t_2$ | *adjust $t_1$, $t_2$, $t_3$* |
|           then $(t_3, \mathbf{z}_3, d_3) := (t_2, \mathbf{z}_2, d_2)$; | *$t_2$ becomes right end* |
|           else $(t_1, \mathbf{z}_1, d_1) := (t_2, \mathbf{z}_2, d_2)$; | *$t_2$ becomes left end* |
|         end if; | |
|         $(t_2, \mathbf{z}_2, d_2) := (t^*, \mathbf{z}^*, d^*)$; | $d_2$ remains minimum |
|     end if; | |
| end while. | |

# Numerical Stability

For determinant values $d_1$, $d_2$, and $d_3$, respectively at consecutive $t_1$, $t_2$, and $t_3$, $t^* := \min \mathcal{P}((t_1, t_2, t_3), (d_1, d_2, d_3))$ is subject to roundoff error. $t^*$ is computed via

$$T = \frac{t_1^2(d_3 - d_2) + t_2^2(d_1 - d_3) + t_3^2(d_2 - d_1)}{2d_1(t_2 - t_3) + 2d_2(t_3 - t_1) + 2d_3(t_1 - t_2)}.$$

We compute $\widetilde{T}$, replacing in $T$ $d_1$, $d_2$, and $d_3$ respectively by $d_1(1+\epsilon_1)$, $d_2(1+\epsilon_2)$, and $d_3(1+\epsilon_3)$ for errors $\epsilon_1$, $\epsilon_2$, and $\epsilon_3$.

$$\frac{\widetilde{T} - T}{T} = \frac{2\epsilon_1 d_1 t_{23} + 2\epsilon_2 d_2 t_{13} + 2\epsilon_3 d_3 t_{12}}{P}.$$

with $t_{23}$, $t_{13}$, and $t_{12}$ constants of magnitude $> \delta$ and $P = t_1^2(d_3 - d_2) + t_2^2(d_1 - d_3) + t_3^2(d_2 - d_1)$.

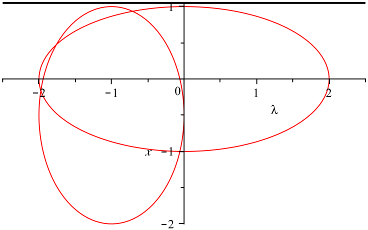
$\Rightarrow$ large relative errors only if $d_1 \approx d_2 \approx d_3$.

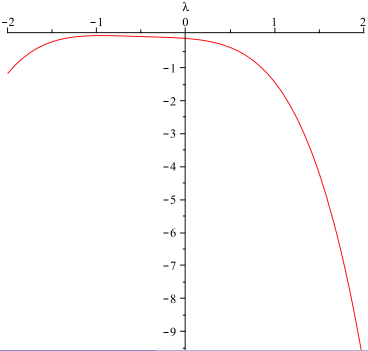
# Numerical Conditioning

Worst case: straight path almost touches ellipses.

$$h(x,\lambda,t) = \left\{ \begin{array}{rcl} (x-1-\epsilon)\left(\frac{\lambda^2}{4}+x^2-1\right) & & \\ \left(\frac{1}{4}(\lambda+1)^2+\frac{4}{9}(x+1/2)^2-1\right) & = & 0 \\ (1-t)(\lambda+2)+t(\lambda-2) & = & 0 \end{array} \right. \qquad t \in [0,1].$$

Plots for $\epsilon = 0.05$:



# Polynomial Systems

## from the literature

1. Molecular Configurations:
   - **Emiris and Mourrain.** Computer algebra methods for studying and computing molecular conformations. *Algorithmica* 1999.
2. Neural Networks:
   - **V.W. Noonburg.** A neural network modeled by an adaptive Lotka-Volterra system. *SIAM J. Appl. Math.* 1989.
3. Symmetrical Stewart-Gough platforms:
   - **Yu Wang and Yi Wang.** Configuration Bifurcations Analysis of Six Degree-of-Freedom Symmetrical Stewart Parallel Mechanism. *Journal of Mechanical Design* 2005.

# Polynomial Systems

## the number of solutions in $C^n$ for generic choices of parameters

| **Polynomial Systems** | **n** | **#Solutions** |
|---|---|---|
| Molecular Configurations | 3 | 16 |
| Neural Networks | 3 | 21 |
| Neural Networks | 4 | 73 |
| Neural Networks | 5 | 233 |
| Neural Networks | 10 | 59049 |
| Neural Networks | 15 | 14,348,907 |
| Symmetrical Stewart-Gough Platforms | 9 | 28 (real) |

Table: Polynomial Systems which have higher-order multiple points

# Molecular Configurations

## applying the sweep homotopy algorithm to this system

- The system is small enough to handle with resultant/symbolic methods or global methods.
- Applying a sweep to molecular configurations:

$$f(x,\lambda) = \begin{cases} \frac{1}{2}(x_2^2 + 4x_2x_3 + x_3^2) + \lambda(x_2^2x_3^2 - 1) = 0 \\ \frac{1}{2}(x_3^2 + 4x_3x_1 + x_1^2) + \lambda(x_3^2x_1^2 - 1) = 0 \\ \frac{1}{2}(x_1^2 + 4x_1x_2 + x_2^2) + \lambda(x_1^2x_2^2 - 1) = 0 \\ (\lambda - 1)(1-t) + (\lambda + 1)t = 0. \end{cases}$$

- The tangent flips at the higher-order turning point at the origin.
- For $\lambda = \pm 0.866025403780023$ on symmetrical curves of degree 6 and one of the eigenvalues of the Jacobian matrix changes signs.

# Symmetrical Stewart-Gough platforms

## nine quadratic polynomial equations

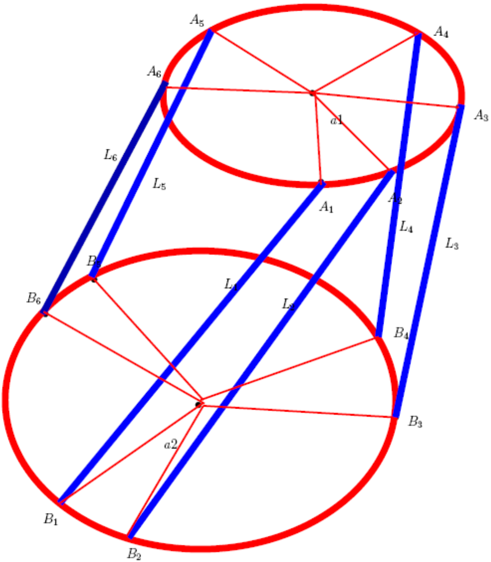
$$f(x, L_1) = \begin{cases} f_i = (x_i - x_{i0})^2 + (y_i - y_{i0})^2 + z_i^2 - L_i^2, i = 1, 2, \ldots, 6 \\ f_7 = (x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2 - 2R_1^2(1 - \beta)) \\ f_8 = (x_1 - x_0)^2 + (y_1 - y_0)^2 + (z_1 - z_0)^2 - R_1^2 \\ f_9 = (x_2 - x_0)^2 + (y_2 - y_0)^2 + (z_2 - z_0)^2 - R_1^2 \end{cases}$$

where

$$\begin{cases} x_i = w_1 x_0 + w_2^{m_1} w_3^{m_2} x_1 + w_2^{m_2} w_3^{m_1} x_2 \\ y_i = w_1 y_0 + w_2^{m_1} w_3^{m_2} y_1 + w_2^{m_2} w_3^{m_1} y_2 \\ z_i = w_1 z_0 + w_2^{m_1} w_3^{m_2} z_1 + w_2^{m_2} w_3^{m_1} z_2 \end{cases}$$

See Wang and Wang's paper for details of the system.

# Symmetrical Stewart-Gough platforms



# Computational Results

on the symmetrical Stewart-Gough platforms

- Applying the Jacobian criterion globally leads to an augmented system with a mixed volume equal to 4,608.
  Tracking 4,608 paths in 16 variables is much more expensive than tracking 512 paths in 9 variables.
  Sweeping to find all critical points works in a more efficient setup: at most 28 paths in 9 variables.

- By fixing $L_i, i = 2, 3, \ldots, 6$, to 1.5, 2.0, and 3.0, the sweep yields four special values for the natural parameter $L_1$ for each $L_i$.

- We have replicated the results from Wang and Wang's paper with higher precision than what they reported.
  In addition, $z_0$ can be either positive or negative.