Numerical Continuation - A step by step introduction

Prelude to the Münsteranian Torturials at

https://www.uni-muenster.de/CeNoS/Lehre/Tutorials/continuation.html

Uwe Thiele

Institut für Theoretische Physik, Universität Münster

Context

- Stand-alone lecture on **Continuation techniques**
- Given in the context of a lecture course *Introduction to the theory of phase transitions*
- Introductory lecture tailored at Bachelor/Master students, possibly also useful for beginning PhD
- Sufficiently detailed to enable everyone to create their own numerical continuation code
- Accompanied by hands-on tutorials hosted at www.uni-muenster.de/CeNoS/Lehre/Tutorials/continuation.html – aka *Münsteranian Torturials*

Aims part 1

- Numerical analysis of bifurcation problems - motivation
- Recap Newton's method in one dimension (root finding)
- Multi-dimensional Newton's method
- Simple parameter continuation
  - General scheme
  - Tangent predictor
  - Newton corrector
- Example problem (Predator-prey model)

Literatur I

[1] E. Doedel, H. B. Keller, and J. P. Kernevez. Numerical analysis and control of bifurcation problems (II) Bifurcation in infinite dimensions. *Int. J. Bifurcation Chaos*, 1:745–72, 1991. doi: 10.1142/S0218127491000555.

[2] E. J. Doedel and B. E. Oldeman. *AUTO07p: Continuation and bifurcation software for ordinary differential equations*. Concordia University, Montreal, 2009.

[3] B. Krauskopf, H. M. Osinga, and J Galan-Vioque, editors. *Numerical Continuation Methods for Dynamical Systems*. Springer, Dordrecht, 2007.

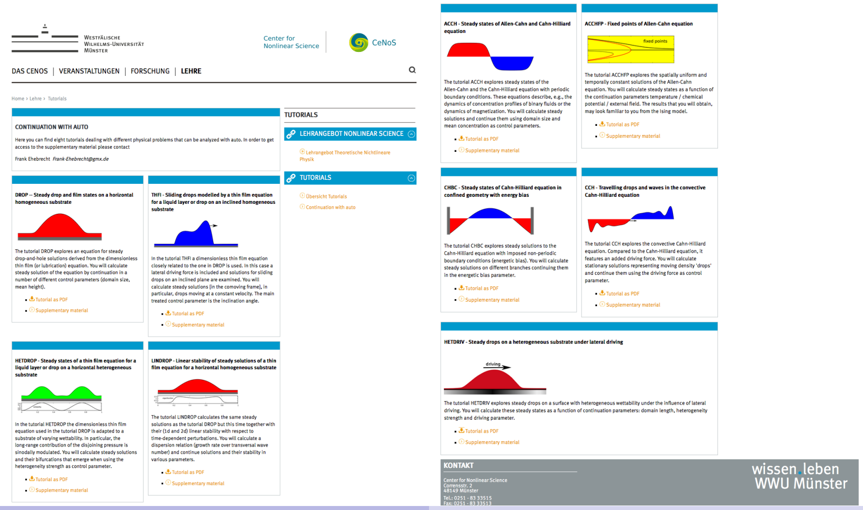
[4] Y. A. Kuznetsov. *Elements of Applied Bifurcation Theory*. Springer, New York, 3rd edition, 2010.

[5] H. A. Dijkstra, F. W. Wubs, A. K. Cliffe, E. Doedel, I. F. Dragomirescu, B. Eckhardt, A. Y. Gelfgat, A. Hazel, V. Lucarini, A. G. Salinger, E. T. Phipps, J. Sanchez-Umbria, H. Schuttelaars, L. S. Tuckerman, and U. Thiele. Numerical bifurcation methods and their application to fluid dynamics: Analysis beyond simulation. *Commun. Comput. Phys.*, 15:1–45, 2014. doi: 10.4208/cicp.240912.180613a.

Münsteranian Torturials: Continuation (1d)

hosted by Center of Nonlinear Science (CeNoS) of WWU Münster

http://www.uni-muenster.de/CeNoS/Lehre/Tutorials/auto.html



Numerical analysis by direct time simulation

- Normally known how to solve Initial Value Problems (e.g. for n first order ODE)

$$
\begin{aligned}
\frac{dy_1}{dt} &= f_1(y_1, y_2, \dots, y_n) \\
\frac{dy_2}{dt} &= f_2(y_1, y_2, \dots, y_n) \\
&\vdots \\
\frac{dy_n}{dt} &= f_n(y_1, y_2, \dots, y_n)
\end{aligned}
$$

  using single- and multi-step (time-stepping) methods, e.g., 4th order Runge-Kutta

Numerical analysis by direct time simulation

- Convenient vector notation for system of n equations:

$$\frac{d\widehat{\mathbf{y}}}{dt} = \widehat{\mathbf{F}}(\widehat{\mathbf{y}})$$

  with $\widehat{\mathbf{y}} = (y_1, y_2, \dots, y_n)^T$ and $\widehat{\mathbf{F}} = (f_1, f_2, \dots, f_n)^T$

- Time-stepping methods
  - \+ give valuable insight into transient behaviour
  - \+ predict system behaviour for given particular parameter values and initial conditions (IC)
  - – are tedious, when one is interested in the entire solution structure of the given nonlinear ODE and its change with parameter(s)
  - – are unable to determine unstable solutions, i.e., can only give an incomplete picture of bifurcations

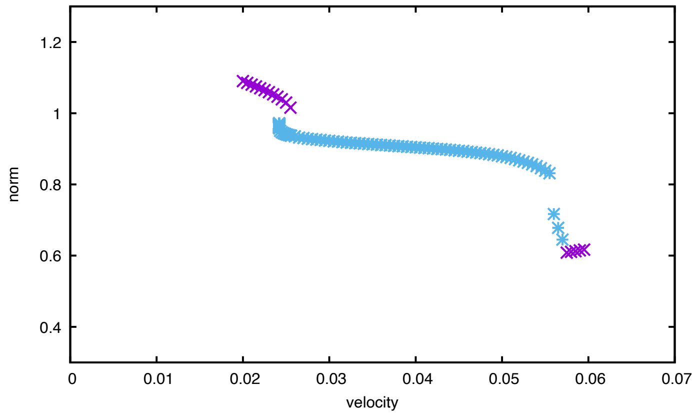
Path-continuation and bifurcation techniques

- Effective way to study the bifurcation structure of ODE (and PDE) and its change with control parameter(s)
- Powerful set of techniques based on bifurcation theory that allow one to
  - Follow steady and stationary states in parameter space (parameter continuation)
  - Determine stability of such states
  - Identify bifurcation points, i.e. loci where new branches of states emerge (might be time-periodic states, states of different symmetry, etc.)
  - Follow 'new' branches in parameter space.
  - Follow bifurcation points in (higher dimensional) parameter space
- $\rightarrow$ more complete characterisation of system behaviour, and for a better understanding of transitions to more complex behaviour

Example - Langmuir-Blodgett transfer

Timestepping

$\rightarrow$ stable steady & time-periodic states, incl. hysteresis

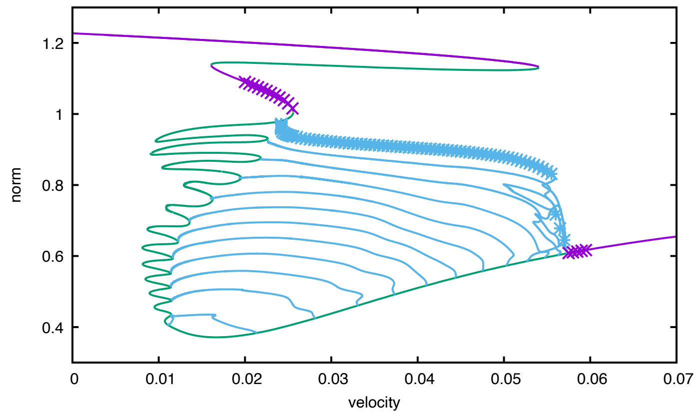


Data from M. Köpf, S. Gurevich, R. Friedrich, UT, New J. Phys. 14, 023016 (2012); M. Köpf and UT, Nonlinearity **27**, 2711 (2014)

Example - Langmuir-Blodgett transfer

Continuation (lines) & Timestepping (symbols)

$\rightarrow$ full bifurcation structure



Data from M. Köpf, S. Gurevich, R. Friedrich, UT, New J. Phys. 14, 023016 (2012); M. Köpf and UT, Nonlinearity **27**, 2711 (2014)

General problem setting for parameter continuation

- Assume we have a system of n unknowns y_i depending on one free parameter λ

$$\frac{d\hat{\mathbf{y}}}{dt} = \hat{\mathbf{G}}(\hat{\mathbf{y}}, \lambda)$$

- For each λ, steady states $\hat{\mathbf{y}}$ (equilibria, fixed points) might exist, defined by

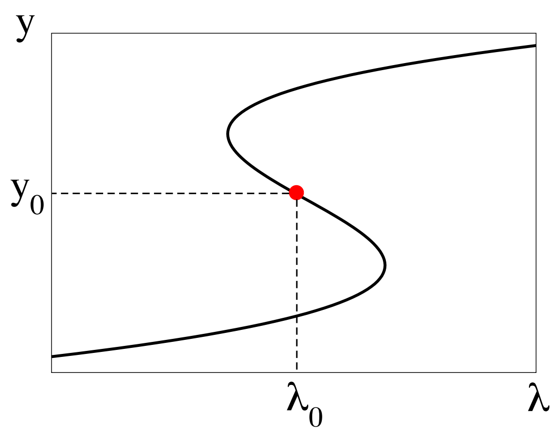
$$\frac{d\hat{\mathbf{y}}}{dt} = 0 \qquad \text{i.e.} \qquad \hat{\mathbf{G}}(\hat{\mathbf{y}}, \lambda) = 0$$

Denote one fixed point at some $\lambda = \lambda_0$ by $\hat{\mathbf{y}}_0$, i.e.,

$$\hat{\mathbf{G}}(\hat{\mathbf{y}}_0, \lambda_0) = 0$$

General problem setting for parameter continuation

- One can show that any regular solution (to be defined later) $(\hat{\mathbf{y}}_0, \lambda_0)$ lies on (is part of) a unique one-dimensional continuum of states (also called a solution branch)



Recap: Newton's method – 1 dimension

- Definition of forward difference scheme for f':

$$f'(y^{(i)}) = \frac{f(y^{(i+1)}) - f(y^{(i)})}{y^{(i+1)} - y^{(i)}} \qquad \text{with} \qquad f(y^{(i+1)}) = 0$$

- gives iterative method for finding solution of equation $f(y) = 0$ using steps:

$$y^{(i+1)} = y^{(i)} - \frac{f(y^{(i)})}{f'(y^{(i)})}$$

- Also write as "linear inhomogeneous equation" $f'(y^{(i)})\Delta y^{(i)} = -f(y^{(i)})$ with $\Delta y^{(i)} = y^{(i+1)} - y^{(i)}$

Recap: Newton's method – n dimensions

- Multi-dimensional equivalent can be used to find solutions $\widehat{\mathbf{y}}$ of the system of equations (for particular λ)

$$\widehat{\mathbf{G}}(\widehat{\mathbf{y}}, \lambda) = 0$$

using the iterative procedure

$$\widehat{\mathbf{G}}_{\widehat{\mathbf{y}}}(\widehat{\mathbf{y}}^{(i)}, \lambda)\, \Delta\widehat{\mathbf{y}}^{(i)} = -\widehat{\mathbf{G}}(\widehat{\mathbf{y}}^{(i)}, \lambda) \qquad (\star)$$

that gives a (hopefully) converging set of vectors $(\widehat{\mathbf{y}}^{(0)}, \widehat{\mathbf{y}}^{(1)}, \widehat{\mathbf{y}}^{(2)}, \widehat{\mathbf{y}}^{(3)}, \dots)$

- $\widehat{\mathbf{G}}_{\widehat{\mathbf{y}}}(\widehat{\mathbf{y}}^{(i)}, \lambda)$ is the Jacobian matrix

Recap: Jacobian

- The Jacobian $\widehat{\mathbf{G}}_{\widehat{\mathbf{y}}}(\widehat{\mathbf{y}}^{(i)}; \lambda)$ is the 1st derivative of vector $\widehat{\mathbf{G}}$ w.r.t. the vector $\widehat{\mathbf{y}}$:

$$\widehat{\mathbf{G}}_{\widehat{\mathbf{y}}} = \frac{\partial \widehat{\mathbf{G}}}{\partial \widehat{\mathbf{y}}} = \begin{pmatrix} \frac{\partial G_1}{\partial y_1} & \frac{\partial G_1}{\partial y_2} & \cdots & \frac{\partial G_1}{\partial y_n} \\ \frac{\partial G_2}{\partial y_1} & \ddots & & \vdots \\ \vdots & & \ddots & \vdots \\ \frac{\partial G_n}{\partial y_1} & \cdots & \cdots & \frac{\partial G_n}{\partial y_n} \end{pmatrix}$$

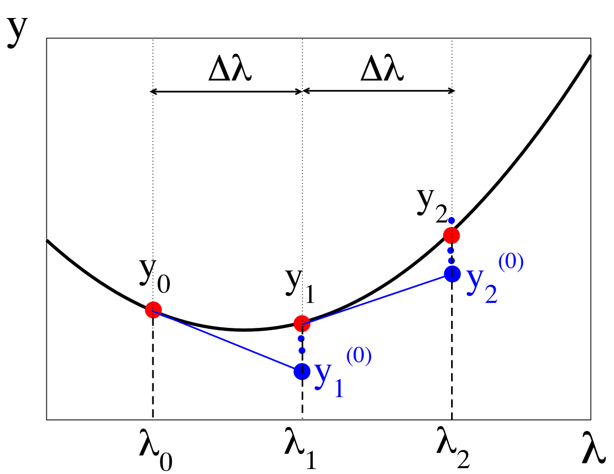
calculated at $(\widehat{\mathbf{y}}^{(i)}, \lambda)$

- Index notation: $(\widehat{\mathbf{G}}_{\widehat{\mathbf{y}}})_{lk} = \partial G_l / \partial y_k = G_{l,k}$

Simple parameter continuation - 1st step

- Determine $(\widehat{\mathbf{y}}_0, \lambda_0)$ that solves $\widehat{\mathbf{G}}(\widehat{\mathbf{y}}_0, \lambda_0) = 0$ (e.g., clever choice of λ where solution is known analytically)
- Want to obtain the solution $(\widehat{\mathbf{y}}_1, \lambda_1)$ at $\lambda_1 = \lambda_0 + \Delta\lambda$
- Strategy
  1. Use tangent of curve $\widehat{\mathbf{y}}(\lambda)$ at point $(\widehat{\mathbf{y}}_0, \lambda_0)$ to obtain predictor $\widehat{\mathbf{y}}_1^{(0)}$ for $\widehat{\mathbf{y}}_1$ at λ_1.
  2. Use Newton's method to iterate at fixed $\lambda = \lambda_1$ (starting with $\widehat{\mathbf{y}}_1^{(0)}$) and obtain $\widehat{\mathbf{y}}_1$ to arbitrary exactness.

Simple parameter continuation - sketch



Simple parameter continuation - step $j+1$

- Take result of previous step $(\widehat{\mathbf{y}}_j, \lambda_j)$ that is a numerical approximation that solves $\widehat{\mathbf{G}}(\widehat{\mathbf{y}}_j, \lambda_j) = 0$.
- We want to obtain the solution $(\widehat{\mathbf{y}}_{j+1}, \lambda_{j+1})$ at $\lambda_{j+1} = \lambda_j + \Delta\lambda$
- Strategy
  (1) Use tangent of curve $\widehat{\mathbf{y}}(\lambda)$ at point $(\widehat{\mathbf{y}}_j, \lambda_j)$ as predictor $\widehat{\mathbf{y}}_{j+1}^{(0)}$ for $\widehat{\mathbf{y}}_{j+1}$ at λ_{j+1}.
  (2) Use Newton's method to iterate at fixed λ_{j+1} (starting with $\widehat{\mathbf{y}}_{j+1}^{(0)}$) and obtain $\widehat{\mathbf{y}}_{j+1}$ to arbitrary exactness.
- Then repeat to change λ to continue along solution branch

Parameter continuation - obtaining the tangent

- How do we get the tangent direction $\partial\widehat{\mathbf{y}}/\partial\lambda$?
- Differentiate $\widehat{\mathbf{G}}(\widehat{\mathbf{y}}(\lambda), \lambda) = 0$ with respect to λ (chain rule):

$$\frac{\partial\widehat{\mathbf{G}}}{\partial\widehat{\mathbf{y}}}\frac{\partial\widehat{\mathbf{y}}}{\partial\lambda} + \frac{\partial\widehat{\mathbf{G}}}{\partial\lambda} = 0$$

- Solve inhomogeneous linear algebraic system of equations

$$\widehat{\mathbf{G}}_{\widehat{\mathbf{y}}}\frac{\partial\widehat{\mathbf{y}}}{\partial\lambda} = -\widehat{\mathbf{G}}_\lambda \qquad (\star\star)$$

  for tangent vector $\partial\widehat{\mathbf{y}}/\partial\lambda$.
- Notation:

$$\widehat{\mathbf{G}}_{\widehat{\mathbf{y}}} = \frac{\partial\widehat{\mathbf{G}}}{\partial\widehat{\mathbf{y}}} \quad \text{and} \quad \widehat{\mathbf{G}}_\lambda = \frac{\partial\widehat{\mathbf{G}}}{\partial\lambda}$$

Tangent predictor step

1. Start with $\widehat{\mathbf{y}}_j$ at λ_j.
2. Get tangent direction $\frac{\partial\widehat{\mathbf{y}}}{\partial\lambda}|_j$ at $(\widehat{\mathbf{y}}_j, \lambda_j)$ solving $(\star\star)$, i.e.

$$\widehat{\mathbf{G}}_{\widehat{\mathbf{y}}}(\widehat{\mathbf{y}}_j, \lambda_j)\left.\frac{\partial\widehat{\mathbf{y}}}{\partial\lambda}\right|_j = -\widehat{\mathbf{G}}_\lambda(\widehat{\mathbf{y}}_j, \lambda_j)$$

   (and normalising $|\partial\widehat{\mathbf{y}}/\partial\lambda| = 1$ - here not done)
3. Obtain initial 'guess' $\widehat{\mathbf{y}}_{j+1}^{(0)}$ at $\lambda_{j+1} = \lambda_j + \Delta\lambda$ by

$$\widehat{\mathbf{y}}_{j+1}^{(0)} = \widehat{\mathbf{y}}_j + \Delta\lambda\left.\frac{\partial\widehat{\mathbf{y}}}{\partial\lambda}\right|_j$$

Newton correction at fixed λ_{j+1}

1. Take initial guess $\hat{\mathbf{y}}_{j+1}^{(0)}$ at λ_{j+1}.
2. Obtain next iteration by solving inhomogeneous linear algebraic system of equations
$$\hat{\mathbf{G}}_{\hat{\mathbf{y}}}(\hat{\mathbf{y}}_{j+1}^{(i)}, \lambda_{j+1})\, \Delta\hat{\mathbf{y}}_{j+1}^{(i)} = -\hat{\mathbf{G}}(\hat{\mathbf{y}}_{j+1}^{(i)}, \lambda_{j+1})$$
for $\Delta\hat{\mathbf{y}}_{j+1}^{(i)} = \hat{\mathbf{y}}_{j+1}^{(i+1)} - \hat{\mathbf{y}}_{j+1}^{(i)}$.
3. Obtain $\hat{\mathbf{y}}_{j+1}^{(i+1)} = \hat{\mathbf{y}}_{j+1}^{(i)} + \Delta\hat{\mathbf{y}}_{j+1}^{(i)}$.

Repeat Newton step (1.-3.) until wanted accuracy is reached (i.e. $||\Delta\hat{\mathbf{y}}_{j+1}^{(i)}||$ smaller than some given threshhold value).

Example: Predator prey model

- In studies of the time-evolution of populations of interacting species one often uses ODEs describing the mean density or overall number of animals
- A typical two-species model is
$$\frac{dy_1}{dt} = g_1(y_1, y_2, \lambda)$$
$$\frac{dy_2}{dt} = g_2(y_1, y_2, \lambda)$$
with
$$g_1(y_1, y_2, \lambda) = 3y_1(1-y_1) - y_1 y_2 - \lambda(1-\exp(-5y_1))$$
$$g_2(y_1, y_2, \lambda) = -y_2 + 3y_1 y_2$$
where y_1 and y_2 are normalized prey and predator numbers, respectively.
- Meaning of terms: eating/getting eaten; getting fished; (saturated) birth/death rates

Example: Predator prey model

- Fixed points / equilibria given by
$$\frac{dy_1}{dt} = \frac{dy_2}{dt} = 0$$
i.e.,
$$0 = 3y_1(1-y_1) - y_1 y_2 - \lambda(1-\exp(-5y_1))$$
$$0 = -y_2 + 3y_1 y_2$$
- For $\lambda = \lambda_0 = 0$ we have three fixed points $\hat{\mathbf{y}} = (y_1, y_2)$
$$(0,0) \qquad (1,0) \qquad \left(\frac{1}{3}, 2\right)$$
- $(0,0)$ is a 'trivial' fixed point valid for any λ
- Let us focus on the 2nd one: $(1,0)$ and continue it for $\lambda > 0$.

Example: Predator prey model

- We need the Jacobian
$$\hat{\mathbf{G}}_{\hat{\mathbf{y}}} = \frac{\partial\hat{\mathbf{G}}}{\partial\hat{\mathbf{y}}} = \begin{pmatrix} 3-6y_1-y_2-5\lambda\exp(-5y_1) & -y_1 \\ 3y_2 & 3y_1-1 \end{pmatrix}$$
and the derivative of $\hat{\mathbf{G}}$ with respect to the continuation parameter
$$\hat{\mathbf{G}}_{\lambda} = \frac{\partial\hat{\mathbf{G}}}{\partial\lambda} = \begin{pmatrix} \exp(-5y_1)-1 \\ 0 \end{pmatrix}$$
- At $\hat{\mathbf{y}}_0 = (1,0)$ and $\lambda_0 = 0$ we have
$$\hat{\mathbf{G}}_{\hat{\mathbf{y}}} = \begin{pmatrix} -3 & -1 \\ 0 & 2 \end{pmatrix} \qquad \hat{\mathbf{G}}_{\lambda} = \begin{pmatrix} \exp(-5)-1 \\ 0 \end{pmatrix} = \begin{pmatrix} -0.993 \\ 0 \end{pmatrix}$$

Example: Predator prey model

- The tangent vector is obtained by solving $(\star\star)$

$$\begin{pmatrix} -3 & -1 \\ 0 & 2 \end{pmatrix} \left.\frac{\partial \hat{\mathbf{y}}}{\partial \lambda}\right|_0 = \begin{pmatrix} -0.993 \\ 0 \end{pmatrix}$$

We get

$$\left.\frac{\partial \hat{\mathbf{y}}}{\partial \lambda}\right|_0 = \begin{pmatrix} -0.331 \\ 0 \end{pmatrix} \quad \text{and} \quad \hat{\mathbf{y}}_1^{(0)} = \widehat{\mathbf{y}_0} + \Delta\lambda \left.\frac{\partial \hat{\mathbf{y}}}{\partial \lambda}\right|_0 = \begin{pmatrix} 0.967 \\ 0 \end{pmatrix}$$

where we specified $\Delta\lambda = 0.1$.

- $\hat{\mathbf{y}}_1^{(0)}$ is our starting guess for the Newton iteration at λ_1.

Example: Predator prey model

- General Newton step at λ_1 specifies $(\star)$, i.e.

$$\widehat{\mathbf{G}}_{\hat{\mathbf{y}}}(\hat{\mathbf{y}}_1^{(i)}, \lambda_1)\, \Delta\hat{\mathbf{y}}_1^{(i)} = -\widehat{\mathbf{G}}(\hat{\mathbf{y}}_1^{(i)}, \lambda_1) \qquad (\star\star\star)$$

for $\Delta\hat{\mathbf{y}}_1^{(i)} = \hat{\mathbf{y}}_1^{(i+1)} - \hat{\mathbf{y}}_1^{(i)}$.

- Step 1 from $\hat{\mathbf{y}}_1^{(0)}$ to $\hat{\mathbf{y}}_1^{(1)}$ corresponds to solving $(\star\star\star)$ for $i = 0$, i.e.

$$\begin{pmatrix} -2.805 & -0.967 \\ 0.0 & 1.901 \end{pmatrix} \Delta\hat{\mathbf{y}}_1^{(0)} = -\begin{pmatrix} -0.003 \\ 0.0 \end{pmatrix}$$

$$\text{to obtain} \qquad \hat{\mathbf{y}}_1^{(1)} = -\begin{pmatrix} 0.966 \\ 0.0 \end{pmatrix}$$

Example: Predator prey model

- Step 2 from $\hat{\mathbf{y}}_1^{(1)}$ to $\hat{\mathbf{y}}_1^{(2)}$ corresponds to solving $(\star\star\star)$ with $i = 1$, i.e.

$$\begin{pmatrix} -2.799 & -0.966 \\ 0.0 & 1.897 \end{pmatrix} \Delta\hat{\mathbf{y}}_1^{(1)} = \begin{pmatrix} 0.0 \\ 0.0 \end{pmatrix}$$

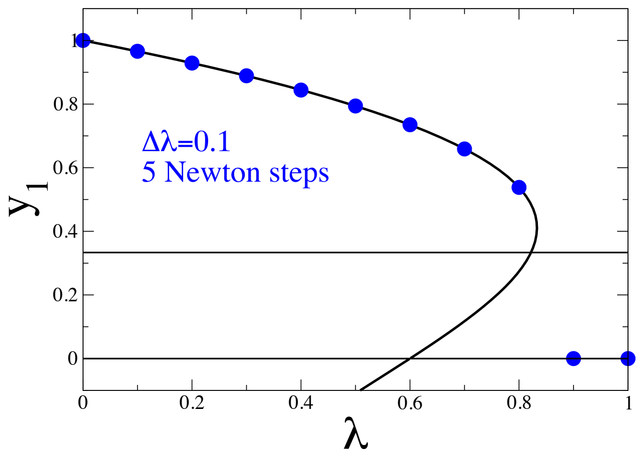
$$\text{to obtain} \qquad \hat{\mathbf{y}}_1^{(2)} = \begin{pmatrix} 0.966 \\ 0.0 \end{pmatrix}$$

- As $\hat{\mathbf{y}}_1^{(2)} = \hat{\mathbf{y}}_1^{(1)}$ to 3sd we have found

$$\hat{\mathbf{y}}_1 = \begin{pmatrix} 0.966 \\ 0.0 \end{pmatrix}$$

- Now, one would play this again: increase λ to $\lambda_2 = \lambda_1 + \Delta\lambda$, do the tangent prediction and Newton iterations, etc.

Parameter continuation - plot of results



Parameter continuation - issues with simple method

- Method uses predictor based on tangent $\partial\widehat{\mathbf{y}}/\partial\lambda$ where λ is the control parameter of the problem
- Newton's method used to correct solution at fixed λ
- This works nicely when locally there is one-to-one correspondence of solutions $\widehat{\mathbf{y}}$ and parameter λ
- Method breaks down at saddle-node bifurcations as there exist two solutions $\widehat{\mathbf{y}}$ for $\lambda < \lambda_{\rm sn}$ and none for $\lambda > \lambda_{\rm sn}$
- With other words: one can not go around folds (that occur frequently, see our example)
- How can we fix that?

Aims part 2

- Pseudo-arclength continuation
  - General scheme
  - Tangent predictor
  - Newton corrector
- Example problem (Predator-prey model)

Pseudo-arclength continuation

- We need parameter that is unique along a branch even if the branch undergoes saddle-node bifurcations
- Good option: Arclength s along the branch
- Then we treat λ as an additional element of the solution vector $\widehat{\mathbf{y}}$, i.e., we introduce $\widehat{\mathbf{x}} = (\widehat{\mathbf{y}}, \lambda)$ and determine both components in dependence of the new control parameter s
- However, s is not known beforehand
- Use local approximation (pythagoras)

$$|\Delta\widehat{\mathbf{y}}|^2 + (\Delta\lambda)^2 = (\Delta s)^2$$

to obtain additional equation $p(\widehat{\mathbf{y}}, \lambda, s) = 0$ that supplements $\widehat{\mathbf{G}}(\widehat{\mathbf{y}}, \lambda) = 0$

Transform arclength condition

- Start with

$$|\Delta\widehat{\mathbf{y}}|^2 + (\Delta\lambda)^2 = (\Delta s)^2$$

- Use

$$|\Delta\widehat{\mathbf{y}}|^2 = (\widehat{\mathbf{y}}_{j+1} - \widehat{\mathbf{y}}_j)\frac{(\widehat{\mathbf{y}}_{j+1} - \widehat{\mathbf{y}}_j)}{\Delta s}\Delta s \approx (\widehat{\mathbf{y}}_{j+1} - \widehat{\mathbf{y}}_j)\frac{\partial\widehat{\mathbf{y}}}{\partial s}\Delta s$$

and equally

$$(\Delta\lambda)^2 \approx (\lambda_{j+1} - \lambda_j)\frac{\partial\lambda}{\partial s}\Delta s$$

- Therefore, the additional equation is

$$p(\widehat{\mathbf{y}}, \lambda, s) = (\widehat{\mathbf{y}}_{j+1} - \widehat{\mathbf{y}}_j)\frac{\partial\widehat{\mathbf{y}}}{\partial s} + (\lambda_{j+1} - \lambda_j)\frac{\partial\lambda}{\partial s} - \Delta s = 0$$

Pseudo-arclength continuation – Notation

- Compact notation allows us to use the formalism introduced for simple continuation scheme
- Treat $\lambda(s)$ as additional dependent variable beside $\widehat{\mathbf{y}}(s)$
- Join them into vector $\widehat{\mathbf{x}} = (\widehat{\mathbf{y}}, \lambda)$
- Introduce extended system of equations

$$\widehat{\mathbf{E}}(\widehat{\mathbf{x}}, s) = \begin{pmatrix} \widehat{\mathbf{G}}(\widehat{\mathbf{y}}, \lambda) \\ p(\widehat{\mathbf{y}}, \lambda, s) \end{pmatrix} = \widehat{\mathbf{0}}$$

- with Jacobian

$$\widehat{\mathbf{E}}_{\widehat{\mathbf{x}}} = \left(\begin{array}{c|c} \widehat{\mathbf{G}}_{\widehat{\mathbf{y}}} & \widehat{\mathbf{G}}_{\lambda} \\ \hline p_{\widehat{\mathbf{y}}} & p_{\lambda} \end{array}\right) = \left(\begin{array}{c|c} \widehat{\mathbf{G}}_{\widehat{\mathbf{y}}} & \widehat{\mathbf{G}}_{\lambda} \\ \hline \frac{\partial \widehat{\mathbf{y}}}{\partial s} & \frac{\partial \lambda}{\partial s} \end{array}\right)$$

Pseudo-arclength continuation - any step

- For $j = 0$ take starting value $\widehat{\mathbf{x}}_0 = (\widehat{\mathbf{y}}_0, \lambda_0)$, otherwise take result of previous step $\widehat{\mathbf{x}}_j = (\widehat{\mathbf{y}}_j, \lambda_j)$ that solves $\widehat{\mathbf{E}}(\widehat{\mathbf{x}}_j, s_j) = 0$. ($s_j$ may be shifted to zero each step, but may also be monitored)
- Want the solution $(\widehat{\mathbf{x}}_{j+1}, s_{j+1})$ at $s_{j+1} = s_j + \Delta s$
- Strategy
  - (1) Use tangent of curve $\widehat{\mathbf{x}}(s)$ at point $(\widehat{\mathbf{x}}_j, s_j)$ as predictor $\widehat{\mathbf{x}}_{j+1}^{(0)}$ for $\widehat{\mathbf{x}}_{j+1}$ at $s_{j+1} = s_j + \Delta s$.
  - (2) Use Newton's method to iterate (starting with $\widehat{\mathbf{x}}_{j+1}^{(0)}$) and obtain $\widehat{\mathbf{x}}_{j+1}$ at fixed s_{j+1} to arbitrary exactness.
- Then repeat to change s to continue along solution branch

Pseudo-arclength continuation - obtaining the tangent

- How do we get the tangent direction $\partial \widehat{\mathbf{x}}/\partial s$?
- Differentiate $\widehat{\mathbf{E}}(\widehat{\mathbf{x}}(s), s) = 0$ with respect to s (chain rule):

$$\frac{\partial \widehat{\mathbf{E}}}{\partial \widehat{\mathbf{x}}} \frac{\partial \widehat{\mathbf{x}}}{\partial s} + \frac{\partial \widehat{\mathbf{E}}}{\partial s} = 0 \qquad (\star\star)$$

- Solve inhomogeneous algebraic system of equations $(\star\star)$ for tangent vector $\partial \widehat{\mathbf{x}}/\partial s$.
- Notation:

$$\widehat{\mathbf{E}}_{\widehat{\mathbf{x}}} = \frac{\partial \widehat{\mathbf{E}}}{\partial \widehat{\mathbf{x}}} \quad \text{and} \quad \widehat{\mathbf{E}}_s = \frac{\partial \widehat{\mathbf{E}}}{\partial s}$$

- Problem: $(\star\star)$ is nonlinear in $\partial \widehat{\mathbf{x}}/\partial s$ as it is contained in $\widehat{\mathbf{E}}_{\widehat{\mathbf{x}}}$

Pseudo-arclength continuation - obtaining the tangent

- Solve $(\star\star)$ iteratively, i.e.,solve for $k = 1, 2, \ldots$

$$\left(\begin{array}{c|c} \widehat{\mathbf{G}}_{\widehat{\mathbf{y}}} & \widehat{\mathbf{G}}_{\lambda} \\ \hline \frac{\partial \widehat{\mathbf{y}}}{\partial s}^{(k)} & \frac{\partial \lambda}{\partial s}^{(k)} \end{array}\right) \left(\begin{array}{c} \frac{\partial \widehat{\mathbf{y}}}{\partial s}^{(k+1)} \\ \hline \frac{\partial \lambda}{\partial s}^{(k+1)} \end{array}\right) + \left(\begin{array}{c} \frac{\partial \widehat{\mathbf{G}}}{\partial s} \\ \hline \frac{\partial p}{\partial s} \end{array}\right) = \widehat{\mathbf{0}}$$

- Above we use $\frac{\partial \widehat{\mathbf{G}}}{\partial s} = \widehat{\mathbf{0}}$ and $\frac{\partial p}{\partial s} = -1$.
- For starting values $\frac{\partial \widehat{\mathbf{x}}}{\partial s}^{(0)}$ of iteration use arbitrary choice in first continuation step ($j = 0$), and the values from previous continuation step otherwise
- As the final equation of the system above is merely a normalisation condition, one should iterate only once and normalise[1]

[1] Otherwise result might oscillate between two vectors in tangent direction with absolute values ξ and $1/\xi$ for arbitrary ξ.

Pseudo-arclength continuation - tangent predictor step

1. Start with $\widehat{\mathbf{x}}_j$ at s_j.
2. We obtained tangent direction $\frac{\partial \widehat{\mathbf{x}}}{\partial s}|_j$ at $(\widehat{\mathbf{x}}_j, s_j)$
3. Obtain initial 'guess' $\widehat{\mathbf{x}}_{j+1}^{(0)}$ at $s_{j+1} = s_j + \Delta s$ by

$$\widehat{\mathbf{x}}_{j+1}^{(0)} = \widehat{\mathbf{x}}_j + \Delta s \left. \frac{\partial \widehat{\mathbf{x}}}{\partial s} \right|_j \Bigg/ \left| \left. \frac{\partial \widehat{\mathbf{x}}}{\partial s} \right|_j \right|$$

using the normalised tangent direction

Pseudo-arclength continuation - Newton correction

1. Take initial guess $\widehat{\mathbf{x}}_{j+1}^{(0)}$ at s_{j+1}.
2. Obtain next iteration by solving inhomogeneous linear algebraic system of equations $(\star)$ at fixed s_{j+1}, i.e.

$$\widehat{\mathbf{E}}_{\widehat{\mathbf{x}}}(\widehat{\mathbf{x}}_{j+1}^{(i)}, s_{j+1})\, \Delta\widehat{\mathbf{x}}_{j+1}^{(i)} = -\widehat{\mathbf{E}}(\widehat{\mathbf{x}}_{j+1}^{(i)}, s_{j+1})$$

for $\Delta\widehat{\mathbf{x}}_{j+1}^{(i)} = \widehat{\mathbf{x}}_{j+1}^{(i+1)} - \widehat{\mathbf{x}}_{j+1}^{(i)}$.

3. Obtain $\widehat{\mathbf{x}}_{j+1}^{(i+1)} = \widehat{\mathbf{x}}_{j+1}^{(i)} + \Delta\widehat{\mathbf{x}}_{j+1}^{(i)}$.

Repeat Newton step (1.-3.) until wanted accuracy is reached (i.e. $||\Delta\widehat{\mathbf{x}}_{j+1}^{(i)}||$ smaller than some given threshhold value).

Example: Predator prey model

- Remember, a typical two-species model is

$$\frac{d\widehat{\mathbf{y}}}{dt} = \widehat{\mathbf{G}}(\widehat{\mathbf{y}}, \lambda)$$

with

$$\widehat{\mathbf{G}}(\widehat{\mathbf{y}}, \lambda) = \begin{pmatrix} 3y_1(1-y_1) - y_1 y_2 - \lambda(1-\exp(-5y_1)) \\ -y_2 + 3y_1 y_2 \end{pmatrix}$$

Example: Predator prey model

- Fixed points / equilibria given by

$$\widehat{\mathbf{G}}(\widehat{\mathbf{y}}, \lambda) = \widehat{\mathbf{0}}$$

- Introduce arclength s as control parameter and get augmented system

$$\widehat{\mathbf{E}}(\widehat{\mathbf{x}}, s) = \begin{pmatrix} \widehat{\mathbf{G}}(\widehat{\mathbf{x}}) \\ \hline p(\widehat{\mathbf{x}}, s) \end{pmatrix} = \widehat{\mathbf{0}}$$

with $\widehat{\mathbf{x}} = (\widehat{\mathbf{y}}, \lambda)$

Example: Predator prey model

- We need the Jacobian

$$\widehat{\mathbf{E}}_{\widehat{\mathbf{x}}} = \begin{pmatrix} 3-6y_1-y_2-5\lambda\exp(-5y_1) & -y_1 & \exp(-5y_1)-1 \\ 3y_2 & 3y_1-1 & 0 \\ \frac{\partial y_1}{\partial s} & \frac{\partial y_2}{\partial s} & \frac{\partial \lambda}{\partial s} \end{pmatrix}$$

and the derivative of $\widehat{\mathbf{E}}$ with respect to the continuation parameter s

$$\widehat{\mathbf{E}}_s = \frac{\partial \widehat{\mathbf{E}}}{\partial s} = \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix}$$

- At $\widehat{\mathbf{y}}_0 = (1,0)$ and $\lambda_0 = 0$ we have

$$\widehat{\mathbf{G}}_{\widehat{\mathbf{y}}} = \begin{pmatrix} -3 & -1 \\ 0 & 2 \end{pmatrix} \qquad \widehat{\mathbf{G}}_\lambda = \begin{pmatrix} \exp(-5)-1 \\ 0 \end{pmatrix} = \begin{pmatrix} -0.993 \\ 0 \end{pmatrix}$$

Example: Predator prey model

- The tangent vector $\widehat{\mathbf{x}}_s$ is obtained by solving $(\star\star)$ iteratively:

$$\begin{pmatrix} -3 & -1 & -0.993 \\ 0 & 2 & 0 \\ 1/\sqrt{3} & 1/\sqrt{3} & 1/\sqrt{3} \end{pmatrix} \left.\frac{\partial \widehat{\mathbf{x}}}{\partial s}\right|_0 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}$$

where the last row in $\widehat{\mathbf{E}}_{\widehat{\mathbf{x}}}$ is a random initial choice (result converges after 1 iteration)
After normalisation, we get

$$\left.\frac{\partial \widehat{\mathbf{x}}}{\partial s}\right|_0 = \begin{pmatrix} -0.314 \\ 0.0 \\ 0.949 \end{pmatrix} \quad \text{and} \quad \widehat{\mathbf{x}}_1^{(0)} = \widehat{\mathbf{x}_0} + \Delta s \left.\frac{\partial \widehat{\mathbf{y}}}{\partial \lambda}\right|_0 = \begin{pmatrix} 0.969 \\ 0 \\ 0.095 \end{pmatrix}$$

where we specified $\Delta s = 0.1$.

- $\widehat{\mathbf{x}}_1^{(0)}$ is our starting guess for the Newton iteration at s_1.

Example: Predator prey model

- General Newton step at s_1 is

$$\widehat{\mathbf{E}}_{\widehat{\mathbf{x}}}(\widehat{\mathbf{x}}_1^{(i)}, s_1)\,\Delta\widehat{\mathbf{x}}_1^{(i)} = -\widehat{\mathbf{E}}(\widehat{\mathbf{x}}_1^{(i)}, s_1) \qquad (\star\star\star)$$

for $\Delta\widehat{\mathbf{x}}_1^{(i)} = \widehat{\mathbf{x}}_1^{(i+1)} - \widehat{\mathbf{x}}_1^{(i)}$

- Step 1 from $\widehat{\mathbf{x}}_1^{(0)}$ to $\widehat{\mathbf{x}}_1^{(1)}$ corresponds to solving $(\star\star\star)$ for $i = 0$, i.e.

$$\begin{pmatrix} -2.815 & -0.969 & -0.992 \\ 0.0 & 1.906 & 0.0 \\ -0.314 & 0.0 & 0.949 \end{pmatrix} \Delta\widehat{\mathbf{x}}_1^{(0)} = -\begin{pmatrix} -0.0032 \\ 0.0 \\ 0.0 \end{pmatrix}$$

$$\text{to obtain} \qquad \widehat{\mathbf{x}}_1^{(1)} = \begin{pmatrix} 0.968 \\ 0.0 \\ 0.095 \end{pmatrix}$$

Example: Predator prey model

- Step 2 from $\widehat{\mathbf{x}}_1^{(1)}$ to $\widehat{\mathbf{x}}_1^{(2)}$ corresponds to solving $(\star\star\star)$ with $i = 1$, i.e.

$$\begin{pmatrix} -2.809 & -0.968 & -0.992 \\ 0.0 & 1.903 & 0.0 \\ -0.314 & 0.0 & 0.949 \end{pmatrix} \Delta\widehat{\mathbf{x}}_1^{(1)} = -\begin{pmatrix} -0.0 \\ 0.0 \\ 0.0 \end{pmatrix}$$

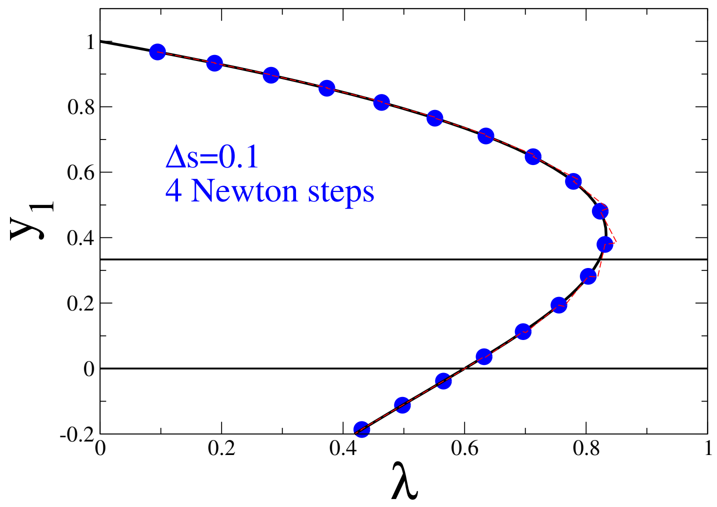
$$\text{to obtain} \qquad \widehat{\mathbf{x}}_1^{(2)} = \begin{pmatrix} 0.968 \\ 0.0 \\ 0.095 \end{pmatrix}$$

- As $\widehat{\mathbf{x}}_1^{(2)} = \widehat{\mathbf{x}}_1^{(1)}$ to 3sd we have found

$$\widehat{\mathbf{x}}_1 = \begin{pmatrix} 0.968 \\ 0.0 \\ 0.095 \end{pmatrix}$$

- Now one would play this again: increase s to $s_2 = s_1 + \Delta s$, do the tangent prediction and correction by Newton iterations, etc.

Parameter continuation - plot of results



Parameter continuation - alternatives

- Predictor step
  - use secant instead of tangent - needs information about $\widehat{\mathbf{x}}_j$ and $\widehat{\mathbf{x}}_{j-1}$ (need tangent predictor for 1st step)
- Corrector step [various choices of $p(\widehat{\mathbf{x}}, s)$]
  - Natural continuation: fix any component of $\widehat{\mathbf{x}}_{j+1}$ (special case simple continuation: fix λ_{j+1}) - best choice component with largest $|\partial_s x|$ as this is the fastest changing one
  - Pseudo-arclength: Newton steps orthogonal to tangent direction
  - Moore-Penrose continuation: $p(\widehat{\mathbf{x}}, s)$ changes during Newton iterations

Outlook:
Detection and continuation of bifurcations

General strategy:

- Device a test function $\tau(\widehat{\mathbf{x}}, \mu)$ that crosses zero at bifurcation in question (e.g., based on Jacobian). μ is an additional control parameter beside primary control parameter λ
- When following $\widehat{\mathbf{x}}$ at fixed μ, monitor $\tau(\widehat{\mathbf{x}}, \mu)$; when zero-crossing is detected, obtain exact λ_{bif} through Newton on augmented $\widehat{\mathbf{E}}_{\mathrm{aug}} = (\quad \widehat{\mathbf{E}} \quad \mid \tau)^T = 0$
- Continue loci of bifurcation by continuing (in s) solutions to $\widehat{\mathbf{E}}_{\mathrm{aug}}(\widehat{\mathbf{x}}_{\mathrm{aug}}) = \widehat{\mathbf{0}}$ with $\widehat{\mathbf{x}}_{\mathrm{aug}} = (\ \widehat{\mathbf{y}} \mid \lambda \ \ \mu)^T$

Software for continuation techniques

General strategy:

- `auto07p`: very versatile, 'any' ODE problem (github.com/auto-07p), coupling with FFTW available Tutorials available on CeNoS website (www.uni-muenster.de/CeNoS/Lehre/Tutorials/continuation.html)
- `pde2path`: continuation for systems of PDEs (www.staff.uni-oldenburg.de/hannes.uecker/pde2path)
- `oomph-lib`: released version can do time simulation & continuation for PDE (oomph-lib.maths.man.ac.uk)
- Others: `matcont`, `DDE-BIFTOOL`, `PyDSTool`, `loca`, ...

Further themes

- Coarse bifurcation theory: use continuation tool as wrapper on 'any' time simulator, be it continuous/discrete/black box (Kevrekidis, Avitabile, Lloyd)
- Continuation of stable/unstable manifolds in phase space (Doedel)
- Continuation of homoclinic/heteroclinic solutions to ODE (`homcont` part of `auto07p`)
- Tricks to (i) follow global bifurcations, (ii) obtain and follow self-similar solutions, (iii) obtain real eigenvalues as branching points, . . .