Lecture 1: Introduction to bifurcation analysis

Patrick E. Farrell

University of Oxford

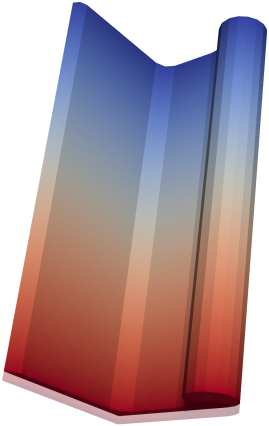
May 29

Can you conduct an experiment twice ...

...and get two different answers?

Can you conduct an experiment twice . . .

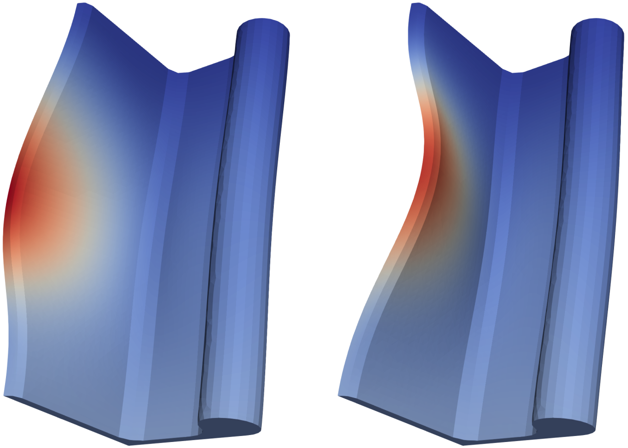
. . . and get two different answers?



Axial displacement test of an Embraer aircraft stiffener.

Can you conduct an experiment twice . . .

. . . and get two different answers?

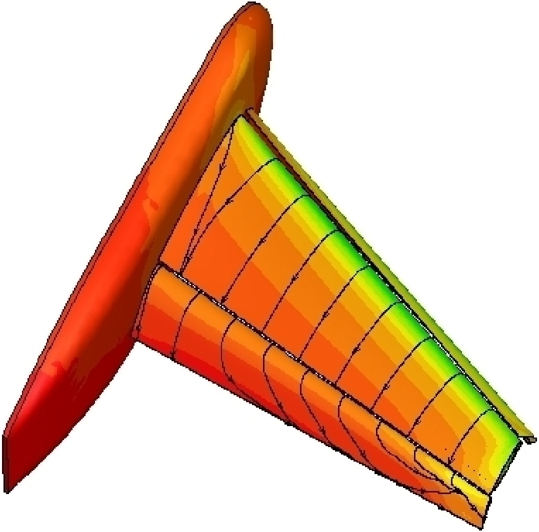


Two different, stable configurations.

When a problem has multiple solutions, it is usually crucial.

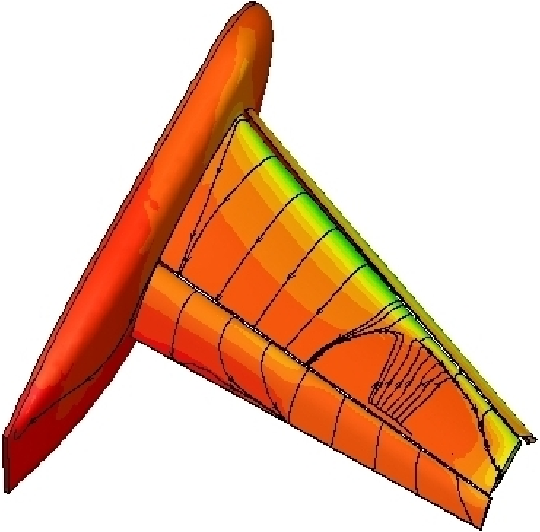
The AIAA/NASA high lift prediction test case (Kamenetskiy et al., 2013)

When a problem has multiple solutions, it is usually crucial.



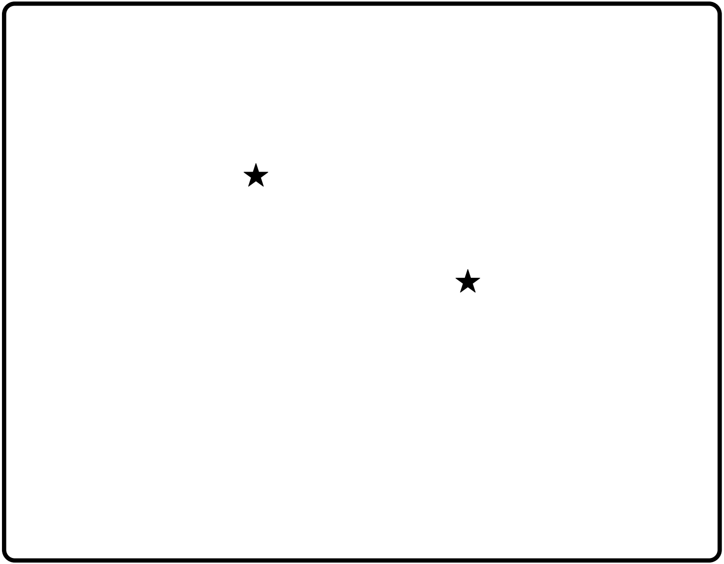
The AIAA/NASA high lift prediction test case (Kamenetskiy et al., 2013)

When a problem has multiple solutions, it is usually crucial.



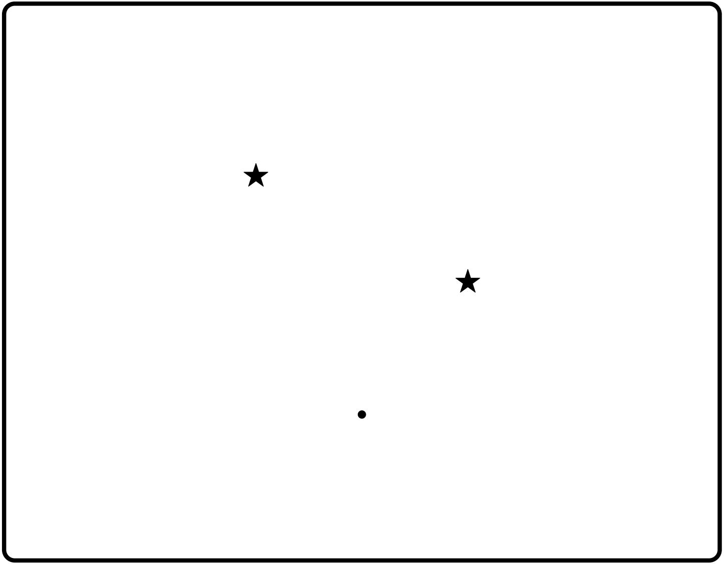
The AIAA/NASA high lift prediction test case (Kamenetskiy et al., 2013)

When a problem has multiple solutions, it is usually crucial.



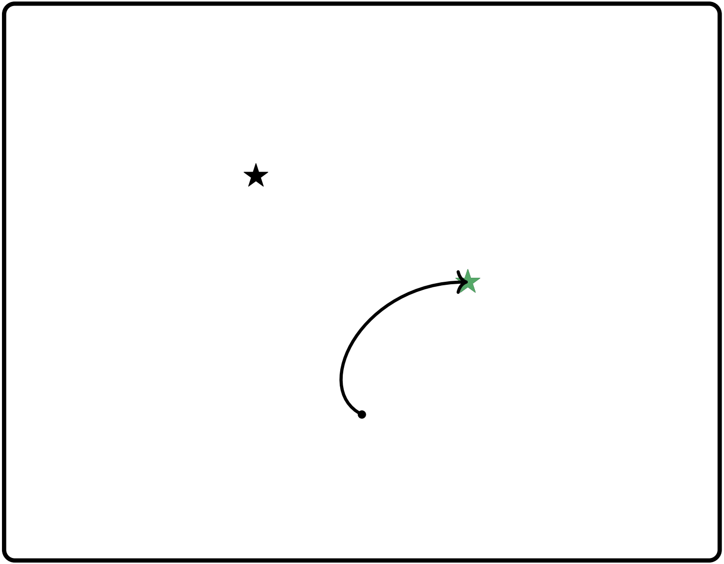
A PDE with two unknown solutions

When a problem has multiple solutions, it is usually crucial.



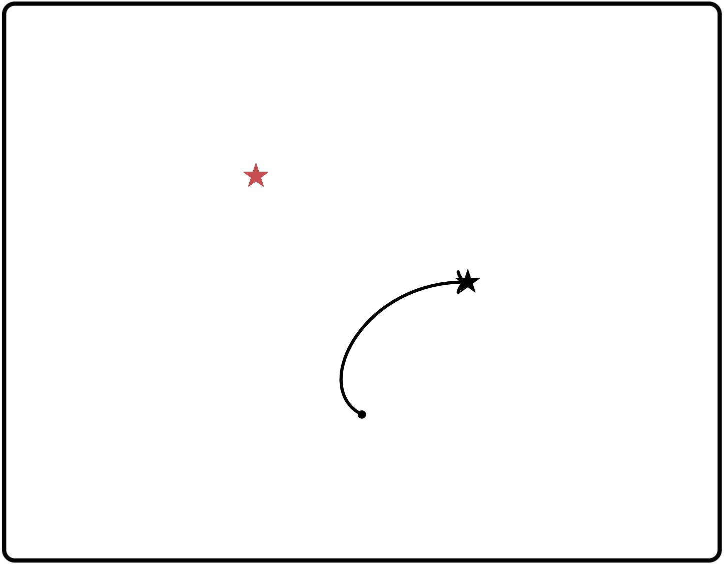
Start from some initial guess

When a problem has multiple solutions, it is usually crucial.



We converge to one solution, our prediction

When a problem has multiple solutions, it is usually crucial.



But nature has chosen another (unknown) solution!

When a problem has multiple solutions, it is usually crucial.

We have encountered unexpected multiple solutions in both simple and complex configurations in computational fluid dynamics (CFD); this phenomenon is both extremely important and not well understood. It has **serious implications for the use of CFD as a predictive tool**.

— Venkat Venkatakrishnan
Computational Aerodynamic Optimization
Boeing Research & Technology

Section 2

Scope

Mathematical formulation

Compute the multiple *solutions* $u^\star$ of a stationary nonlinear equation

$$F(u^\star, \lambda) = 0$$
$$F \in C^1(X \times \mathbb{R}, Y)$$

as a function of a parameter $\lambda \in \mathbb{R}$.

Mathematical formulation

Compute the multiple *solutions* $u^\star$ of a stationary nonlinear equation

$$F(u^\star, \lambda) = 0$$
$$F \in C^1(X \times \mathbb{R}, Y)$$

as a function of a parameter $\lambda \in \mathbb{R}$.

Case #1: aircraft stiffener

$u^\star$ displacement, λ loading, F hyperelasticity

Case #2: aircraft wing

$u^\star$ velocity and pressure, λ angle of attack, F Navier–Stokes

Mathematical formulation

Compute the multiple *solutions* $u^\star$ of a stationary nonlinear equation

$$F(u^\star, \lambda) = 0$$
$$F \in C^1(X \times \mathbb{R}, Y)$$

as a function of a parameter $\lambda \in \mathbb{R}$.

$F \in C^1$ means that it is Fréchet-differentiable: there exists a function

$$F_u : X \times \mathbb{R} \to L(X, Y)$$

Mathematical formulation

Compute the multiple *solutions* $u^\star$ of a stationary nonlinear equation

$$F(u^\star, \lambda) = 0$$
$$F \in C^1(X \times \mathbb{R}, Y)$$

as a function of a parameter $\lambda \in \mathbb{R}$.

$F \in C^1$ means that it is Fréchet-differentiable: there exists a function

$$F_u : X \times \mathbb{R} \to L(X, Y)$$

with the approximation property

$$\lim_{v \to 0} \frac{\|F(u + v, \lambda) - F(u, \lambda) - F_u(u, \lambda)v\|}{\|v\|} = 0 \text{ for all } v \in X.$$

Mathematical formulation

Compute the multiple *solutions* $u^\star$ of a stationary nonlinear equation

$$F(u^\star, \lambda) = 0$$
$$F \in C^1(X \times \mathbb{R}, Y)$$

as a function of a parameter $\lambda \in \mathbb{R}$.

$F \in C^1$ means that it is Fréchet-differentiable: there exists a function

$$F_u : X \times \mathbb{R} \to L(X, Y)$$

with the approximation property

$$\lim_{v \to 0} \frac{\|F(u + v, \lambda) - F(u, \lambda) - F_u(u, \lambda)v\|}{\|v\|} = 0 \text{ for all } v \in X.$$

Moreover F_u is continuous. The same holds for F_λ.

Mathematical formulation

Compute the multiple *solutions* $u^\star$ of a stationary nonlinear equation

$$F(u^\star, \lambda) = 0$$
$$F \in C^1(X \times \mathbb{R}, Y)$$

as a function of a parameter $\lambda \in \mathbb{R}$.

Warning

We (usually) can't guarantee to find *all* solutions. But finding many is better than finding one.

Lecture 1

Introduction to bifurcation theory; great theorems of nonlinear functional analysis.

Lecture 1

Introduction to bifurcation theory; great theorems of nonlinear functional analysis.

Lecture 2

Classical numerical algorithms for computing bifurcation diagrams. Branch continuation, bifurcation detection and localisation, branch switching.

Lecture 1

Introduction to bifurcation theory; great theorems of nonlinear functional analysis.

Lecture 2

Classical numerical algorithms for computing bifurcation diagrams. Branch continuation, bifurcation detection and localisation, branch switching.

Lecture 3

Deflation techniques for computing *disconnected* bifurcation diagrams.

What is *not* in scope:

- time-dependent problems (dynamical systems);

What is *not* in scope:

- time-dependent problems (dynamical systems);
- bifurcations in maps (discrete systems);

What is *not* in scope:

- time-dependent problems (dynamical systems);
- bifurcations in maps (discrete systems);
- bifurcations of high codimension (multiple parameters);

What is *not* in scope:

- time-dependent problems (dynamical systems);
- bifurcations in maps (discrete systems);
- bifurcations of high codimension (multiple parameters);
- algorithms that only work for ODEs/coarse discretisations.

What is *not* in scope:

- time-dependent problems (dynamical systems);
- bifurcations in maps (discrete systems);
- bifurcations of high codimension (multiple parameters);
- algorithms that only work for ODEs/coarse discretisations.
- the relationship between symmetries and bifurcations.

What is *not* in scope:

- time-dependent problems (dynamical systems);
- bifurcations in maps (discrete systems);
- bifurcations of high codimension (multiple parameters);
- algorithms that only work for ODEs/coarse discretisations.
- the relationship between symmetries and bifurcations.

Goal for the course

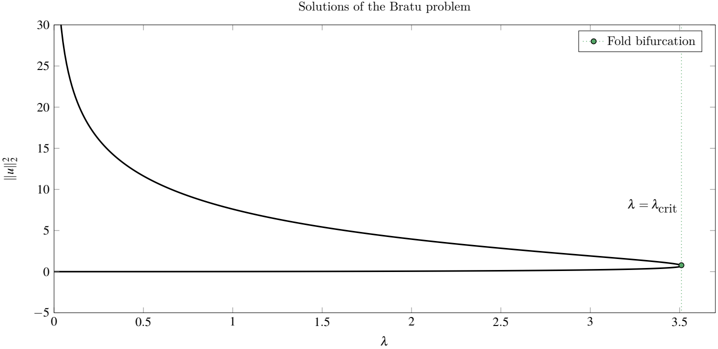
Develop practical numerical methods for computing multiple solutions of fine discretisations of nonlinear BVPs.

Example: Liouville–Bratu–Gelfand problem

$$u'' + \lambda e^u = 0, \quad u(0) = 0 = u(1).$$

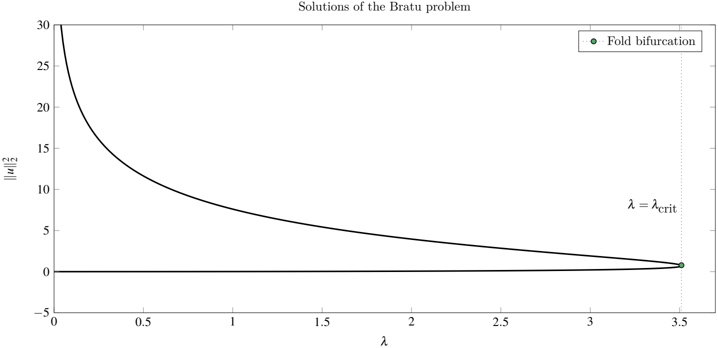
Example: Liouville–Bratu–Gelfand problem

$$u'' + \lambda e^u = 0, \quad u(0) = 0 = u(1).$$



Example: Liouville–Bratu–Gelfand problem

$$u'' + \lambda e^u = 0, \quad u(0) = 0 = u(1).$$

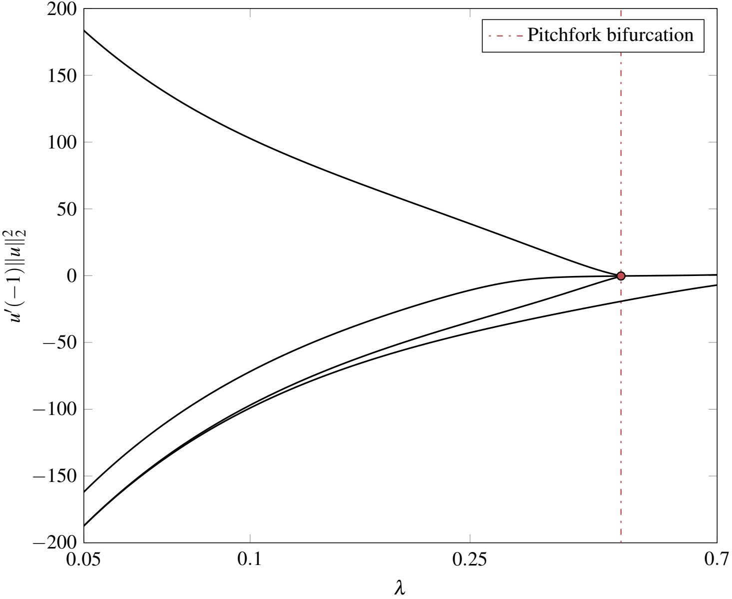


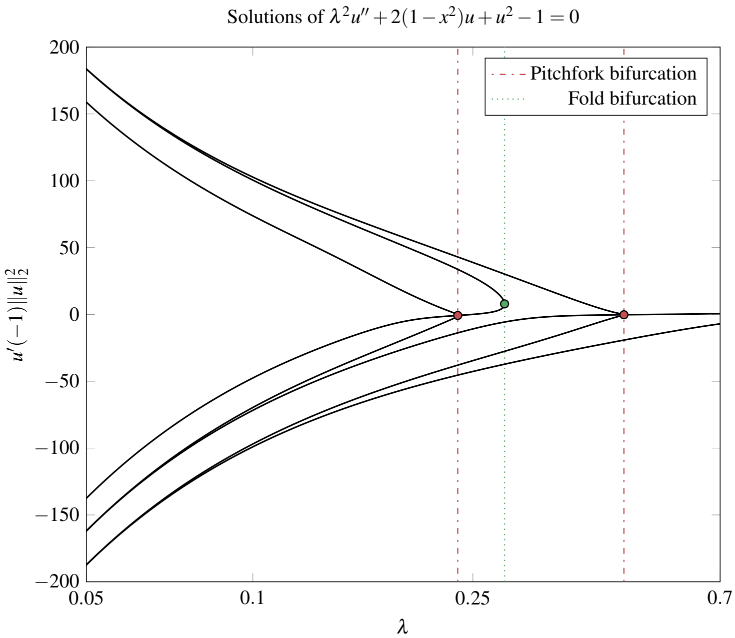
$$\#\text{ solutions} = \begin{cases} 1 & \lambda \in \{0, \lambda_{\text{crit}}\}, \\ 2 & \lambda \in (0, \lambda_{\text{crit}}), \\ 0 & \text{otherwise.} \end{cases}$$

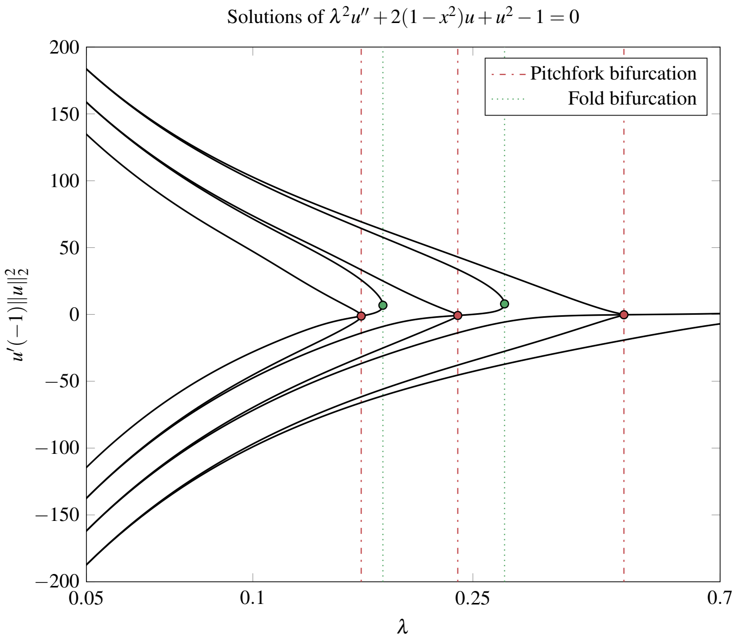
Example: Carrier's problem

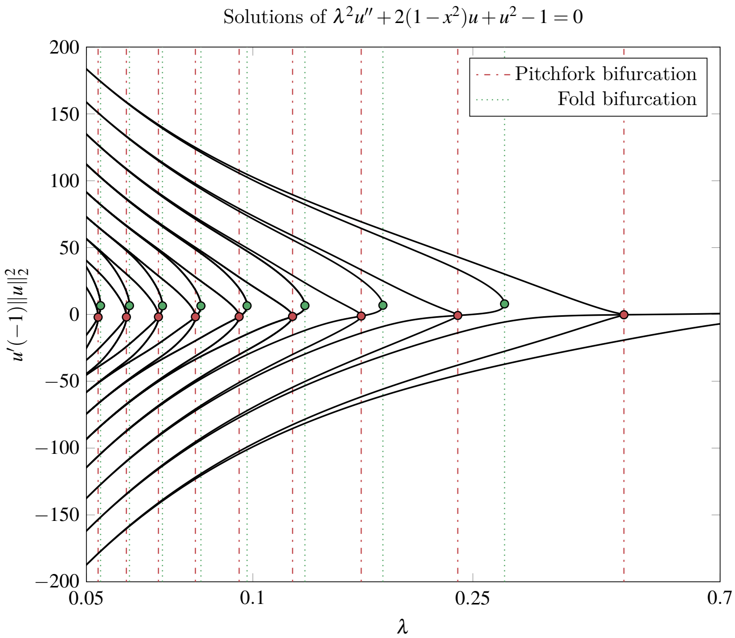
$$\lambda^2 u'' + 2(1 - x^2)u + u^2 - 1 = 0, \quad u(-1) = 0 = u(1).$$

Solutions of $\lambda^2 u'' + 2(1-x^2)u + u^2 - 1 = 0$




Solutions of $\lambda^2 u'' + 2(1 - x^2)u + u^2 - 1 = 0$
Pitchfork bifurcation
Fold bifurcation
$u'(-1)\|u\|_2^2$
λ
200
150
100
50
0
−50
−100
−150
−200
0.05
0.1
0.25
0.7


Solutions of $\lambda^2 u'' + 2(1-x^2)u + u^2 - 1 = 0$
Pitchfork bifurcation
Fold bifurcation
200
150
100
50
0
−50
−100
−150
−200
0.05
0.1
0.25
0.7
$u'(-1)\|u\|_2^2$
λ


Solutions of $\lambda^2 u'' + 2(1-x^2)u + u^2 - 1 = 0$
Pitchfork bifurcation
Fold bifurcation
200
150
100
50
0
−50
−100
−150
−200
$u'(-1)\|u\|_2^2$
0.05
0.1
0.25
0.7
λ

Section 3

Great Theorems of Nonlinear Functional Analysis

We now review some theory about the solution of nonlinear PDE:

We now review some theory about the solution of nonlinear PDE:

- the Newton–Kantorovich theorem;

We now review some theory about the solution of nonlinear PDE:

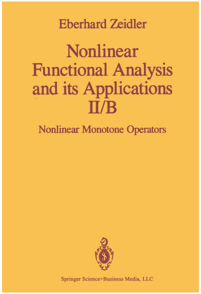
- the Newton–Kantorovich theorem;
- the Rall–Rheinboldt theorem;

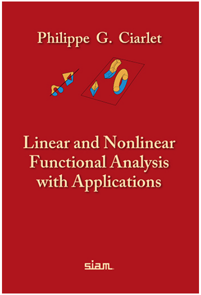
We now review some theory about the solution of nonlinear PDE:

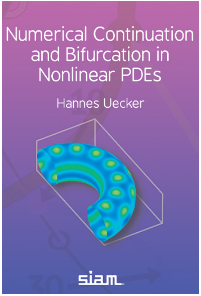
- the Newton–Kantorovich theorem;
- the Rall–Rheinboldt theorem;
- the implicit function theorem.

We now review some theory about the solution of nonlinear PDE:

- the Newton–Kantorovich theorem;
- the Rall–Rheinboldt theorem;
- the implicit function theorem.







Primary references.

Subsection 1

Newton–Kantorovich

The *Newton–Kantorovich* algorithm is an algorithm for solving nonlinear equations on the *infinite-dimensional* level (for a fixed parameter value).

The *Newton–Kantorovich* algorithm is an algorithm for solving nonlinear equations on the *infinite-dimensional* level (for a fixed parameter value).

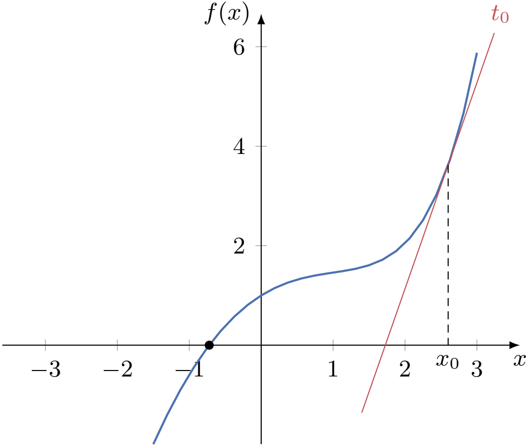
This will target the solution of the nonlinear problem by solving a sequence of linear problems. Each of these linear problems can then be discretised (e.g. with a finite element method).

The *Newton–Kantorovich* algorithm is an algorithm for solving nonlinear equations on the *infinite-dimensional* level (for a fixed parameter value).

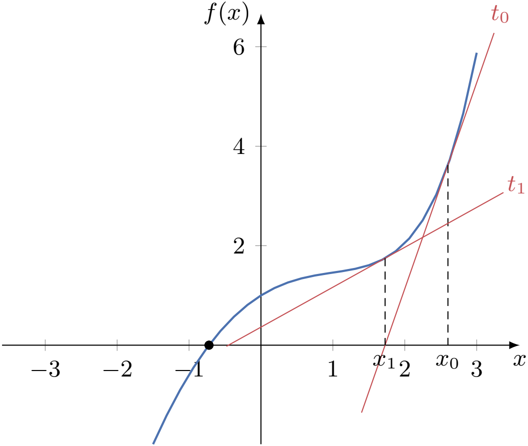
This will target the solution of the nonlinear problem by solving a sequence of linear problems. Each of these linear problems can then be discretised (e.g. with a finite element method).

First, let's recall Newton's method in $\mathbb{R}$ and $\mathbb{R}^N$.

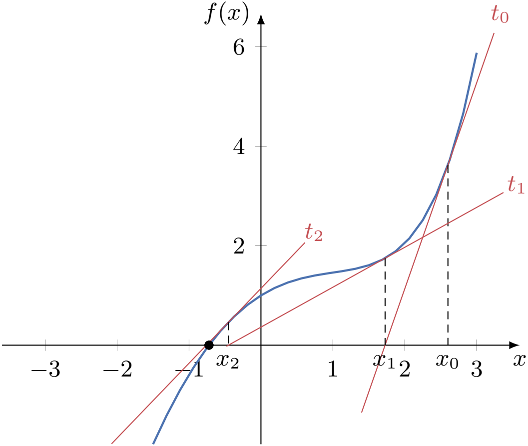
Core idea: solve succession of linearised rootfinding problems.



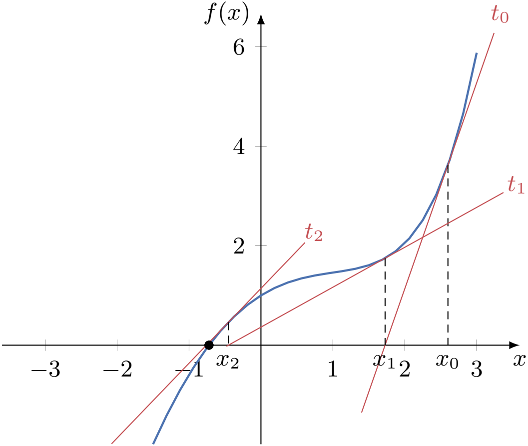
Core idea: solve succession of linearised rootfinding problems.



Core idea: solve succession of linearised rootfinding problems.



Core idea: solve succession of linearised rootfinding problems.



solve $f'(x_k)\delta x_k = -f(x_k)$; update $x_{k+1} = x_k + \delta x_k$.

Termination

The algorithm terminates if $f(x_k) = 0$, as desired.

Termination

The algorithm terminates if $f(x_k) = 0$, as desired.

Invertibility

We require $f'(x_k)$ to be nonzero at every iteration.

Termination

The algorithm terminates if $f(x_k) = 0$, as desired.

Invertibility

We require $f'(x_k)$ to be nonzero at every iteration.

Poor global convergence

The initial guess matters. With poor initial guesses, Newton's method may diverge to infinity, or get stuck in a cycle.

Termination

The algorithm terminates if $f(x_k) = 0$, as desired.

Invertibility

We require $f'(x_k)$ to be nonzero at every iteration.

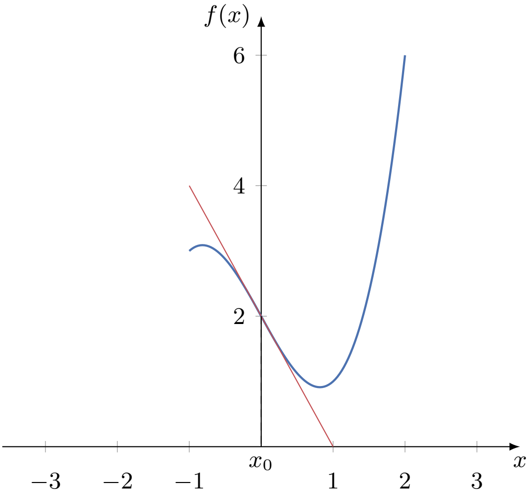
Poor global convergence

The initial guess matters. With poor initial guesses, Newton's method may diverge to infinity, or get stuck in a cycle.

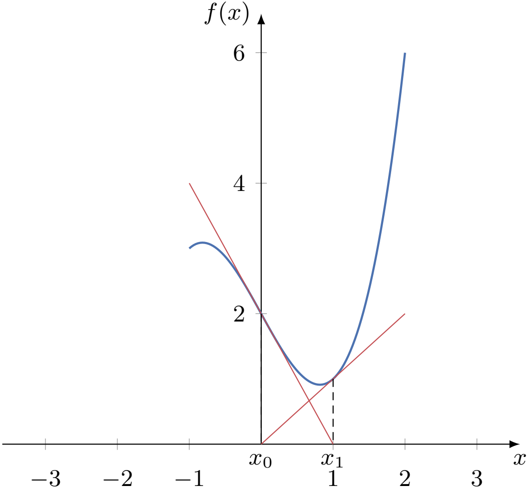
Good local convergence

If f is smooth, the solution is isolated, and the guess close, Newton converges quadratically.

Far from a solution, Newton's method can get stuck in a loop.



Far from a solution, Newton's method can get stuck in a loop.



This geometric reasoning is hard to generalise to higher dimensions. Let's look at a derivation that *does* extend.

This geometric reasoning is hard to generalise to higher dimensions. Let's look at a derivation that *does* extend.

Consider the Taylor expansion of f around x_k:

$$f(x_k + \delta x_k) = f(x_k) + f'(x_k)\delta x_k + \mathcal{O}(\delta x_k^2).$$

This geometric reasoning is hard to generalise to higher dimensions. Let's look at a derivation that *does* extend.

Consider the Taylor expansion of f around x_k:

$$f(x_k + \delta x_k) = f(x_k) + f'(x_k)\delta x_k + \mathcal{O}(\delta x_k^2).$$

Linearise the model by ignoring higher-order terms:

$$f(x_k + \delta x) \approx f(x_k) + f'(x_k)\delta x_k$$

This geometric reasoning is hard to generalise to higher dimensions. Let's look at a derivation that *does* extend.

Consider the Taylor expansion of f around x_k:

$$f(x_k + \delta x_k) = f(x_k) + f'(x_k)\delta x_k + \mathcal{O}(\delta x_k^2).$$

Linearise the model by ignoring higher-order terms:

$$f(x_k + \delta x) \approx f(x_k) + f'(x_k)\delta x_k$$

and find δx such that $f(x_k + \delta x) \approx 0$:

$$0 = f(x_k) + f'(x_k)\delta x_k.$$

This geometric reasoning is hard to generalise to higher dimensions. Let's look at a derivation that *does* extend.

Consider the Taylor expansion of f around x_k:

$$f(x_k + \delta x_k) = f(x_k) + f'(x_k)\delta x_k + \mathcal{O}(\delta x_k^2).$$

Linearise the model by ignoring higher-order terms:

$$f(x_k + \delta x) \approx f(x_k) + f'(x_k)\delta x_k$$

and find δx such that $f(x_k + \delta x) \approx 0$:

$$0 = f(x_k) + f'(x_k)\delta x_k.$$

This yields

$$\delta x_k = [f'(x_k)]^{-1} f(x_k).$$

This naturally extends to $F \in C^1(\mathbb{R}^N; \mathbb{R}^N)$. Newton's method is to

$$\text{solve } F_x(x_k)\delta x_k = -F(x_k); \text{ update } x_{k+1} = x_k + \delta x_k,$$

where F_x is the Jacobian of F.

This naturally extends to $F \in C^1(\mathbb{R}^N; \mathbb{R}^N)$. Newton's method is to

$$\text{solve } F_x(x_k)\delta x_k = -F(x_k); \text{ update } x_{k+1} = x_k + \delta x_k,$$

where F_x is the Jacobian of F.

For the iteration to be well-defined, we need $F_x(x_k)$ to be invertible.

Newton's method has an important property that becomes apparent in higher dimensions.

Newton's method has an important property that becomes apparent in higher dimensions.

Given $F : \mathbb{R}^N \to \mathbb{R}^N$, and $x_0 \in \mathbb{R}^N$, we construct the sequence $x_0, x_1, \ldots$.

Newton's method has an important property that becomes apparent in higher dimensions.

Given $F : \mathbb{R}^N \to \mathbb{R}^N$, and $x_0 \in \mathbb{R}^N$, we construct the sequence $x_0, x_1, \dots$.

Now imagine that we change units or coordinate systems for our outputs F. Instead of solving $F(x) = 0$, we want to solve $\tilde{F}(x) = AF(x) = 0$, where $A \in \mathbb{R}^{N\times N}$ is constant and nonsingular. Of course, this doesn't change the roots $x^\star$.

Newton's method has an important property that becomes apparent in higher dimensions.

Given $F : \mathbb{R}^N \to \mathbb{R}^N$, and $x_0 \in \mathbb{R}^N$, we construct the sequence $x_0, x_1, \ldots$.

Now imagine that we change units or coordinate systems for our outputs F. Instead of solving $F(x) = 0$, we want to solve $\tilde{F}(x) = AF(x) = 0$, where $A \in \mathbb{R}^{N \times N}$ is constant and nonsingular. Of course, this doesn't change the roots $x^\star$.

Theorem (Affine covariance)

Premultiplying F by a constant nonsingular $A \in \mathbb{R}^{N \times N}$ does not change the Newton sequence.

Let $\tilde{F}(x) \coloneqq AF(x)$. Newton's method applied to $\tilde{F}$ from $x_0 = \tilde{x}_0$ generates a sequence

$$\tilde{x}_0, \tilde{x}_1, \tilde{x}_2, \ldots .$$

Proof.

For $i = 0$, we have $x_i = \tilde{x}_i$ by assumption.

Let $\tilde{F}(x) := AF(x)$. Newton's method applied to $\tilde{F}$ from $x_0 = \tilde{x}_0$ generates a sequence

$$\tilde{x}_0, \tilde{x}_1, \tilde{x}_2, \ldots.$$

Proof.

For $i = 0$, we have $x_i = \tilde{x}_i$ by assumption.

Assume the claim is true at iteration i. Then the Newton update for $\tilde{F}$ satisfies

$$[\tilde{F}_x(\tilde{x}_i)]^{-1}\tilde{F}(\tilde{x}_i) = [AF_x(x_i)]^{-1}AF(x_i)$$

Let $\tilde{F}(x) := AF(x)$. Newton's method applied to $\tilde{F}$ from $x_0 = \tilde{x}_0$ generates a sequence

$$\tilde{x}_0, \tilde{x}_1, \tilde{x}_2, \ldots.$$

Proof.

For $i = 0$, we have $x_i = \tilde{x}_i$ by assumption.

Assume the claim is true at iteration i. Then the Newton update for $\tilde{F}$ satisfies

$$\begin{aligned}[\tilde{F}_x(\tilde{x}_i)]^{-1}\tilde{F}(\tilde{x}_i) &= [AF_x(x_i)]^{-1}AF(x_i) \\ &= [F_x(x_i)]^{-1}A^{-1}AF(x_i)\end{aligned}$$

Let $\tilde{F}(x) \coloneqq AF(x)$. Newton's method applied to $\tilde{F}$ from $x_0 = \tilde{x}_0$ generates a sequence

$$\tilde{x}_0, \tilde{x}_1, \tilde{x}_2, \ldots .$$

Proof.

For $i = 0$, we have $x_i = \tilde{x}_i$ by assumption.

Assume the claim is true at iteration i. Then the Newton update for $\tilde{F}$ satisfies

$$\begin{aligned}
[\tilde{F}_x(\tilde{x}_i)]^{-1}\tilde{F}(\tilde{x}_i) &= [AF_x(x_i)]^{-1}AF(x_i) \\
&= [F_x(x_i)]^{-1}A^{-1}AF(x_i) \\
&= [F_x(x_i)]^{-1}F(x_i).
\end{aligned}$$

Let $\tilde{F}(x) := AF(x)$. Newton's method applied to $\tilde{F}$ from $x_0 = \tilde{x}_0$ generates a sequence

$$\tilde{x}_0, \tilde{x}_1, \tilde{x}_2, \ldots.$$

Proof.

For $i = 0$, we have $x_i = \tilde{x}_i$ by assumption.

Assume the claim is true at iteration i. Then the Newton update for $\tilde{F}$ satisfies

$$\begin{aligned}
[\tilde{F}_x(\tilde{x}_i)]^{-1}\tilde{F}(\tilde{x}_i) &= [AF_x(x_i)]^{-1}AF(x_i) \\
&= [F_x(x_i)]^{-1}A^{-1}AF(x_i) \\
&= [F_x(x_i)]^{-1}F(x_i).
\end{aligned}$$

Hence $x_{i+1} = \tilde{x}_{i+1}$, and the result follows by induction.

Let $\tilde{F}(x) \coloneqq AF(x)$. Newton's method applied to $\tilde{F}$ from $x_0 = \tilde{x}_0$ generates a sequence

$$\tilde{x}_0, \tilde{x}_1, \tilde{x}_2, \ldots.$$

Proof.

For $i = 0$, we have $x_i = \tilde{x}_i$ by assumption.

Assume the claim is true at iteration i. Then the Newton update for $\tilde{F}$ satisfies

$$\begin{aligned} [\tilde{F}_x(\tilde{x}_i)]^{-1}\tilde{F}(\tilde{x}_i) &= [AF_x(x_i)]^{-1}AF(x_i) \\ &= [F_x(x_i)]^{-1}A^{-1}AF(x_i) \\ &= [F_x(x_i)]^{-1}F(x_i). \end{aligned}$$

Hence $x_{i+1} = \tilde{x}_{i+1}$, and the result follows by induction.

We get exactly the same iterates $x_0, x_1, \ldots$, whether we apply Newton to $F(x) = 0$ or $AF(x) = 0$.

Why does this matter?

Why does this matter?

Philosophical remark

Since Newton's method is affine covariant, *the conditions for any theorem guaranteeing its convergence* should also be affine covariant.

Why does this matter?

Philosophical remark

Since Newton's method is affine covariant, *the conditions for any theorem guaranteeing its convergence* should also be affine covariant.

This is not true of proofs found in many books!

Why does this matter?

Philosophical remark

Since Newton's method is affine covariant, *the conditions for any theorem guaranteeing its convergence* should also be affine covariant.

This is not true of proofs found in many books!

Moreover, any sensible strategy for globalising the convergence of Newton's method from poor initial guesses x_0 must also preserve this property. This insight leads to the current state of the art for globalising Newton's method.

Peter Deuflhard, 1944–2019

We can visualise the erratic global convergence with a *Newton fractal*.

We can visualise the erratic global convergence with a *Newton fractal*.

Consider the problem

$$\text{find } z \in \mathbb{C} \text{ such that } z^3 - 1 = 0.$$

We could also think of this as a problem in $\mathbb{R}^2$.

We can visualise the erratic global convergence with a *Newton fractal*.

Consider the problem

$$\text{find } z \in \mathbb{C} \text{ such that } z^3 - 1 = 0.$$

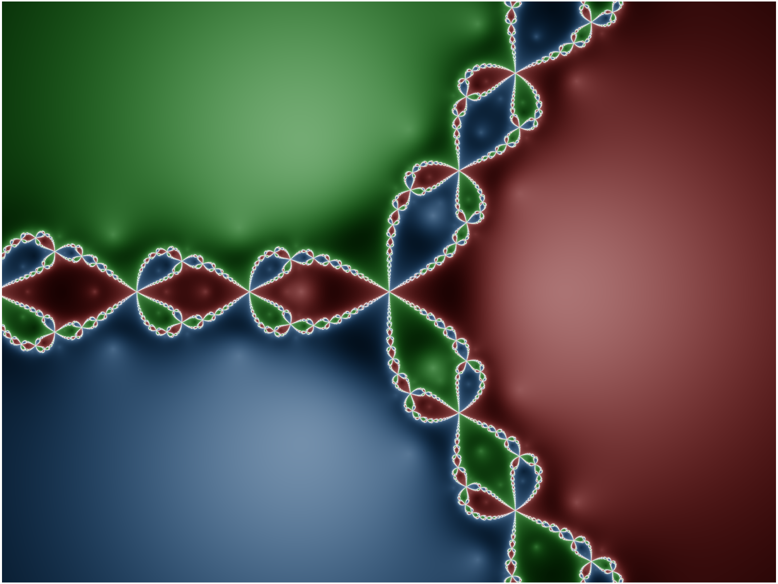
We could also think of this as a problem in $\mathbb{R}^2$.

We know this has three solutions,

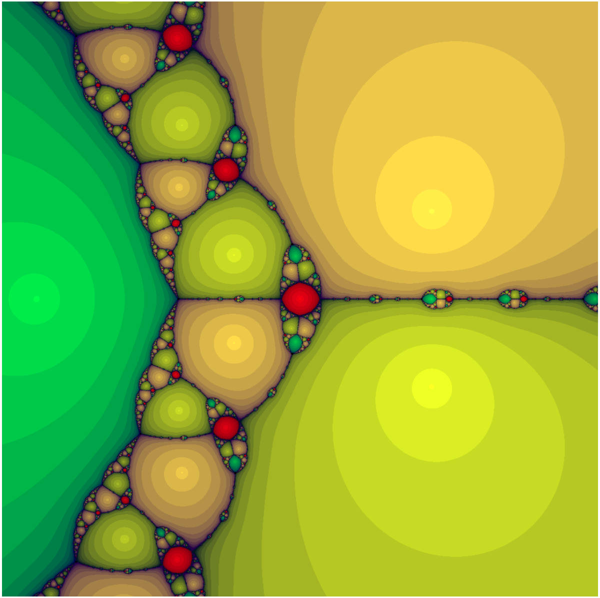
$$z = 1, \quad z = -1/2 + i\sqrt{3}/2, \text{ and } z = -1/2 - i\sqrt{3}/2.$$

Let's take a subset of the complex plane and colour each point as follows. For a given $z_0 \in \mathbb{C}$, we

1. run Newton's method with that initial guess,
2. and colour the point according to which root it converges to.



The Newton fractal for $z^3 - 1 = 0$.



The Newton fractal for $z^3 - 2z + 2 = 0$.

The generalisation of Newton's method to Banach spaces is called the *Newton–Kantorovich* algorithm.

The generalisation of Newton's method to Banach spaces is called the *Newton–Kantorovich* algorithm.

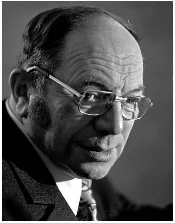
Kantorovich's theorem (1948) is a triumph of both PDE analysis and numerical analysis. It *does not assume the existence of a solution*: given certain conditions on the residual and initial guess, it *proves* the existence and local uniqueness of a solution.

The generalisation of Newton's method to Banach spaces is called the *Newton–Kantorovich* algorithm.

Kantorovich's theorem (1948) is a triumph of both PDE analysis and numerical analysis. It *does not assume the existence of a solution*: given certain conditions on the residual and initial guess, it *proves* the existence and local uniqueness of a solution.

With a good initial guess, and great cleverness, it is possible to devise *computer-assisted proofs* of the existence of solutions to infinite-dimensional nonlinear problems.

- Invented linear programming (via industrial consultancy!).
- Instrumental in saving over a million lives during the siege of Leningrad.
- Involved in the Soviet nuclear bomb project.
- Nearly sent to the gulag for “shadow prices”.
- Pseudo-Nobel prize in Economics (1975).



Leonid Kantorovich, 1912–1986

Theorem (Kantorovich (1948))

Let $F \in C^1(\Omega, Y)$ for open convex $\Omega \subset X$. Given $u_0 \in \Omega$, assume

1. $F_u(u_0)^{-1}$ *exists and set* $\alpha := \|F_u(u_0)^{-1} F(u_0)\|$;

Theorem (Kantorovich (1948))

Let $F \in C^1(\Omega, Y)$ *for open convex* $\Omega \subset X$*. Given* $u_0 \in \Omega$*, assume*

1. $F_u(u_0)^{-1}$ *exists and set* $\alpha := \|F_u(u_0)^{-1} F(u_0)\|$*;*
2. $\|F_u(u_0)^{-1} \left(F_u(v) - F_u(w)\right)\| \leq \omega_0 \|v - w\|$ *for all* $v, w \in \Omega$*;*

Theorem (Kantorovich (1948))

Let $F \in C^1(\Omega, Y)$ for open convex $\Omega \subset X$. Given $u_0 \in \Omega$, assume

1. $F_u(u_0)^{-1}$ *exists and set* $\alpha \coloneqq \|F_u(u_0)^{-1}F(u_0)\|$;
2. $\|F_u(u_0)^{-1}\left(F_u(v) - F_u(w)\right)\| \leq \omega_0\|v - w\|$ *for all* $v, w \in \Omega$;
3. $h_0 \coloneqq \alpha\omega_0 \leq \frac{1}{2}$;

Theorem (Kantorovich (1948))

Let $F \in C^1(\Omega, Y)$ for open convex $\Omega \subset X$. Given $u_0 \in \Omega$, assume

1. $F_u(u_0)^{-1}$ *exists and set* $\alpha := \|F_u(u_0)^{-1} F(u_0)\|$;
2. $\|F_u(u_0)^{-1} \left(F_u(v) - F_u(w)\right)\| \leq \omega_0 \|v - w\|$ *for all* $v, w \in \Omega$;
3. $h_0 := \alpha\omega_0 \leq \frac{1}{2}$;
4. $\overline{B(u_0, \rho_0)} \subset \Omega$ *for* $\rho_0 := (1 - \sqrt{1 - 2h_0})/\omega_0$.

Theorem (Kantorovich (1948))

Let $F \in C^1(\Omega, Y)$ *for open convex* $\Omega \subset X$*. Given* $u_0 \in \Omega$*, assume*

1. $F_u(u_0)^{-1}$ *exists and set* $\alpha := \|F_u(u_0)^{-1} F(u_0)\|$*;*
2. $\|F_u(u_0)^{-1}\left(F_u(v) - F_u(w)\right)\| \leq \omega_0 \|v - w\|$ *for all* $v, w \in \Omega$*;*
3. $h_0 := \alpha\omega_0 \leq \frac{1}{2}$*;*
4. $\overline{B(u_0, \rho_0)} \subset \Omega$ *for* $\rho_0 := (1 - \sqrt{1 - 2h_0})/\omega_0$*.*

Then the Newton sequence defined by

$$u_{k+1} = u_k - F_u(u_k)^{-1} F(u_k)$$

is well defined and remains within $\overline{B(u_0, \rho_0)}$*.*

Theorem (Kantorovich (1948))

Let $F \in C^1(\Omega, Y)$ for open convex $\Omega \subset X$. Given $u_0 \in \Omega$, assume

1. $F_u(u_0)^{-1}$ *exists and set* $\alpha := \|F_u(u_0)^{-1} F(u_0)\|$;
2. $\|F_u(u_0)^{-1} \left(F_u(v) - F_u(w)\right)\| \leq \omega_0 \|v - w\|$ *for all* $v, w \in \Omega$;
3. $h_0 := \alpha\omega_0 \leq \frac{1}{2}$;
4. $\overline{B(u_0, \rho_0)} \subset \Omega$ *for* $\rho_0 := (1 - \sqrt{1 - 2h_0})/\omega_0$.

Then the Newton sequence defined by

$$u_{k+1} = u_k - F_u(u_k)^{-1} F(u_k)$$

is well defined and remains within $\overline{B(u_0, \rho_0)}$.

There exists $u^\star \in \overline{B(u_0, \rho_0)}$ *which solves* $F(u^\star) = 0$, *and* $(u_k) \to u^\star$.

Theorem (Kantorovich (1948))

Let $F \in C^1(\Omega, Y)$ for open convex $\Omega \subset X$. Given $u_0 \in \Omega$, assume

1. $F_u(u_0)^{-1}$ *exists and set* $\alpha := \|F_u(u_0)^{-1}F(u_0)\|$;
2. $\|F_u(u_0)^{-1}\left(F_u(v) - F_u(w)\right)\| \leq \omega_0\|v - w\|$ *for all* $v, w \in \Omega$;
3. $h_0 := \alpha\omega_0 \leq \frac{1}{2}$;
4. $\overline{B(u_0, \rho_0)} \subset \Omega$ *for* $\rho_0 := (1 - \sqrt{1 - 2h_0})/\omega_0$.

Then the Newton sequence defined by

$$u_{k+1} = u_k - F_u(u_k)^{-1}F(u_k)$$

is well defined and remains within $\overline{B(u_0, \rho_0)}$.

There exists $u^\star \in \overline{B(u_0, \rho_0)}$ *which solves* $F(u^\star) = 0$, *and* $(u_k) \to u^\star$.

The solution $u^\star$ *is unique in* $\Omega \cap B(u_0, \rho^+)$ *for a* $\rho^+ > \rho_0$.

Subsection 2

Rall–Rheinboldt

The Newton–Kantorovich theorem is very powerful because you only need to check conditions on the initial guess (and a ball around it).

The Newton–Kantorovich theorem is very powerful because you only need to check conditions on the initial guess (and a ball around it).

If you *assume* the existence of roots, one gets a slightly different theory that is also useful. This allows us to place *balls* around the roots, such that if the Newton sequence starts within a ball, Newton's method converges to the associated root.

Theorem (Rall–Rheinboldt (1974))

Let $F \in C^1(\Omega, Y)$ *for open convex* $\Omega \subset X$. *Let* $u^\star \in \Omega$ *such that* $F(u^\star) = 0$. *Assume that*

1. $F_u(u^\star)^{-1}$ *exists;*

Louis B. Rall, 1930–

Werner C. Rheinboldt, ?–?

Theorem (Rall–Rheinboldt (1974))

Let $F \in C^1(\Omega, Y)$ *for open convex* $\Omega \subset X$*. Let* $u^\star \in \Omega$ *such that* $F(u^\star) = 0$*. Assume that*

1. $F_u(u^\star)^{-1}$ *exists;*
2. $\|F_u(u^\star)^{-1}(F_u(v) - F_u(w))\| \leq \omega^\star \|v - w\|$ *for all* $v, w \in \Omega$*.*

Louis B. Rall, 1930–

Werner C. Rheinboldt, ?–?

Theorem (Rall–Rheinboldt (1974))

Let $F \in C^1(\Omega, Y)$ for open convex $\Omega \subset X$. Let $u^\star \in \Omega$ such that $F(u^\star) = 0$. Assume that

1. $F_u(u^\star)^{-1}$ *exists;*
2. $\|F_u(u^\star)^{-1}\left(F_u(v) - F_u(w)\right)\| \leq \omega^\star \|v - w\|$ *for all* $v, w \in \Omega$.

Then for any $u_0 \in B(u^\star, 2/(3\omega^\star))$, the Newton sequence is well-defined and remains within the ball.

Louis B. Rall, 1930–

Werner C. Rheinboldt, ?–?

Theorem (Rall–Rheinboldt (1974))

Let $F \in C^1(\Omega, Y)$ for open convex $\Omega \subset X$. Let $u^\star \in \Omega$ such that $F(u^\star) = 0$. Assume that

1. $F_u(u^\star)^{-1}$ *exists;*
2. $\|F_u(u^\star)^{-1}\left(F_u(v) - F_u(w)\right)\| \le \omega^\star \|v - w\|$ *for all* $v, w \in \Omega$.

Then for any $u_0 \in B(u^\star, 2/(3\omega^\star))$, the Newton sequence is well-defined and remains within the ball.

The Newton sequence converges to $u^\star$.

Louis B. Rall, 1930–

Werner C. Rheinboldt, ?–?

Theorem (Rall–Rheinboldt (1974))

Let $F \in C^1(\Omega, Y)$ for open convex $\Omega \subset X$. Let $u^\star \in \Omega$ such that $F(u^\star) = 0$. Assume that

1. $F_u(u^\star)^{-1}$ *exists;*
2. $\|F_u(u^\star)^{-1}(F_u(v) - F_u(w))\| \leq \omega^\star \|v - w\|$ *for all* $v, w \in \Omega$.

Then for any $u_0 \in B(u^\star, 2/(3\omega^\star))$, the Newton sequence is well-defined and remains within the ball.

The Newton sequence converges to $u^\star$.

The solution $u^\star$ is unique within $\Omega \cap B(u^\star, 1/\omega^\star)$.

Louis B. Rall, 1930–

Werner C. Rheinboldt, ?–?

Let's examine the conditions of the theorems for a simple case:

$$f : \mathbb{C} \to \mathbb{C}, \quad f(z) = (z-1)(z+1).$$

Let's examine the conditions of the theorems for a simple case:

$$f : \mathbb{C} \to \mathbb{C}, \quad f(z) = (z-1)(z+1).$$

We have $f'(z) = 2z$.

Let's examine the conditions of the theorems for a simple case:

$$f : \mathbb{C} \to \mathbb{C}, \quad f(z) = (z-1)(z+1).$$

We have $f'(z) = 2z$.

Newton–Kantorovich

For $z_0 \neq 0$, we calculate

$$\alpha := |f'(z_0)^{-1} f(z_0)| = |z_0^2 - 1|/2|z_0|,$$

Let's examine the conditions of the theorems for a simple case:

$$f : \mathbb{C} \to \mathbb{C}, \quad f(z) = (z-1)(z+1).$$

We have $f'(z) = 2z$.

Newton–Kantorovich

For $z_0 \neq 0$, we calculate

$$\alpha := |f'(z_0)^{-1} f(z_0)| = |z_0^2 - 1|/2|z_0|,$$

and by inspecting

$$|(2z_0)^{-1}(2v - 2w)| \leq \omega_0 |v - w|$$

Let's examine the conditions of the theorems for a simple case:

$$f : \mathbb{C} \to \mathbb{C}, \quad f(z) = (z-1)(z+1).$$

We have $f'(z) = 2z$.

Newton–Kantorovich

For $z_0 \neq 0$, we calculate

$$\alpha := |f'(z_0)^{-1} f(z_0)| = |z_0^2 - 1|/2|z_0|,$$

and by inspecting

$$|(2z_0)^{-1}(2v - 2w)| \leq \omega_0 |v - w|$$

we find $\omega_0 = 1/|z_0|$.

Let's examine the conditions of the theorems for a simple case:

$$f : \mathbb{C} \to \mathbb{C}, \quad f(z) = (z-1)(z+1).$$

We have $f'(z) = 2z$.

Newton–Kantorovich

For $z_0 \neq 0$, we calculate

$$\alpha := |f'(z_0)^{-1} f(z_0)| = |z_0^2 - 1|/2|z_0|,$$

and by inspecting

$$|(2z_0)^{-1}(2v - 2w)| \leq \omega_0 |v - w|$$

we find $\omega_0 = 1/|z_0|$.

We need $\alpha\omega_0 \leq 1/2$, so Newton–Kantorovich guarantees convergence for

$$\frac{|z_0^2 - 1|}{2|z_0|^2} \leq \frac{1}{2} \implies |1 - z_0^{-2}| \leq 1.$$

Rall–Rheinboldt

The affine covariant Lipschitz constant for $z^\star = 1$ is $1/|z^\star| = 1$.

Rall–Rheinboldt

The affine covariant Lipschitz constant for $z^\star = 1$ is $1/|z^\star| = 1$.

So Rall–Rheinboldt guarantees convergence for

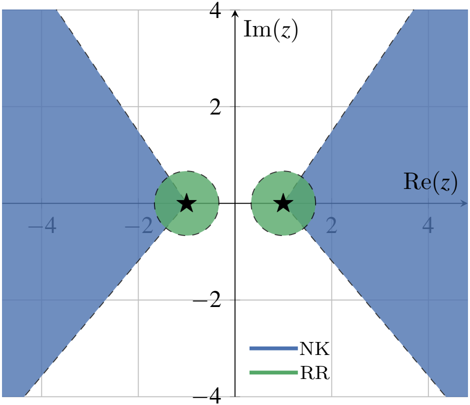
$$|z_0 - 1| \leq 2/3.$$

Rall–Rheinboldt

The affine covariant Lipschitz constant for $z^\star = 1$ is $1/|z^\star| = 1$.

So Rall–Rheinboldt guarantees convergence for

$$|z_0 - 1| \leq 2/3.$$

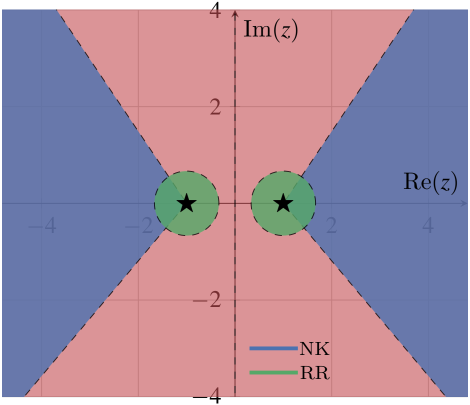


Rall–Rheinboldt

The affine covariant Lipschitz constant for $z^\star = 1$ is $1/|z^\star| = 1$.

So Rall–Rheinboldt guarantees convergence for

$$|z_0 - 1| \leq 2/3.$$



Subsection 3

The Implicit Function Theorem

Newton's method allows us to lock in on one solution u_0 for a fixed parameter value λ_0, given some initial guess near u_0.

Newton's method allows us to lock in on one solution u_0 for a fixed parameter value λ_0, given some initial guess near u_0.

When does the existence of (u_0, λ_0) such that $F(u_0, \lambda_0) = 0$ imply that we can solve F for nearby values of λ?

Newton's method allows us to lock in on one solution u_0 for a fixed parameter value λ_0, given some initial guess near u_0.

When does the existence of (u_0, λ_0) such that $F(u_0, \lambda_0) = 0$ imply that we can solve F for nearby values of λ?

An answer . . .

. . . is given by the Implicit Function Theorem.

Newton's method allows us to lock in on one solution u_0 for a fixed parameter value λ_0, given some initial guess near u_0.

When does the existence of (u_0, λ_0) such that $F(u_0, \lambda_0) = 0$ imply that we can solve F for nearby values of λ?

An answer ...

... is given by the Implicit Function Theorem.

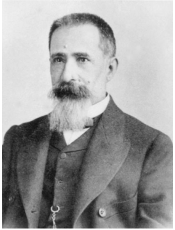
Basically, if $F_u(u_0, \lambda_0)$ is invertible, then you can continue $u = H(\lambda)$ for some interval $(\lambda_0 - \delta, \lambda_0 + \delta)$.

Theorem (Implicit Function Theorem)

Assume that $\Omega \subset X \times \mathbb{R}$ *is open. Let* $F \in C^1(\Omega, Y)$.

Let $(u_0, \lambda_0) \in \Omega$ *such that* $F(u_0, \lambda_0) = 0$ *with* $F_u(u_0, \lambda_0)$ *invertible.*

Then



Ulisse Dini, 1845–1918

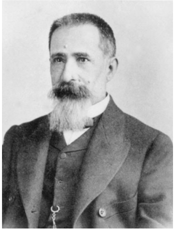
Theorem (Implicit Function Theorem)

Assume that $\Omega \subset X \times \mathbb{R}$ *is open. Let* $F \in C^1(\Omega, Y)$.

Let $(u_0, \lambda_0) \in \Omega$ *such that* $F(u_0, \lambda_0) = 0$ *with* $F_u(u_0, \lambda_0)$ *invertible.*

Then

1. *there exist* $\varepsilon, \delta > 0$ *and* $H \in C(B(\lambda_0, \delta), B(u_0, \varepsilon))$ *such that* $(H(\lambda), \lambda)$ *is the unique solution of* $F(u, \lambda) = 0$ *in* $B(\lambda_0, \delta) \times B(u_0, \varepsilon)$;



Ulisse Dini, 1845–1918

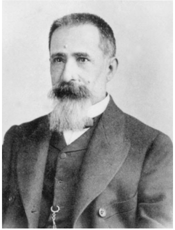
Theorem (Implicit Function Theorem)

Assume that $\Omega \subset X \times \mathbb{R}$ *is open. Let* $F \in C^1(\Omega, Y)$.

Let $(u_0, \lambda_0) \in \Omega$ *such that* $F(u_0, \lambda_0) = 0$ *with* $F_u(u_0, \lambda_0)$ *invertible.*

Then

1. *there exist* $\varepsilon, \delta > 0$ *and* $H \in C(B(\lambda_0, \delta), B(u_0, \varepsilon))$ *such that* $(H(\lambda), \lambda)$ *is the unique solution of* $F(u, \lambda) = 0$ *in* $B(\lambda_0, \delta) \times B(u_0, \varepsilon)$*;*
2. *if* $F \in C^k(\Omega, Y)$, *then* $H \in C^k(B(\lambda_0, \delta), X)$*;*



Ulisse Dini, 1845–1918

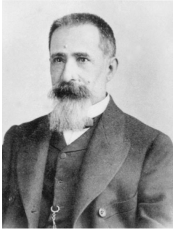
Theorem (Implicit Function Theorem)

Assume that $\Omega \subset X \times \mathbb{R}$ *is open. Let* $F \in C^1(\Omega, Y)$.

Let $(u_0, \lambda_0) \in \Omega$ *such that* $F(u_0, \lambda_0) = 0$ *with* $F_u(u_0, \lambda_0)$ *invertible.*

Then

1. *there exist* $\varepsilon, \delta > 0$ *and* $H \in C(B(\lambda_0, \delta), B(u_0, \varepsilon))$ *such that* $(H(\lambda), \lambda)$ *is the unique solution of* $F(u, \lambda) = 0$ *in* $B(\lambda_0, \delta) \times B(u_0, \varepsilon)$;
2. *if* $F \in C^k(\Omega, Y)$, *then* $H \in C^k(B(\lambda_0, \delta), X)$;
3. *if* F *is analytic,* H *is analytic.*



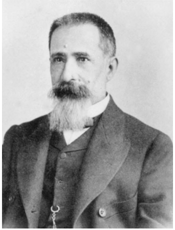
Ulisse Dini, 1845–1918

The history is reviewed in

A Historical Outline of the Theorem of Implicit Functions

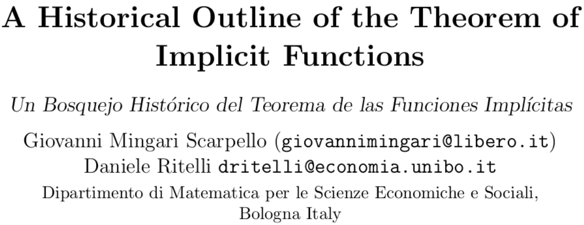
Un Bosquejo Histórico del Teorema de las Funciones Implícitas

Giovanni Mingari Scarpello (giovannimingari@libero.it)
Daniele Ritelli dritelli@economia.unibo.it
Dipartimento di Matematica per le Scienze Economiche e Sociali,
Bologna Italy



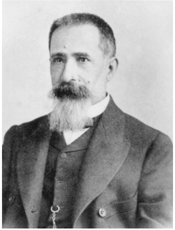
Ulisse Dini, 1845–1918

The history is reviewed in


A Historical Outline of the Theorem of Implicit Functions

Un Bosquejo Histórico del Teorema de las Funciones Implícitas

Giovanni Mingari Scarpello (giovannimingari@libero.it)
Daniele Ritelli dritelli@economia.unibo.it
Dipartimento di Matematica per le Scienze Economiche e Sociali,
Bologna Italy



Ulisse Dini, 1845–1918

which complains

> *Anglo–Saxon scientific and historic literature* ***ignores*** *the Italian mathematician U. Dini.*

Main message

If we want to find where local uniqueness breaks down, look for (u, λ) such that $F_u(u, \lambda)$ not invertible.

Main message

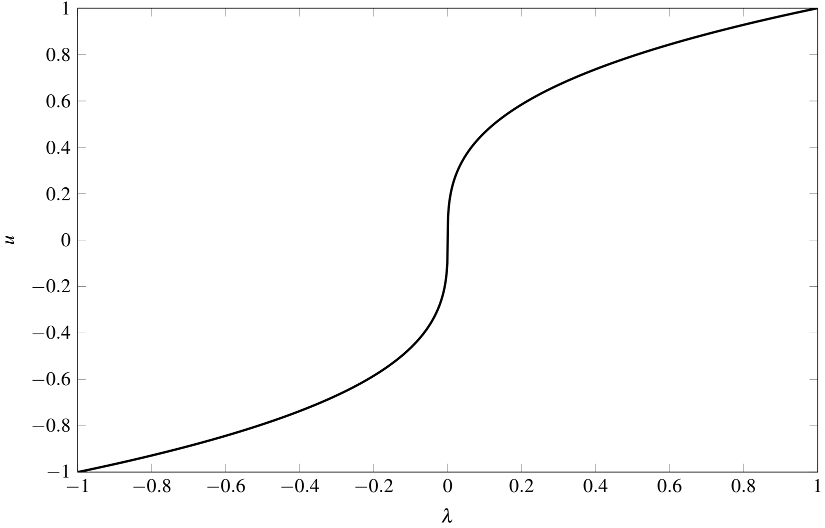
If we want to find where local uniqueness breaks down, look for (u, λ) such that $F_u(u, \lambda)$ not invertible.

Note

$F_u(u, \lambda)$ invertible is *sufficient* for the existence of a local resolution $u = u(\lambda)$, but not *necessary*.

Consider $F(u, \lambda) = u^3 - \lambda$.

Consider $F(u, \lambda) = u^3 - \lambda$.



$F_u(0, 0) = 0$, but the resolution $u = H(\lambda) = \sqrt[3]{\lambda}$ is unique regardless.

Section 4

Examples

Let's see more examples of what can happen when the IFT does not apply.

Fold bifurcation

$$F(u, \lambda) = \lambda - u^2 = 0$$

Fold bifurcation

$$F(u, \lambda) = \lambda - u^2 = 0$$

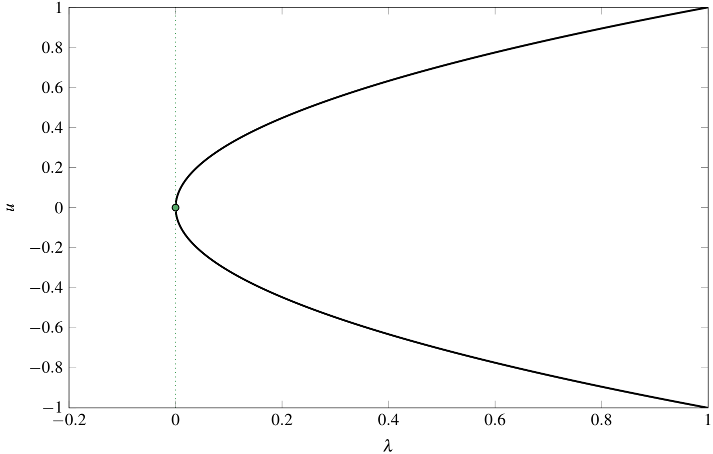
This has solutions

$$u = \pm\sqrt{\lambda}, \quad \lambda \geq 0,$$

and no solutions otherwise.

Fold bifurcation

$$F(u, \lambda) = \lambda - u^2 = 0$$



$F_u(0, 0) = 0$. A branch of solutions is born at a *fold bifurcation*.

Transcritical bifurcation

$$F(u,\lambda) = \lambda u + u^2 = 0$$

Transcritical bifurcation

$$F(u, \lambda) = \lambda u + u^2 = 0$$

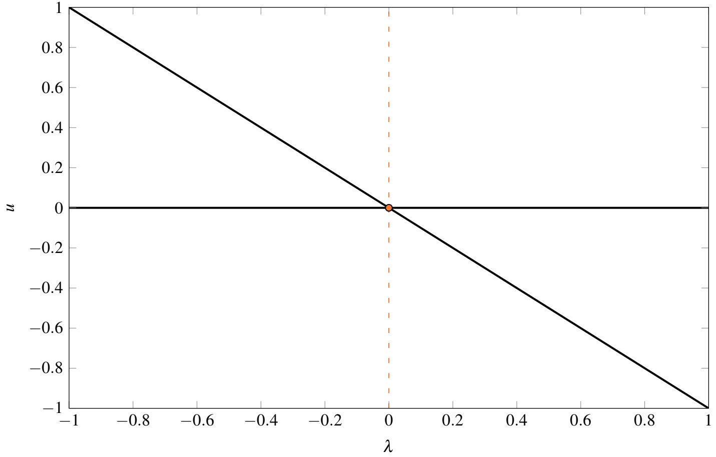
This has solutions

$$u = 0, \quad u = -\lambda$$

for all values of λ.

Transcritical bifurcation

$$F(u,\lambda) = \lambda u + u^2 = 0$$



Two branches cross at a *transcritical bifurcation*.

Pitchfork bifurcation

$$F(u, \lambda) = \lambda u - u^3 = 0$$

Pitchfork bifurcation

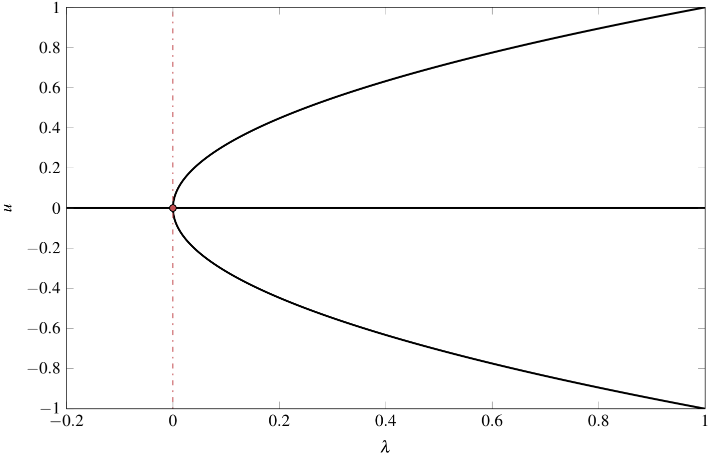
$$F(u, \lambda) = \lambda u - u^3 = 0$$

This has solutions

$$\begin{aligned} u &= 0, & \lambda &\in \mathbb{R}, \\ u &= \pm\sqrt{\lambda}, & \lambda &\geq 0. \end{aligned}$$

Pitchfork bifurcation

$$F(u, \lambda) = \lambda u - u^3 = 0$$



Two branches emerge from the base branch at a *pitchfork bifurcation*.

Structural stability of folds

Fold bifurcations are structurally stable.

Structural stability of folds

Fold bifurcations are structurally stable.

Structural stability of transcritical and pitchfork bifurcations

Transcritical and pitchfork bifurcations are not.

Structural stability of folds

Fold bifurcations are structurally stable.

Structural stability of transcritical and pitchfork bifurcations

Transcritical and pitchfork bifurcations are not.

Numerical implications

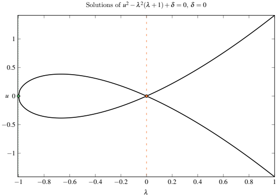
This will have major consequences for our algorithms.

Perturbing a fold + transcritical bifurcation

$$F(u, \lambda) = u^2 - \lambda^2(\lambda + 1) + \delta = 0$$

Perturbing a fold + transcritical bifurcation

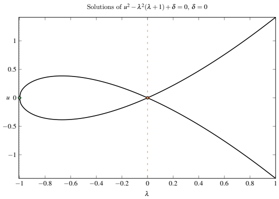
$$F(u,\lambda) = u^2 - \lambda^2(\lambda+1) + \delta = 0$$



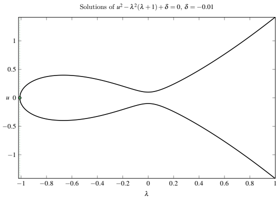
$$\delta = 0$$

Perturbing a fold + transcritical bifurcation

$$F(u, \lambda) = u^2 - \lambda^2(\lambda + 1) + \delta = 0$$



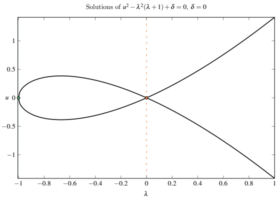
$\delta = 0$



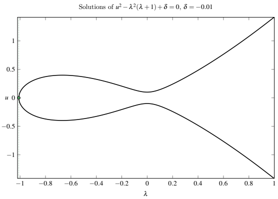
$\delta < 0$

Perturbing a fold + transcritical bifurcation

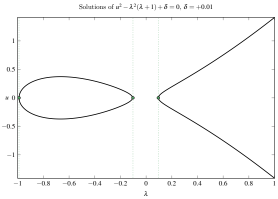
$$F(u,\lambda) = u^2 - \lambda^2(\lambda+1) + \delta = 0$$



$\delta = 0$



$\delta < 0$



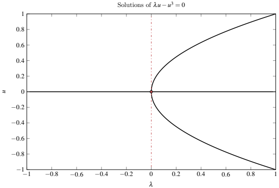
$\delta > 0$

Perturbing a pitchfork bifurcation

$$F(u, \lambda) = \lambda u - u^3 + \delta = 0$$

Perturbing a pitchfork bifurcation

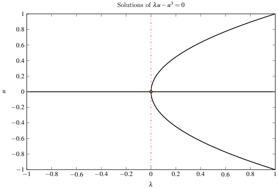
$$F(u, \lambda) = \lambda u - u^3 + \delta = 0$$



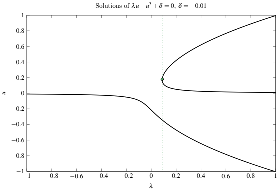
$$\delta = 0$$

Perturbing a pitchfork bifurcation

$$F(u,\lambda) = \lambda u - u^3 + \delta = 0$$



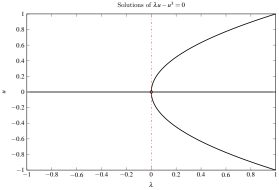
$\delta = 0$



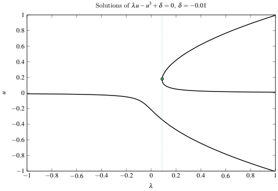
$\delta < 0$

Perturbing a pitchfork bifurcation

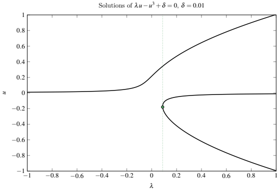
$$F(u, \lambda) = \lambda u - u^3 + \delta = 0$$



$\delta = 0$



$\delta < 0$



$\delta > 0$

These examples motivate the following definition.

Bifurcation point

A bifurcation point $P = (u^\star, \lambda^\star)$ is one where, for all neighbourhoods N containing P, there exists a $\lambda \in \mathbb{R}$ such that $F(u, \lambda) = 0$ has nonunique solutions within N.

These examples motivate the following definition.

Bifurcation point

A bifurcation point $P = (u^\star, \lambda^\star)$ is one where, for all neighbourhoods N containing P, there exists a $\lambda \in \mathbb{R}$ such that $F(u, \lambda) = 0$ has nonunique solutions within N.

In the next lectures, we will study the key question:

How do we compute these bifurcation diagrams?

Lecture 2: Classical algorithms of bifurcation analysis

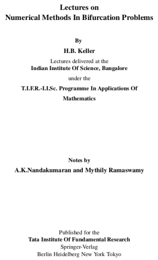
Patrick E. Farrell

University of Oxford

May 30

Challenge

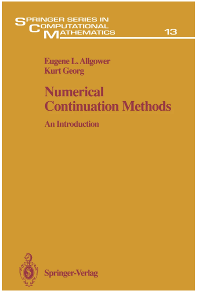
How do we continue branches? How do we detect and pursue bifurcations?

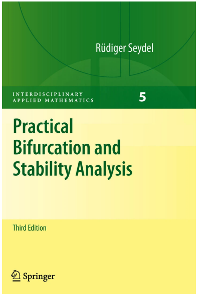

Lectures on
Numerical Methods In Bifurcation Problems

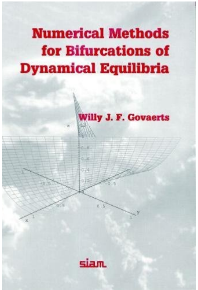
By
H.B. Keller
Lectures delivered at the
Indian Institute Of Science, Bangalore
under the
T.I.F.R.-I.I.Sc. Programme In Applications Of
Mathematics

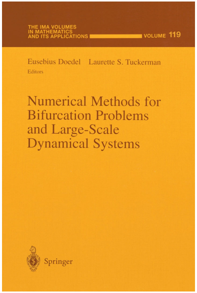
Notes by
A.K.Nandakumaran and Mythily Ramaswamy

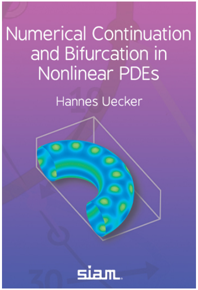
Published for the
Tata Institute Of Fundamental Research
Springer-Verlag
Berlin Heidelberg New York Tokyo









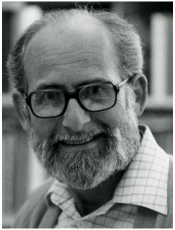


Primary references.

Basic idea of numerical bifurcation analysis:

procedure ANALYSE(u_0, λ_0)

end procedure

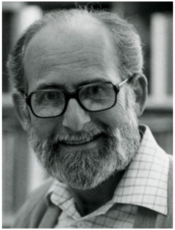


Herbert Keller, 1925–2008

Basic idea of numerical bifurcation analysis:

procedure ANALYSE(u_0, λ_0)
    continue branch of solutions;

end procedure



Herbert Keller, 1925–2008

Continuation

Extending our knowledge of the branch to other values of λ.

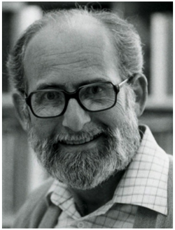
Basic idea of numerical bifurcation analysis:

procedure ANALYSE(u_0, λ_0)

  continue branch of solutions;

  detect bifurcations on the branch;

end procedure



Herbert Keller, 1925–2008

Bifurcation detection

Discovering when a bifurcation has occurred on the branch.

Basic idea of numerical bifurcation analysis:

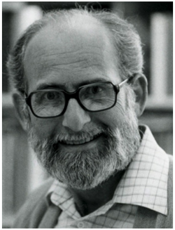
procedure ANALYSE(u_0, λ_0)

continue branch of solutions;

detect bifurcations on the branch;

localise bifurcations;

end procedure



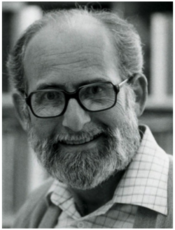
Herbert Keller, 1925–2008

Bifurcation localisation

Identifying precisely the bifurcation point.

Basic idea of numerical bifurcation analysis:

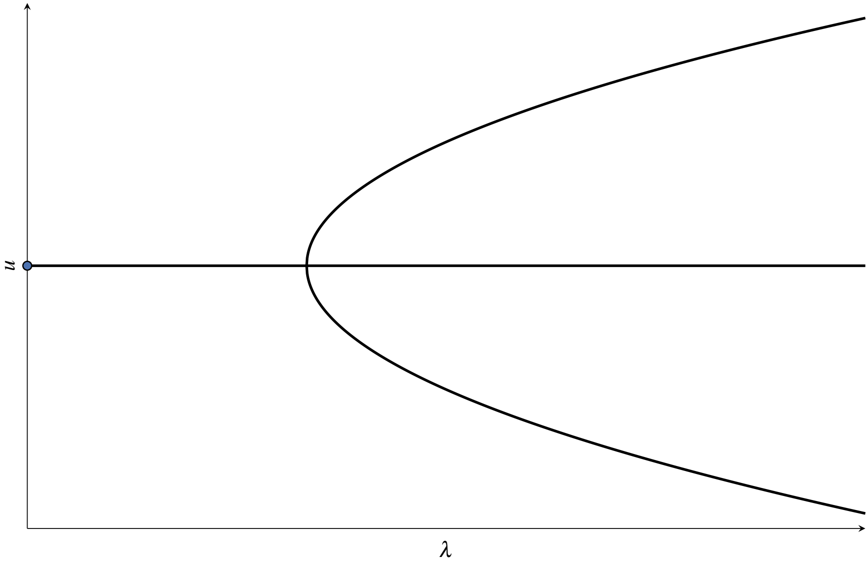
procedure ANALYSE(u_0, λ_0)
    continue branch of solutions;
    detect bifurcations on the branch;
    localise bifurcations;
    switch branches at bifurcations, and recurse.
end procedure



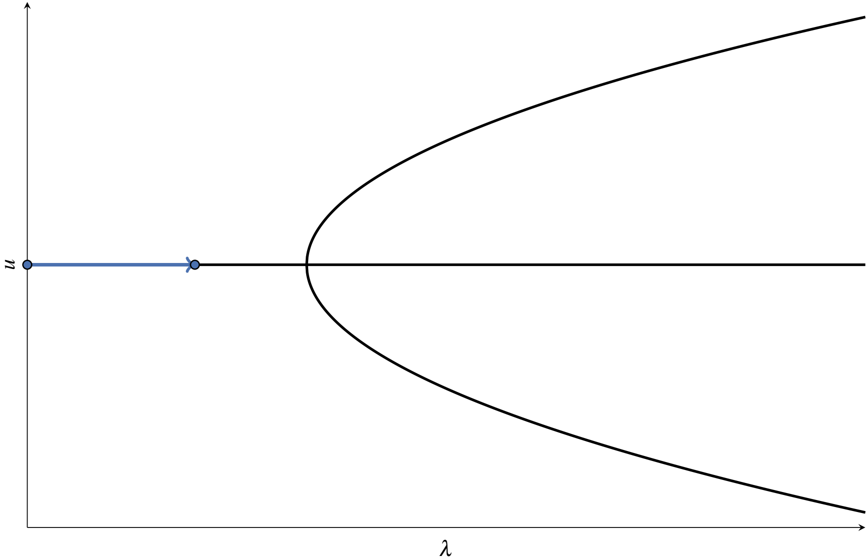
Herbert Keller, 1925–2008

Branch switching

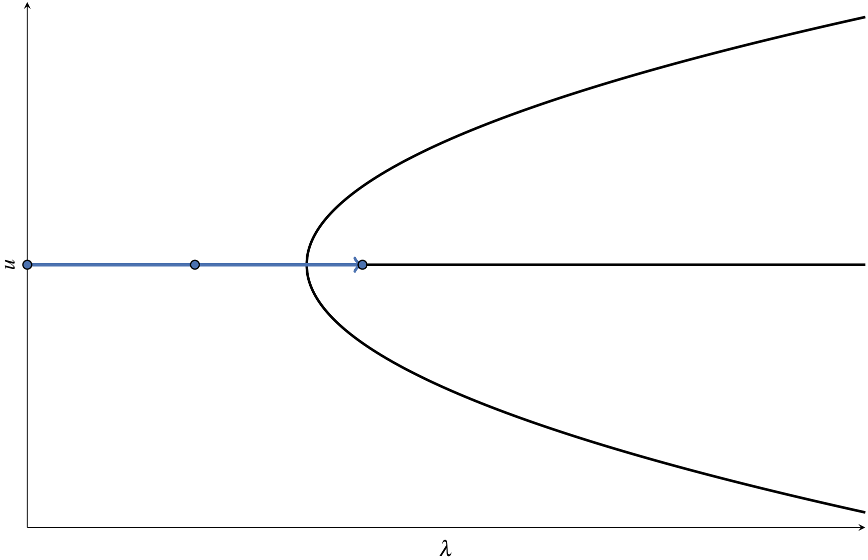
Constructing the emanating branches, and analysing them recursively.



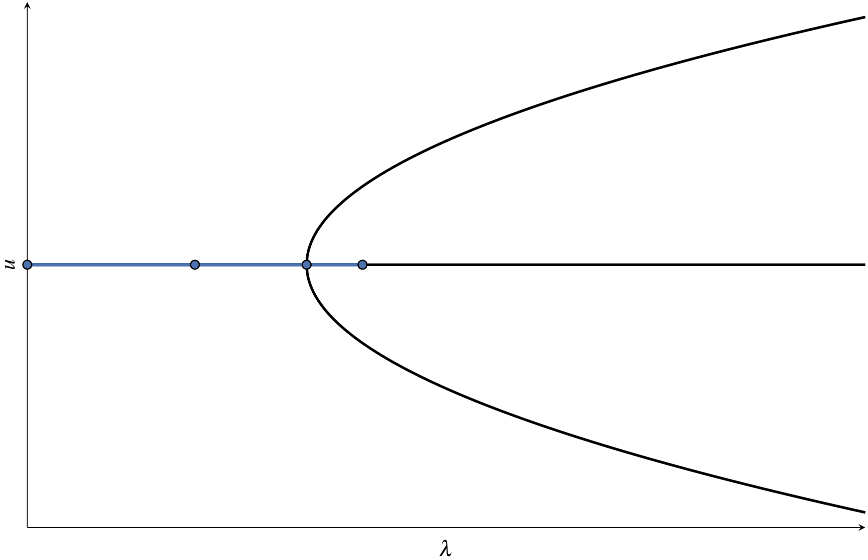
Start with (u_0, λ_0).



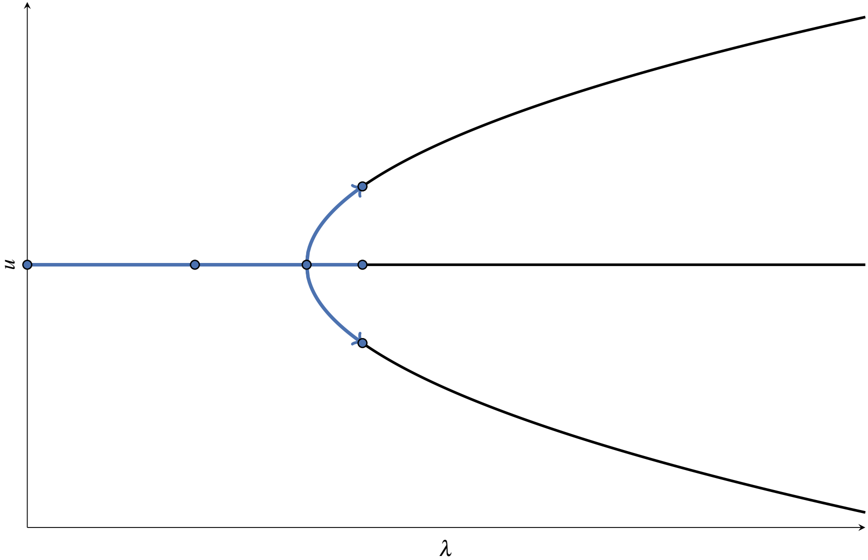
Perform a continuation step.



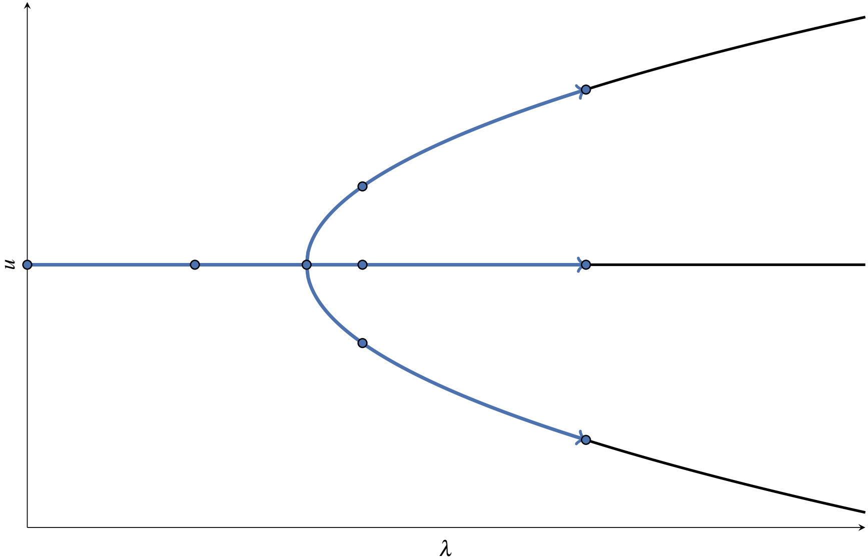
Detect we have passed a bifurcation.



Localise bifurcation point.



Switch branches.



Apply recursively.

Section 1

Continuation algorithms

Suppose we know (u_0, λ_0), with $F_u(u_0, \lambda_0)$ invertible. By the IFT we know we can continue the branch for other values of λ.

Suppose we know (u_0, λ_0), with $F_u(u_0, \lambda_0)$ invertible. By the IFT we know we can continue the branch for other values of λ.

How should we do so? We will meet five algorithms:

- natural (or naïve, or first-order) continuation;

Suppose we know (u_0, λ_0), with $F_u(u_0, \lambda_0)$ invertible. By the IFT we know we can continue the branch for other values of λ.

How should we do so? We will meet five algorithms:

- natural (or naïve, or first-order) continuation;
- tangent (or second-order) continuation, and secant continuation;

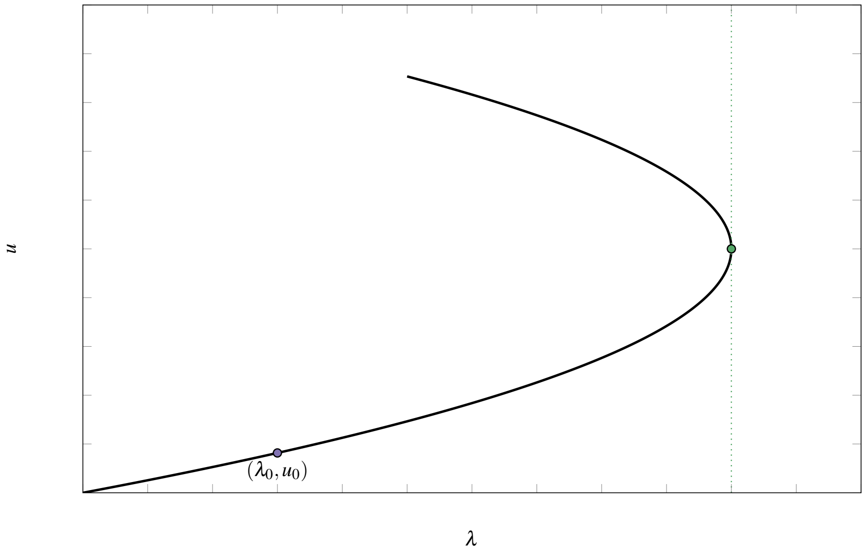
Suppose we know (u_0, λ_0), with $F_u(u_0, \lambda_0)$ invertible. By the IFT we know we can continue the branch for other values of λ.

How should we do so? We will meet five algorithms:

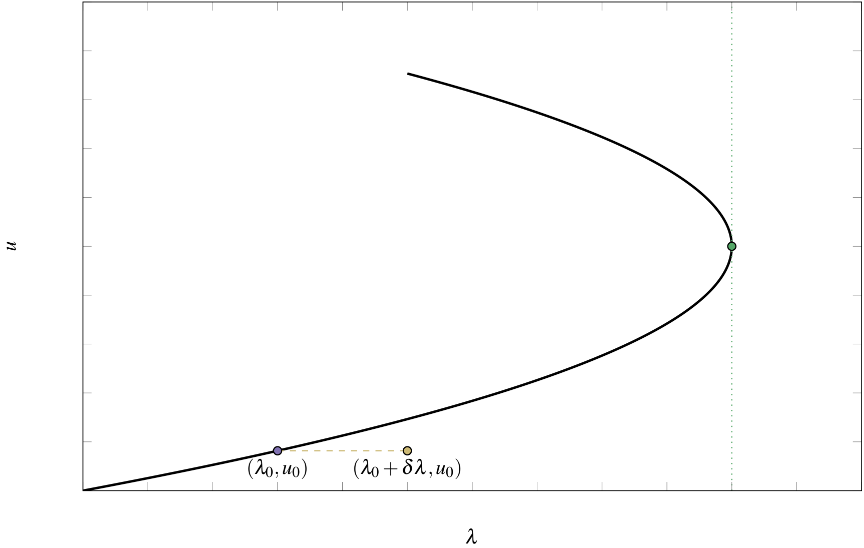
- natural (or naïve, or first-order) continuation;
- tangent (or second-order) continuation, and secant continuation;
- arclength continuation, and pseudo-arclength continuation.

Subsection 1

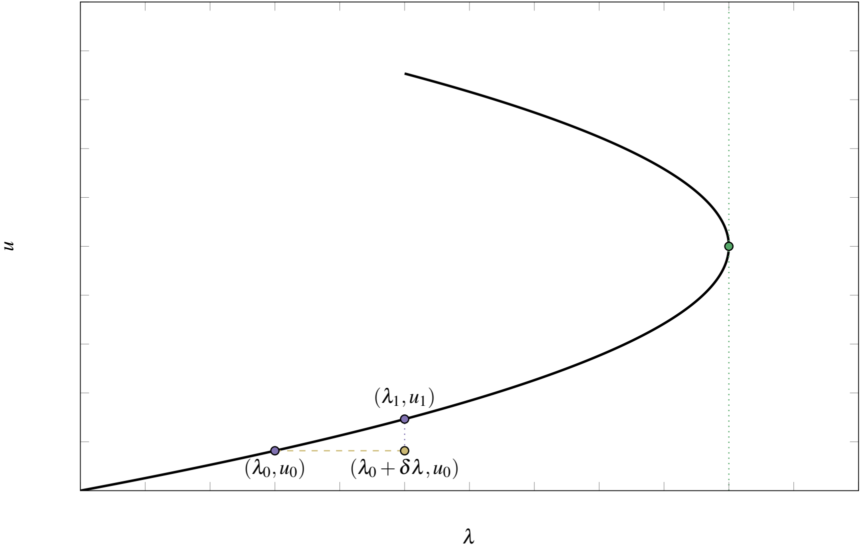
Natural continuation



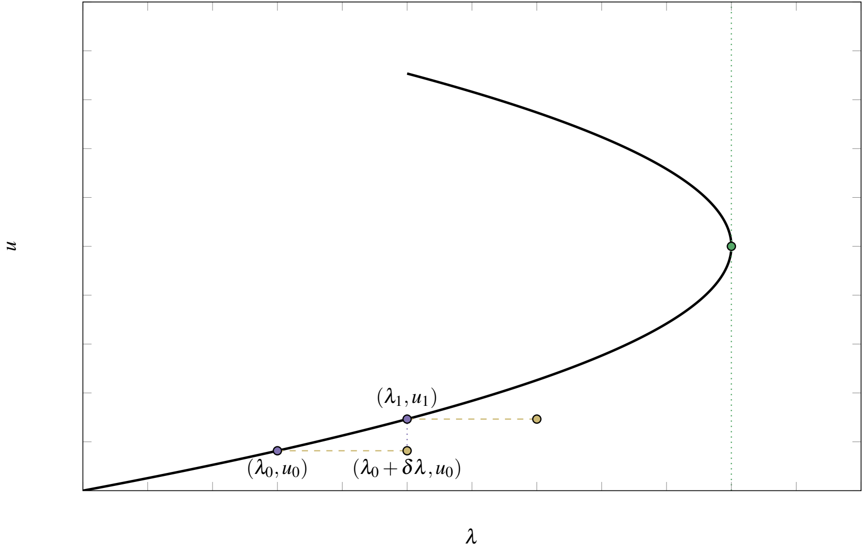
Start with (u_0, λ_0).



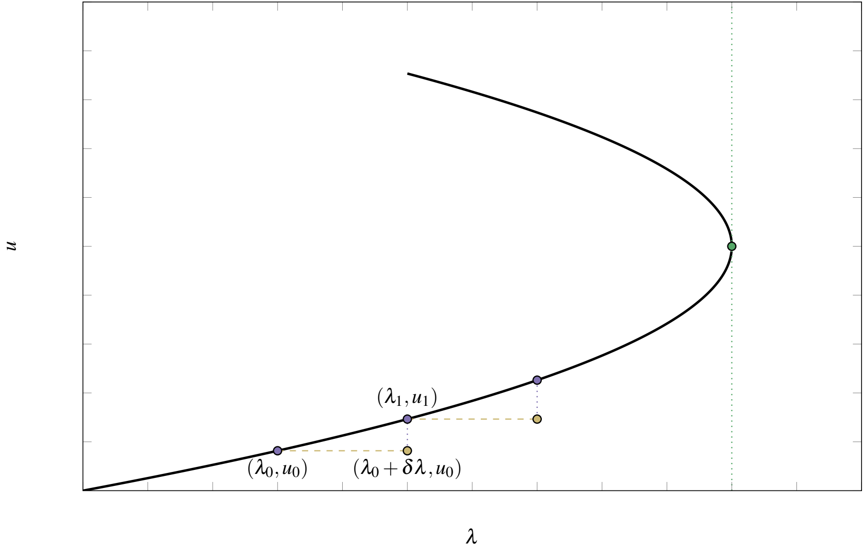
Set u_0 as our guess for $\lambda_0 + \delta\lambda$.



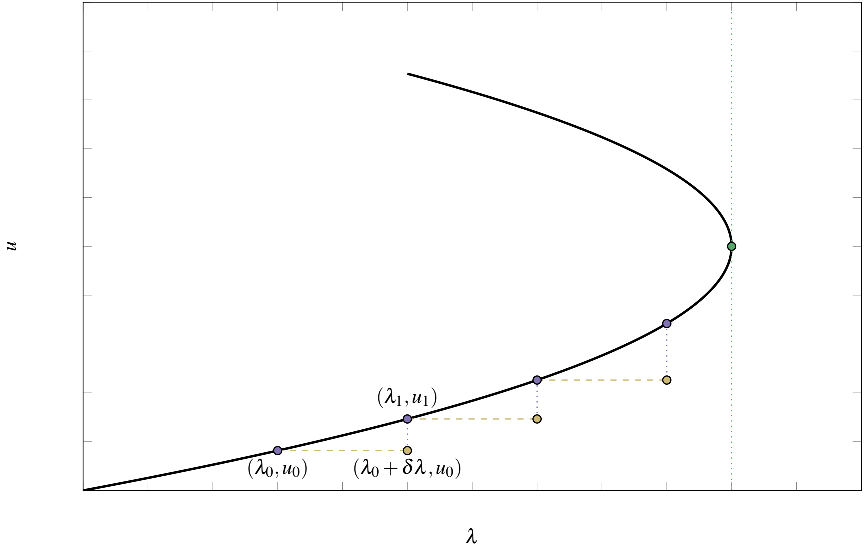
Use Newton–Kantorovich to find the solution for $\lambda_1 = \lambda_0 + \delta\lambda$.



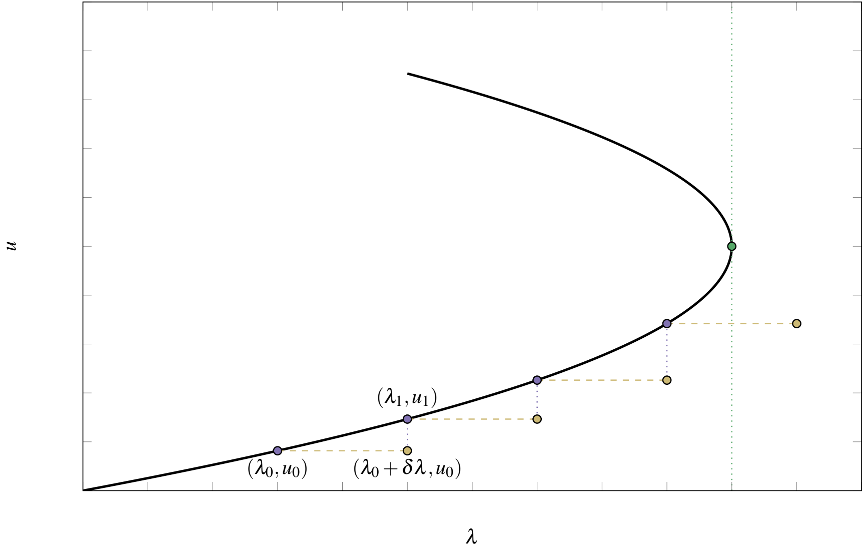
Piecewise-constant guess.



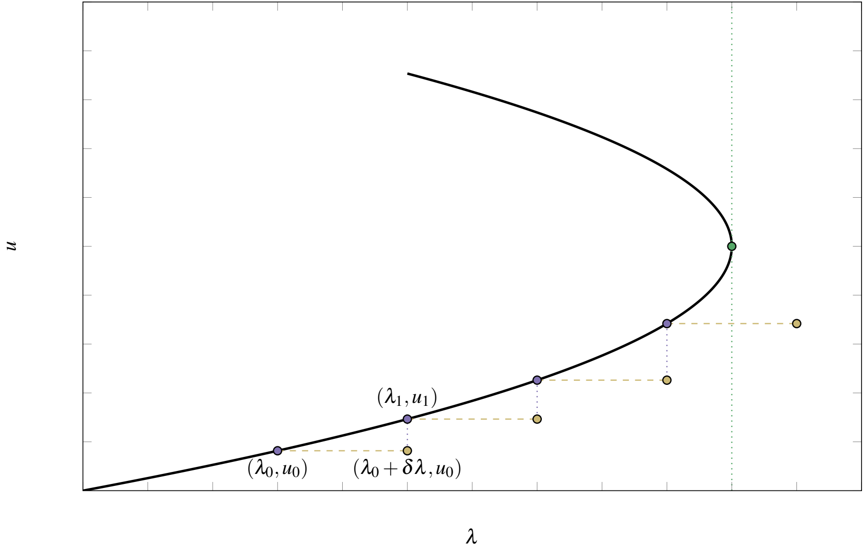
Newton–Kantorovich.



Guess and solve.



Guess . . .



...but there are no solutions to be found for this value of λ.

Good news

This is cheap and easy.

Good news

This is cheap and easy.

Bad news

We can probably construct better guesses.

Good news

This is cheap and easy.

Bad news

We can probably construct better guesses.

Worse news

The algorithm has no hope of continuing around the fold.

Subsection 2

Tangent and secant continuation

Natural continuation estimates

$$u(\lambda_{i+1}) \approx u(\lambda_i),$$

which is the first-order Taylor expansion.

Natural continuation estimates

$$u(\lambda_{i+1}) \approx u(\lambda_i),$$

which is the first-order Taylor expansion.

A better estimate would be

$$u(\lambda_{i+1}) \approx u(\lambda_i) + u_\lambda(\lambda_i)\delta\lambda,$$

the second-order Taylor expansion.

Natural continuation estimates

$$u(\lambda_{i+1}) \approx u(\lambda_i),$$

which is the first-order Taylor expansion.

A better estimate would be

$$u(\lambda_{i+1}) \approx u(\lambda_i) + u_\lambda(\lambda_i)\delta\lambda,$$

the second-order Taylor expansion.

How do we compute $u_\lambda(\lambda_i)$?

Since $F(u, \lambda) = 0$, taking the total derivative of both sides with respect to λ in the direction $\delta\lambda$ yields

$$\frac{\mathrm{d}}{\mathrm{d}\lambda} F(u, \lambda) = F_u(u, \lambda) u_\lambda + F_\lambda(u, \lambda) = 0.$$

Since $F(u, \lambda) = 0$, taking the total derivative of both sides with respect to λ in the direction $\delta\lambda$ yields

$$\frac{\mathrm{d}}{\mathrm{d}\lambda} F(u, \lambda) = F_u(u, \lambda) u_\lambda + F_\lambda(u, \lambda) = 0.$$

If $\lambda \in \mathbb{R}, u \in \mathbb{R}^N$, then $u_\lambda \in \mathbb{R}^N$, $F_\lambda \in \mathbb{R}^N$, and $F_u \in \mathbb{R}^{N \times N}$.

Since $F(u,\lambda) = 0$, taking the total derivative of both sides with respect to λ in the direction $\delta\lambda$ yields

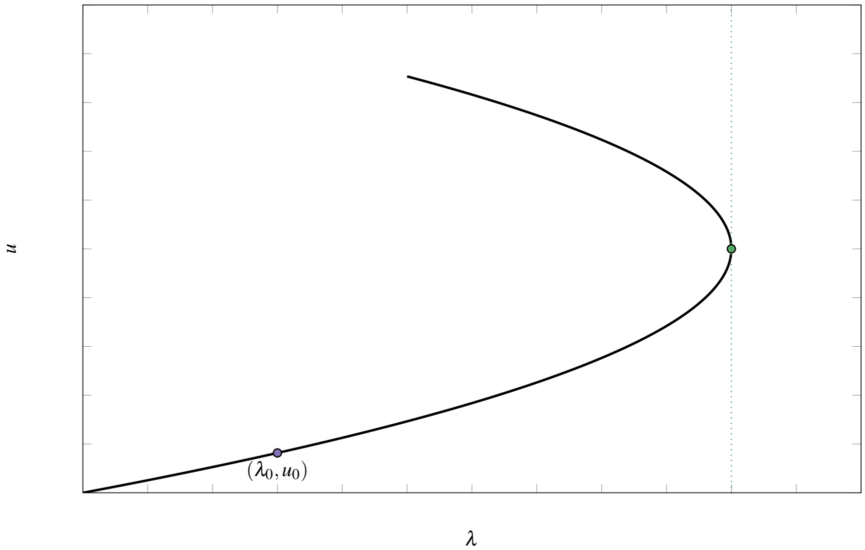
$$\frac{\mathrm{d}}{\mathrm{d}\lambda}F(u,\lambda) = F_u(u,\lambda)u_\lambda + F_\lambda(u,\lambda) = 0.$$

If $\lambda \in \mathbb{R}, u \in \mathbb{R}^N$, then $u_\lambda \in \mathbb{R}^N$, $F_\lambda \in \mathbb{R}^N$, and $F_u \in \mathbb{R}^{N\times N}$.

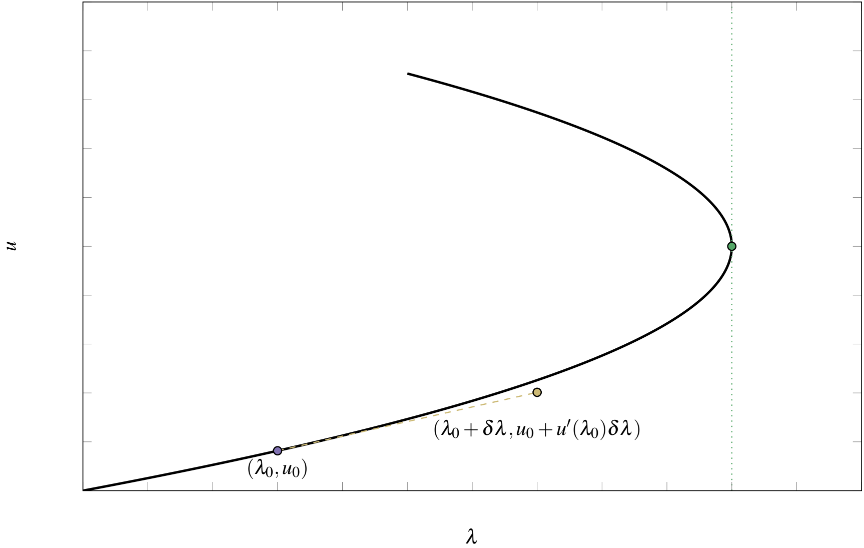
Since the dependence of F on λ is explicit, we can calculate F_λ, and solve

$$F_u(u,\lambda)u_\lambda = -F_\lambda(u,\lambda)$$

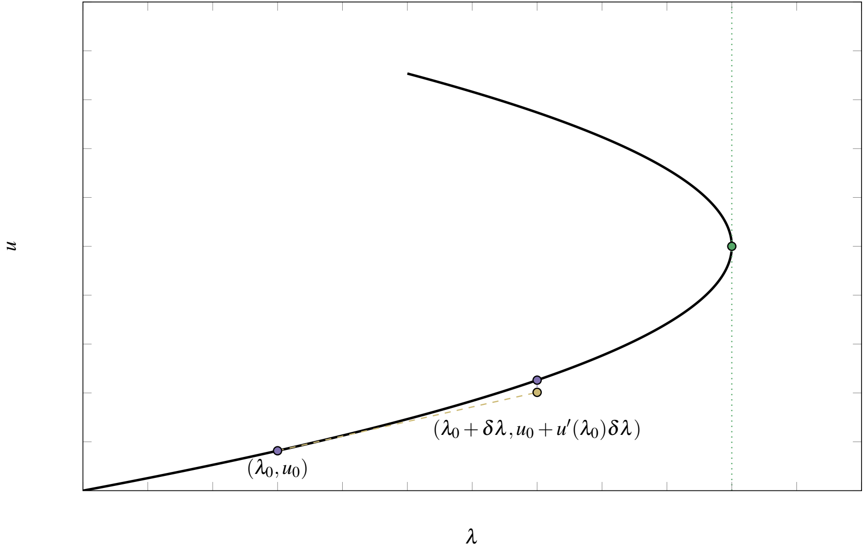
at the cost of one Newton step. This is the *tangent linearisation*.



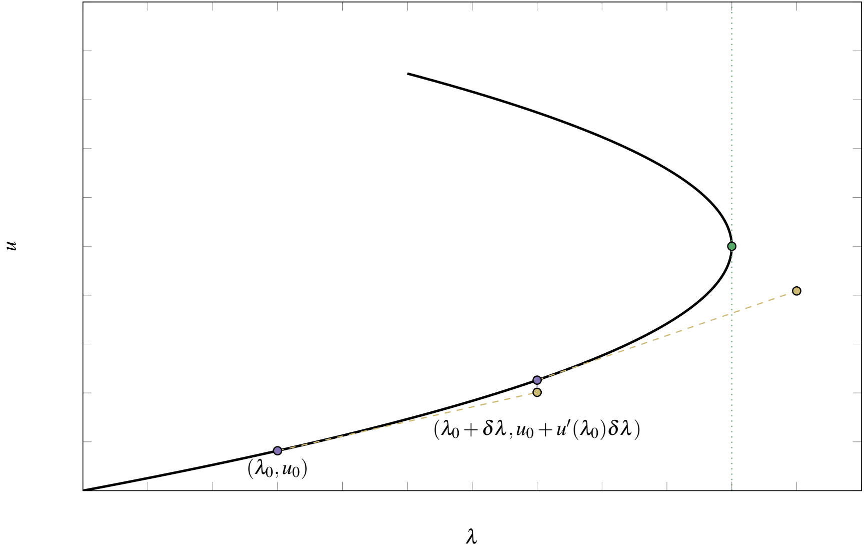
Start with (u_0, λ_0).



Solve tangent linearisation to construct next guess.



Solve nonlinear problem with Newton–Kantorovich.



Solve tangent linearisation to construct next guess.

This constructs much better initial guesses, but is more expensive. We have to save at least two Newton iterations to make this worth it.

This constructs much better initial guesses, but is more expensive. We have to save at least two Newton iterations to make this worth it.

A natural alternative is to approximate the tangent with a secant: build the line joining *two* previous points on the branch, and extrapolate to the next value of λ.

This constructs much better initial guesses, but is more expensive. We have to save at least two Newton iterations to make this worth it.

A natural alternative is to approximate the tangent with a secant: build the line joining *two* previous points on the branch, and extrapolate to the next value of λ.

Secant continuation constructs almost as good initial guesses, for almost no increase in cost over natural continuation (only memory).

Subsection 3

Arclength continuation

Tangent/secant continuation are much more efficient, but still have no hope of continuing around a fold.

Tangent/secant continuation are much more efficient, but still have no hope of continuing around a fold.

The fundamental problem is one of *parameterisation*: we are thinking of our solution curve as

$$u = u(\lambda)$$

but if we only ever increase λ, we cannot turn back around a fold.

Tangent/secant continuation are much more efficient, but still have no hope of continuing around a fold.

The fundamental problem is one of *parameterisation*: we are thinking of our solution curve as

$$u = u(\lambda)$$

but if we only ever increase λ, we cannot turn back around a fold.

A better way

Parameterise the solution curve as

$$(u(s), \lambda(s))$$

where s is the arclength on the curve, measured from (u_0, λ_0).

Tangent/secant continuation are much more efficient, but still have no hope of continuing around a fold.

The fundamental problem is one of *parameterisation*: we are thinking of our solution curve as

$$u = u(\lambda)$$

but if we only ever increase λ, we cannot turn back around a fold.

A better way

Parameterise the solution curve as

$$(u(s), \lambda(s))$$

where s is the arclength on the curve, measured from (u_0, λ_0).

In other words, *at each continuation step we will <u>also</u> solve for the next value of* λ*.* This allows λ to decrease as well as increase, to successfully traverse folds.

Since we are now solving for both u and λ, we need to augment our system of equations with one more real-valued equation:

$$A(u(s), \lambda(s)) := \begin{bmatrix} F(u(s), \lambda(s)) \\ p(u(s), \lambda(s), s) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

Since we are now solving for both u and λ, we need to augment our system of equations with one more real-valued equation:

$$A(u(s), \lambda(s)) := \begin{bmatrix} F(u(s), \lambda(s)) \\ p(u(s), \lambda(s), s) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

We can think of natural and tangent continuation in this framework by setting

$$p(u, \lambda, s) := \lambda - \lambda_{i+1}.$$

Since we are now solving for both u and λ, we need to augment our system of equations with one more real-valued equation:

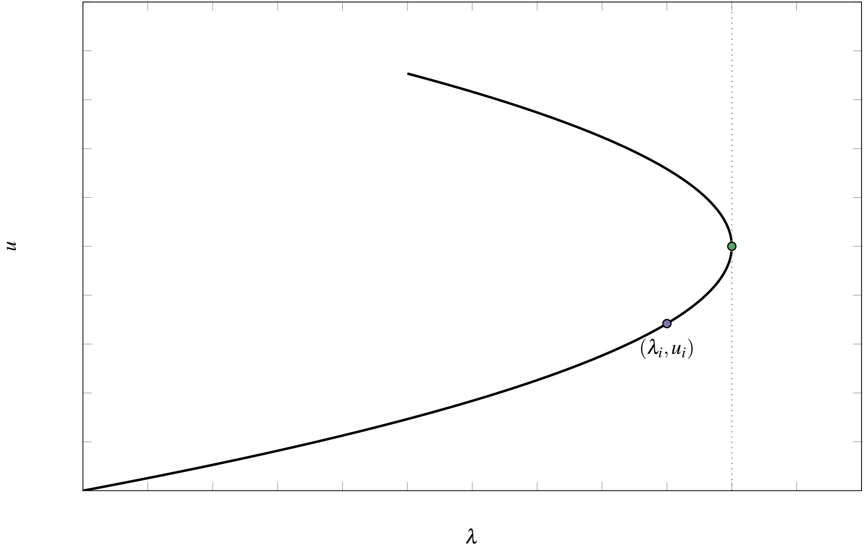
$$A(u(s), \lambda(s)) := \begin{bmatrix} F(u(s), \lambda(s)) \\ p(u(s), \lambda(s), s) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

We can think of natural and tangent continuation in this framework by setting

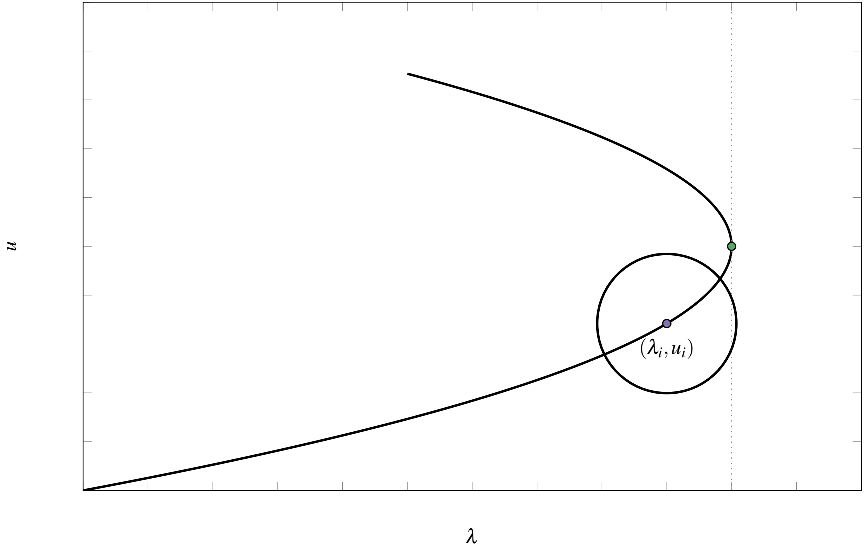
$$p(u, \lambda, s) := \lambda - \lambda_{i+1}.$$

The choice *arclength* continuation makes is to have p encode a desired change in distance:

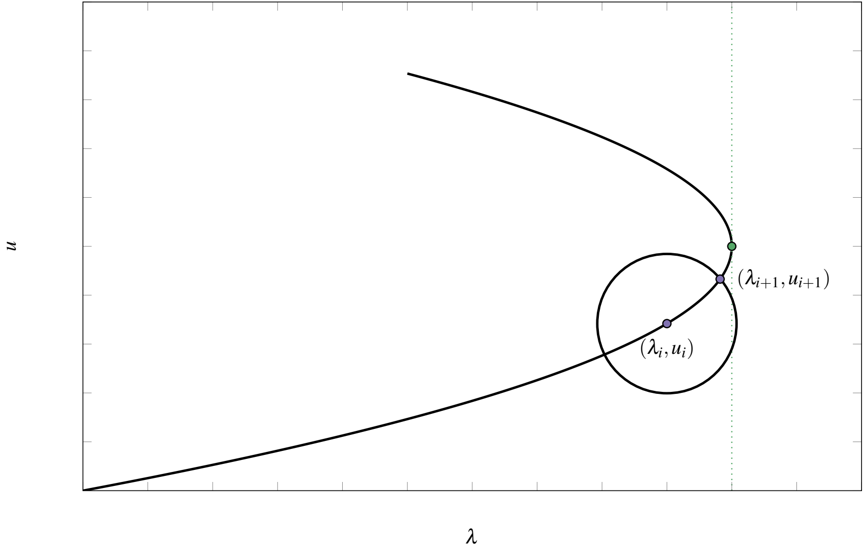
$$p(u, \lambda, s) := \|u - u_i\|^2 + |\lambda - \lambda_i|^2 - (s - s_i)^2.$$



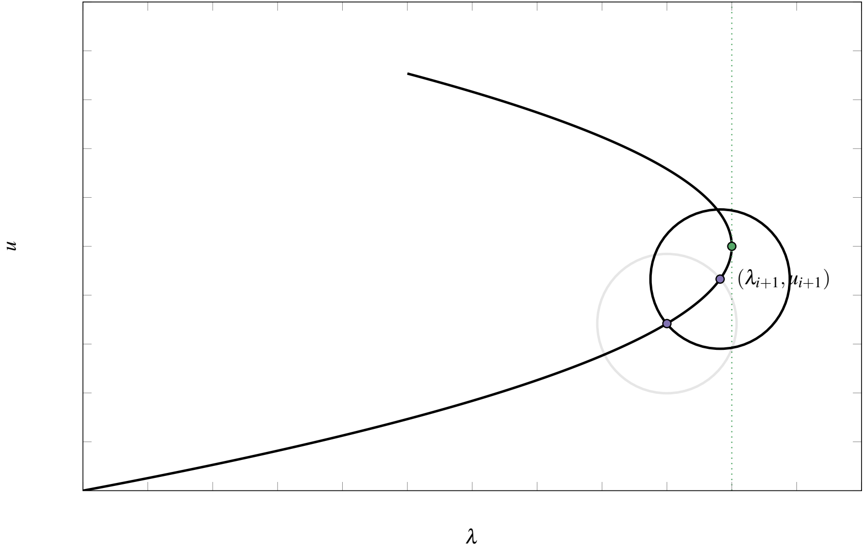
Start with (u_i, λ_i).



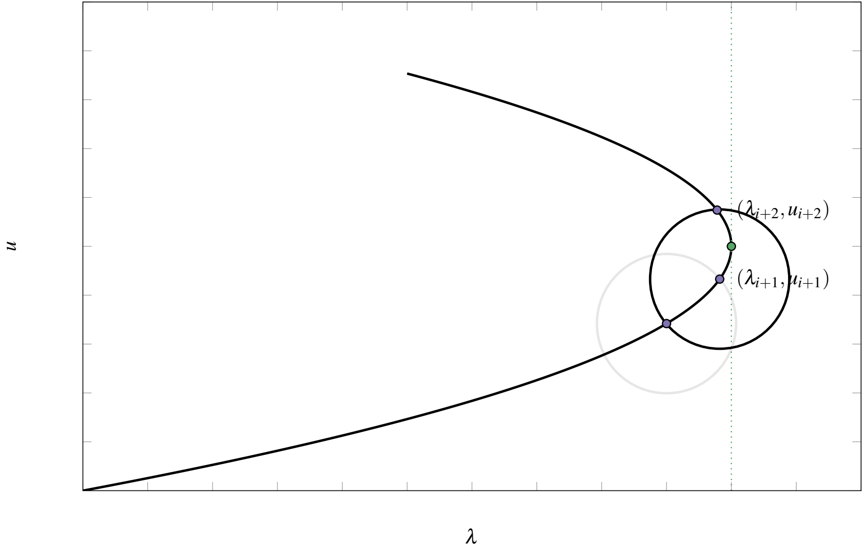
Seek points on the curve that intersect $p(u, \lambda) = 0$.



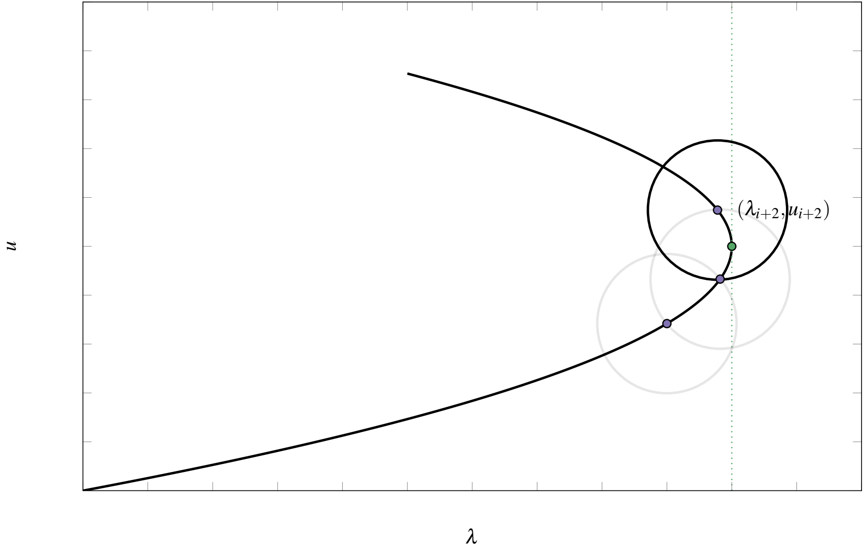
Solve nonlinear problem with Newton–Kantorovich.



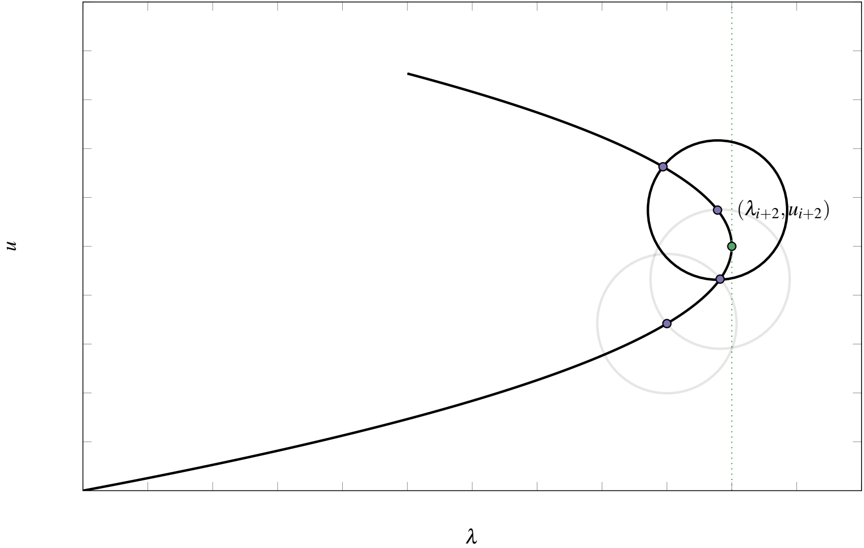
Repeat.



Repeat.



Repeat.



Repeat.

Good news

This allows us to robustly continue around folds.

Good news

This allows us to robustly continue around folds.

Bad news

We now have to solve augmented systems with extra nonlinearity.

Good news

This allows us to robustly continue around folds.

Bad news

We now have to solve augmented systems with extra nonlinearity.

Worse news

The augmented system generically has two solutions!

We attempt to guide Newton–Kantorovich to the solution we want by building a good initial guess.

We attempt to guide Newton–Kantorovich to the solution we want by building a good initial guess.

We compute $(u_s(s_i), \lambda_s(s_i))$ by solving

$$\frac{\mathrm{d}}{\mathrm{d}s} A(u(s), \lambda(s)) = 0,$$

the tangent linearisation of the augmented system.

We attempt to guide Newton–Kantorovich to the solution we want by building a good initial guess.

We compute $(u_s(s_i), \lambda_s(s_i))$ by solving

$$\frac{\mathrm{d}}{\mathrm{d}s} A(u(s), \lambda(s)) = 0,$$

the tangent linearisation of the augmented system.

We then set the initial guess to be $(u(s_i) + u_s(s_i)\delta s, \lambda(s_i) + \lambda_s(s_i)\delta s)$.

We attempt to guide Newton–Kantorovich to the solution we want by building a good initial guess.

We compute $(u_s(s_i), \lambda_s(s_i))$ by solving

$$\frac{\mathrm{d}}{\mathrm{d}s} A(u(s), \lambda(s)) = 0,$$

the tangent linearisation of the augmented system.

We then set the initial guess to be $(u(s_i) + u_s(s_i)\delta s, \lambda(s_i) + \lambda_s(s_i)\delta s)$.

However, this doesn't always work: even with this good initial guess, Newton–Kantorovich can sometimes find the wrong (old) solution.

The basic problem with arclength is that the extra equation added is nonlinear, and hence supports multiple solutions.

The basic problem with arclength is that the extra equation added is nonlinear, and hence supports multiple solutions.

We are free to choose the extra equation. So let's linearise it!

The basic problem with arclength is that the extra equation added is nonlinear, and hence supports multiple solutions.

We are free to choose the extra equation. So let's linearise it!

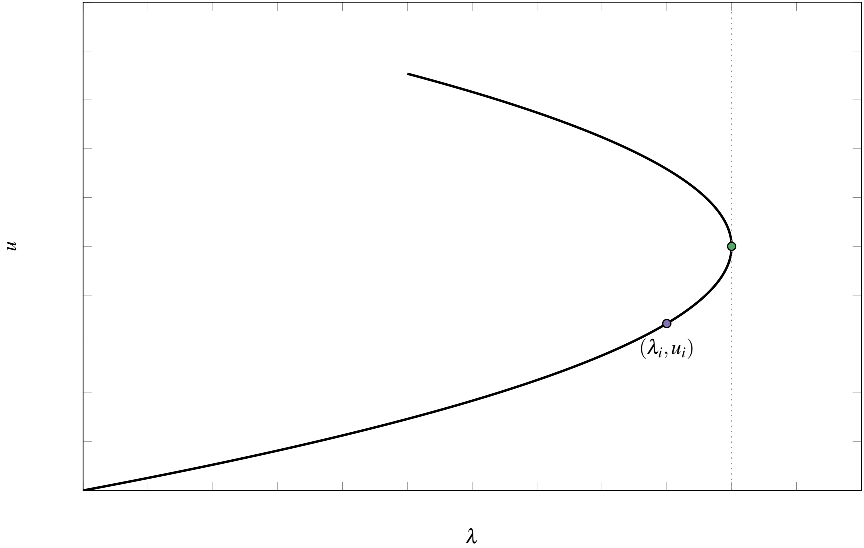
Pseudo-arclength continuation

Assuming that $X \subset L^2(\Omega)$, we can choose

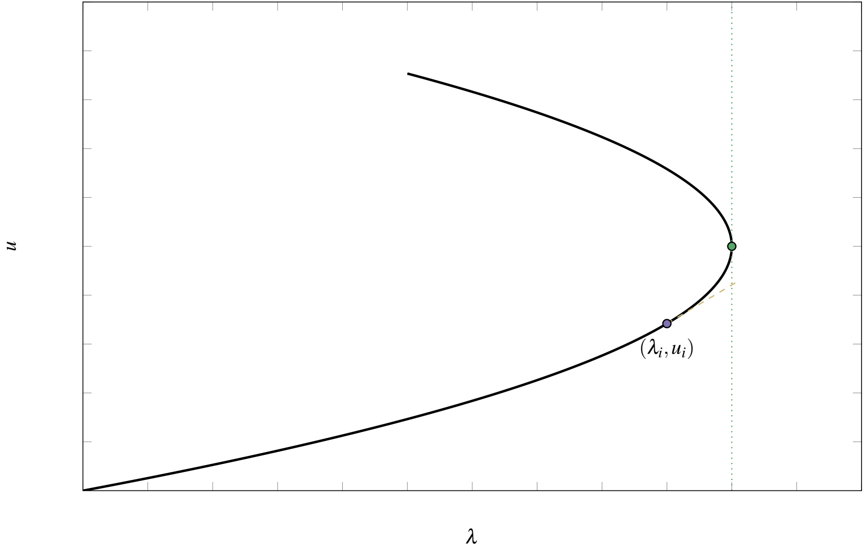
$$p(u, \lambda) := (u - u_i, u_s(s_i))_{L^2(\Omega)} + (\lambda - \lambda_i)\lambda_s(s_i) - (s - s_i)$$

This looks for points on the branch that are orthogonal (in the $L^2(\Omega) \times \mathbb{R}$ inner product) to the tangent, at a distance $s - s_i$ away.

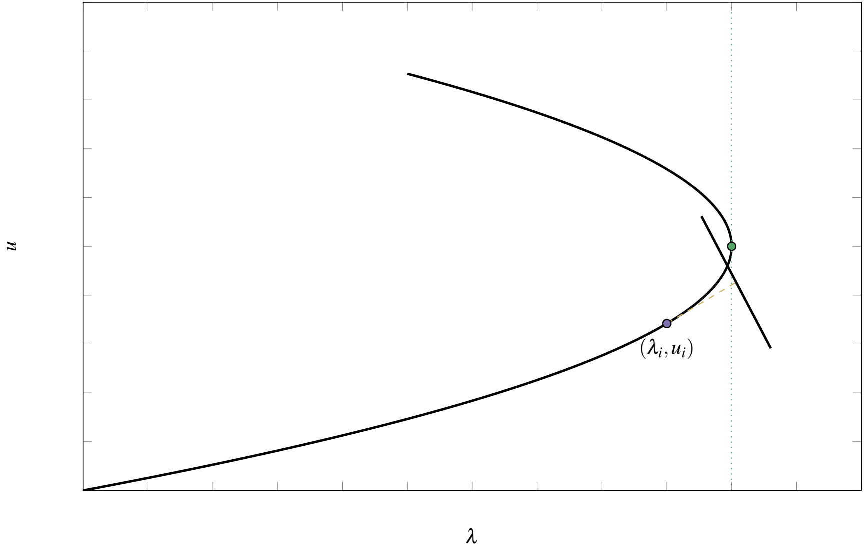
Eduard Riks, ?–?



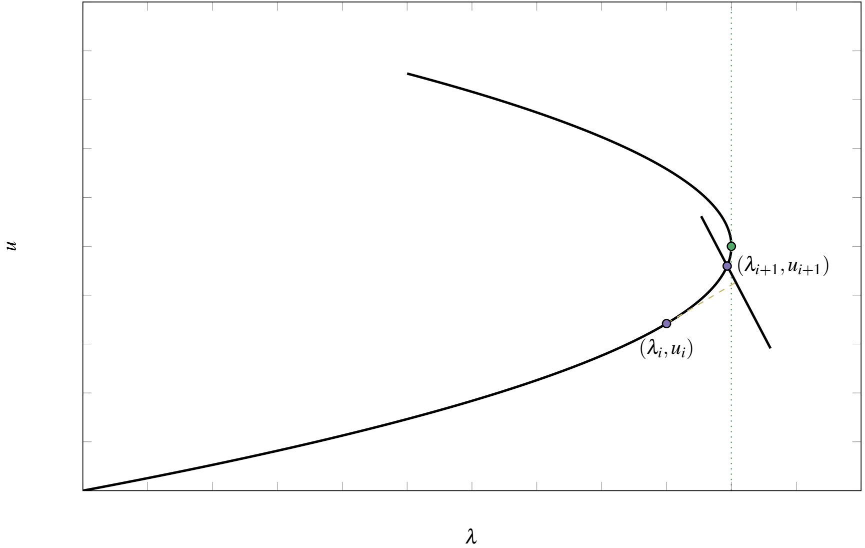
Start with (u_i, λ_i).



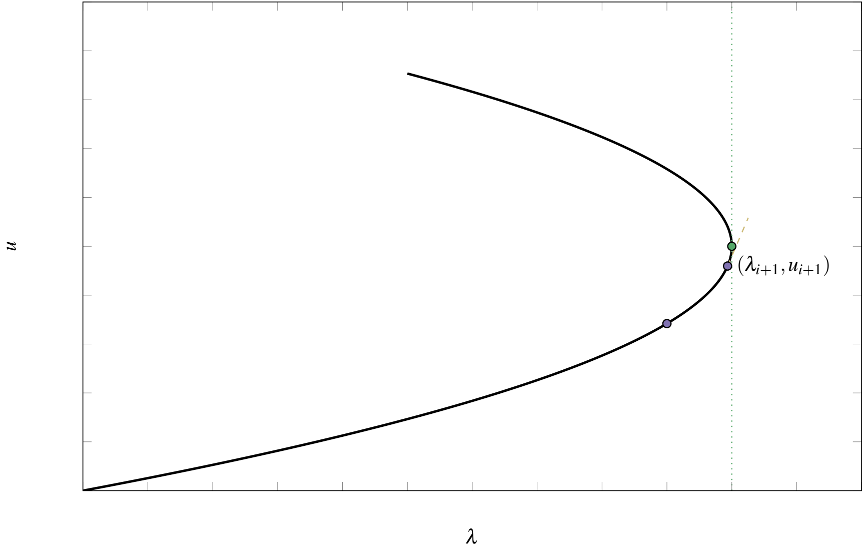
Construct the tangent to the curve.



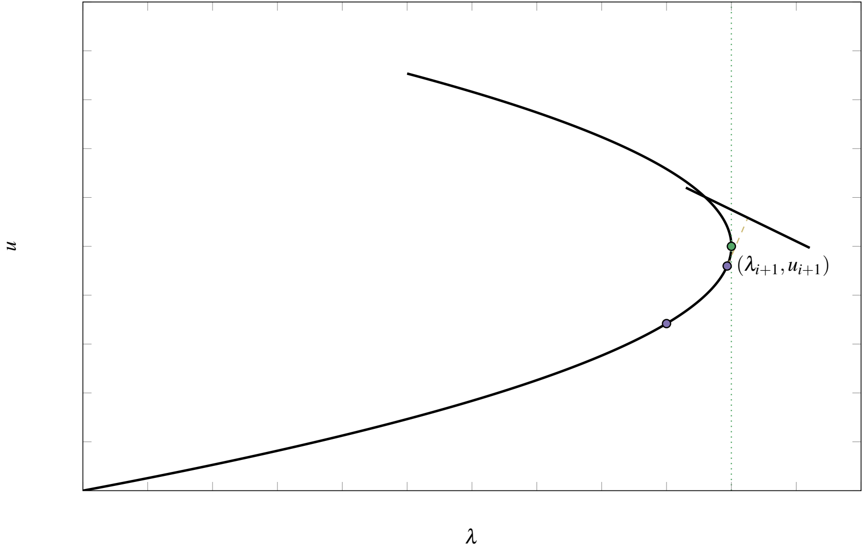
Impose the orthogonality constraint.



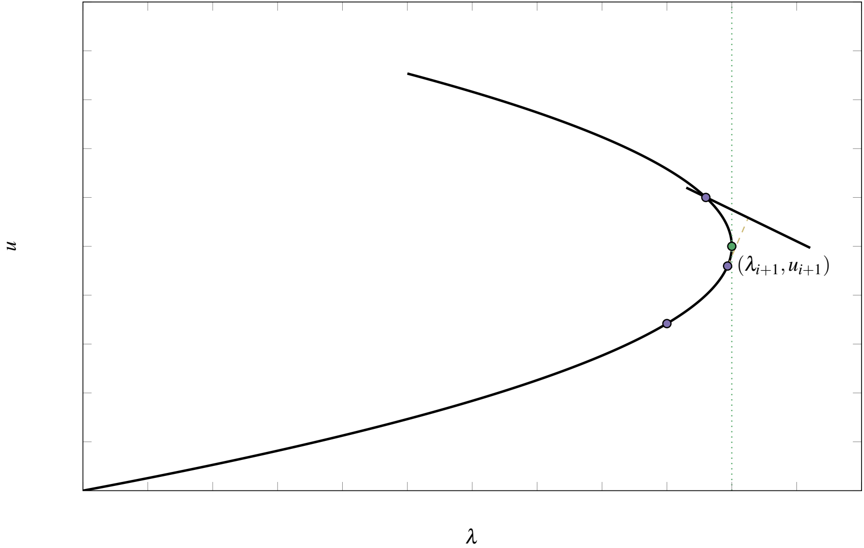
Solve.



Repeat.



Repeat.



Repeat.

Section 2

Bifurcation detection

Challenge

We need some way to detect that we have passed through a bifurcation.

Consider the problem

$$F(u, \lambda) = -u'' - \lambda u + u^3 = 0, \quad u(0) = 0 = u(\pi).$$

Consider the problem

$$F(u,\lambda) = -u'' - \lambda u + u^3 = 0, \quad u(0) = 0 = u(\pi).$$

This has a trivial branch of solutions $\{(0,\lambda) : \lambda \in \mathbb{R}\}$.

Consider the problem

$$F(u, \lambda) = -u'' - \lambda u + u^3 = 0, \quad u(0) = 0 = u(\pi).$$

This has a trivial branch of solutions $\{(0, \lambda) : \lambda \in \mathbb{R}\}$.

By the IFT, we know that bifurcations can only happen where its Fréchet derivative is singular. Its Fréchet derivative on the branch is

$$F_u(0, \lambda; v) = -v'' - \lambda v = 0, \quad v(0) = 0 = v(\pi),$$

Consider the problem

$$F(u, \lambda) = -u'' - \lambda u + u^3 = 0, \quad u(0) = 0 = u(\pi).$$

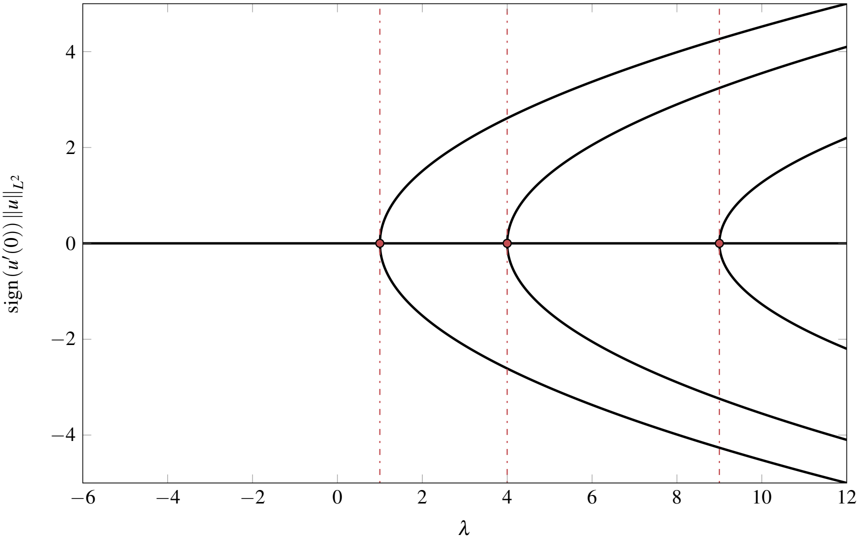
This has a trivial branch of solutions $\{(0, \lambda) : \lambda \in \mathbb{R}\}$.

By the IFT, we know that bifurcations can only happen where its Fréchet derivative is singular. Its Fréchet derivative on the branch is

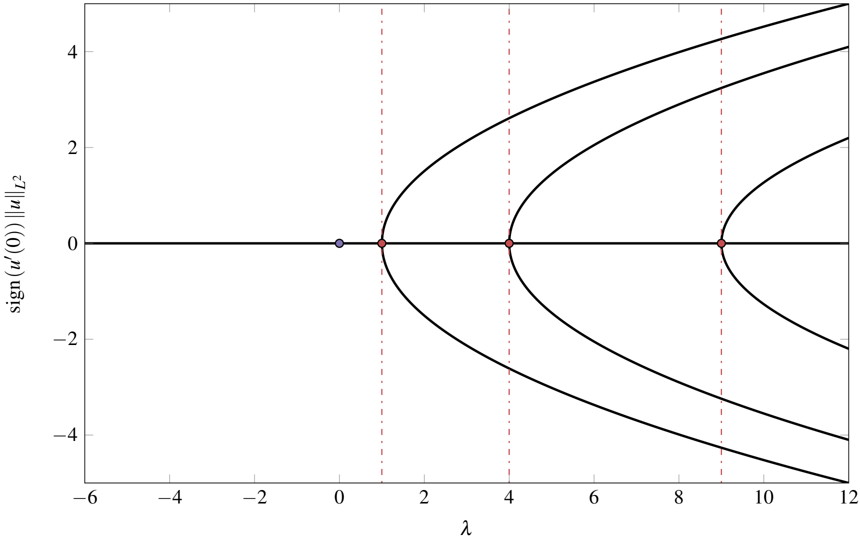
$$F_u(0, \lambda; v) = -v'' - \lambda v = 0, \quad v(0) = 0 = v(\pi),$$

which has nonzero solutions for v whenever λ is an eigenvalue of the Dirichlet Laplacian:

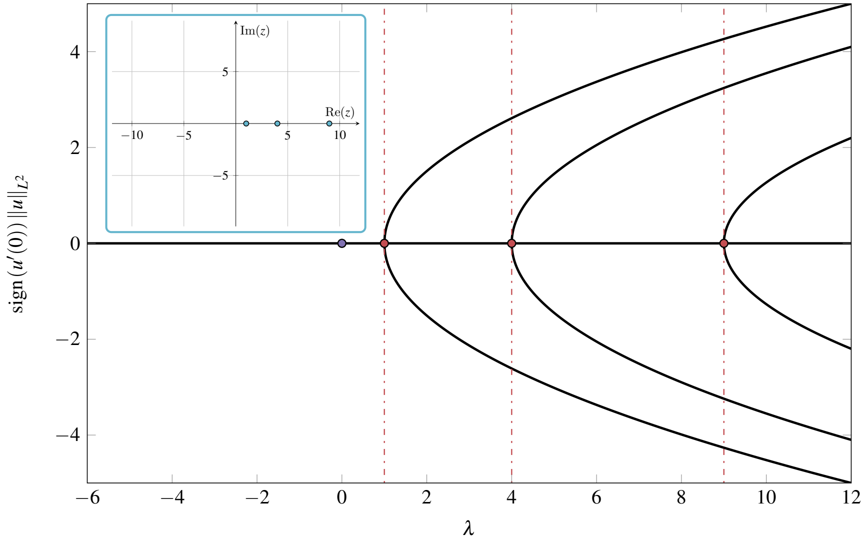
$$\lambda_n = n^2, \quad n \in \mathbb{N}.$$



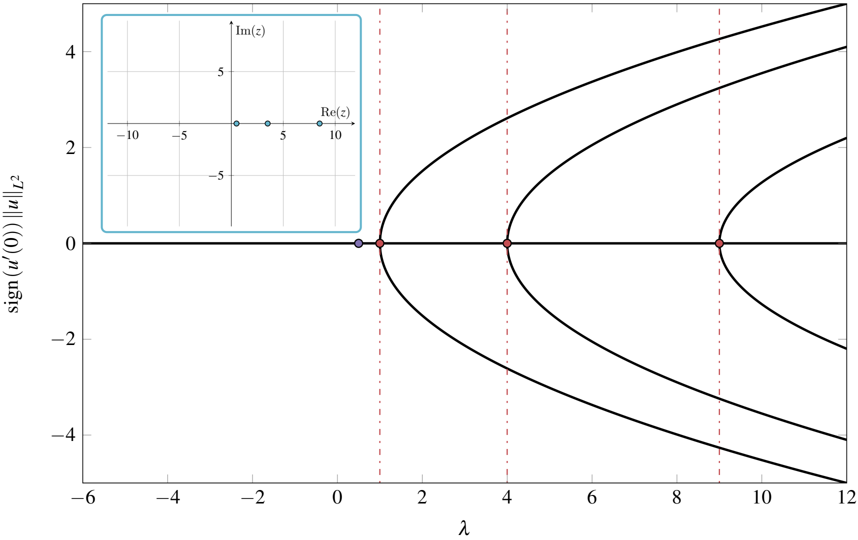
The bifurcation diagram we aim to compute.



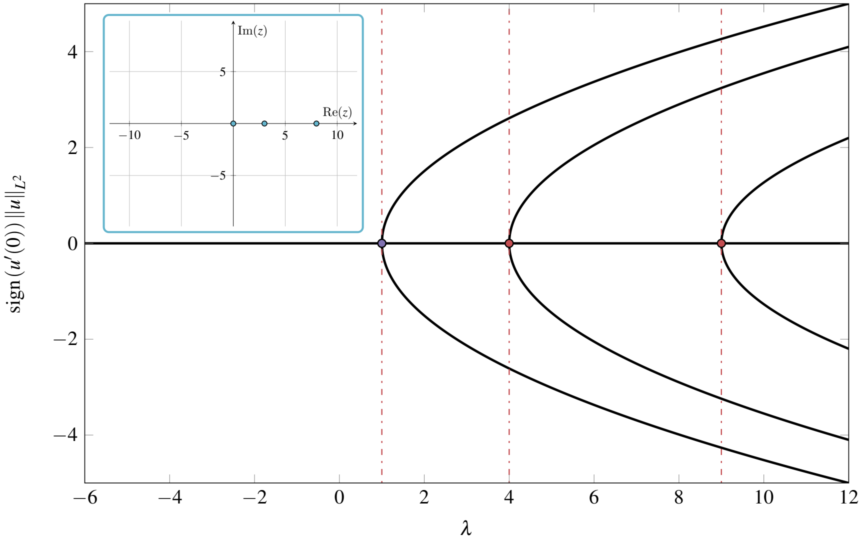
Start our continuation at $(u, \lambda) = (0, 0)$.



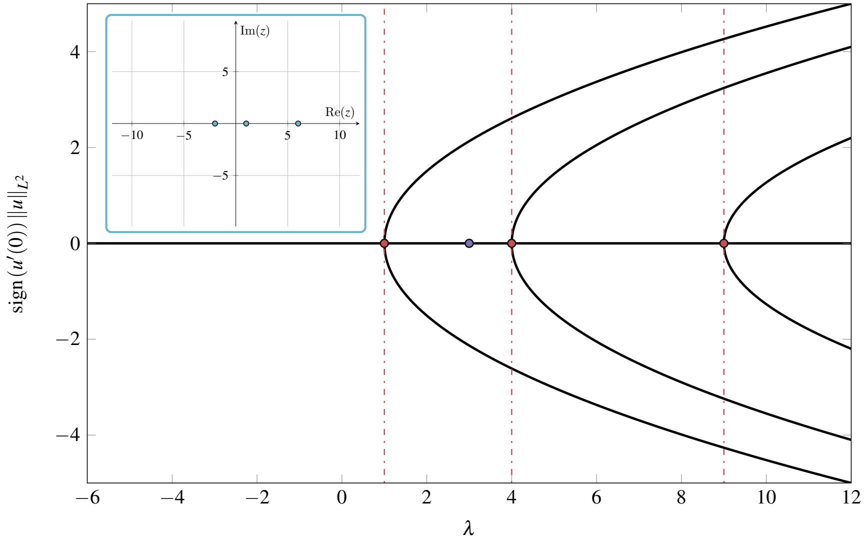
Examine the eigenvalues of F_u at this point.



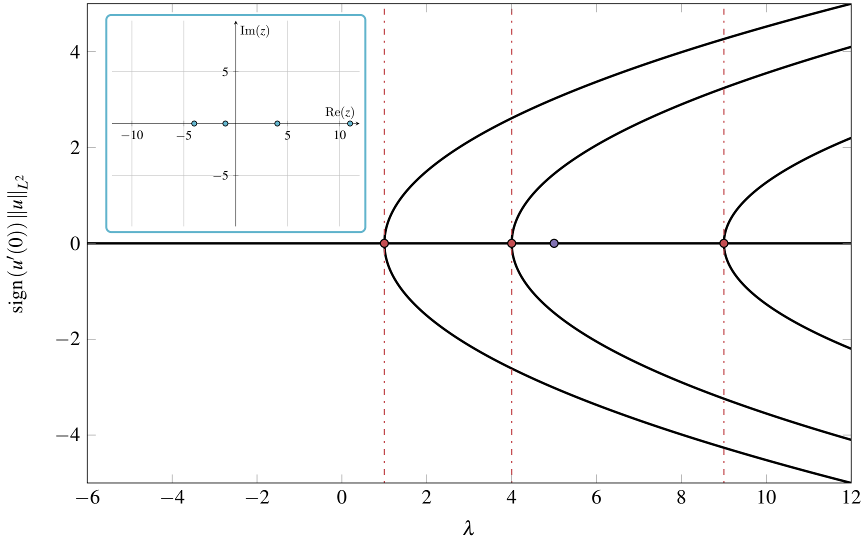
Take a continuation step.



By chance we land on the bifurcation—Fréchet derivative is singular.



Take another continuation step.



Take another continuation step, stepping over the next bifurcation.

So how do we detect when we've continued past a bifurcation?

So how do we detect when we've continued past a bifurcation?

Idea A

Monitor the *sign* of $\det\left(F_u(u,\lambda)\right)$.

So how do we detect when we've continued past a bifurcation?

Idea A

Monitor the *sign* of $\det\left(F_u(u, \lambda)\right)$.

Recall that the determinant of a matrix is the product of its eigenvalues. So when one eigenvalue changes sign, the determinant changes sign.

So how do we detect when we've continued past a bifurcation?

Idea A

Monitor the *sign* of $\det\left(F_u(u, \lambda)\right)$.

Recall that the determinant of a matrix is the product of its eigenvalues. So when one eigenvalue changes sign, the determinant changes sign.

Good news

The determinant is easy to compute from an LU factorisation:

$$A = LU \implies \det(A) = \det(L)\det(U).$$

Bad news

We usually can't afford to compute an LU factorisation . . .

Bad news

We usually can't afford to compute an LU factorisation . . .

Worse news

This misses bifurcations for eigenvalues of even multiplicity.

Bad news

We usually can't afford to compute an LU factorisation ...

Worse news

This misses bifurcations for eigenvalues of even multiplicity.

So we need another idea.

Idea B

At each continuation step, compute a few (e.g. 10) eigenvalues.

Idea B

At each continuation step, compute a few (e.g. 10) eigenvalues.

Good news

You can make this work at large scale with Krylov methods.

Idea B

At each continuation step, compute a few (e.g. 10) eigenvalues.

Good news

You can make this work at large scale with Krylov methods.

Challenge

You want the ones with smallest real part, somewhat fiddly.

Idea B

At each continuation step, compute a few (e.g. 10) eigenvalues.

Good news

You can make this work at large scale with Krylov methods.

Challenge

You want the ones with smallest real part, somewhat fiddly.

Comment

This is the main choice in PDE-oriented codes
(e.g. `pde2path` and `BifurcationKit.jl`).

Section 3

Bifurcation localisation

Our ultimate goal is to switch branches at bifurcation points. In order to do this, we'll need to locate the bifurcation points precisely.

Our ultimate goal is to switch branches at bifurcation points. In order to do this, we'll need to locate the bifurcation points precisely.

Idea A

Apply bisection to the detection algorithm.

In other words, you know two points on the branch that straddle the bifurcation. At each iteration, cut the interval between them in half and keep the subinterval that contains the bifurcation.

Our ultimate goal is to switch branches at bifurcation points. In order to do this, we'll need to locate the bifurcation points precisely.

Idea A

Apply bisection to the detection algorithm.

In other words, you know two points on the branch that straddle the bifurcation. At each iteration, cut the interval between them in half and keep the subinterval that contains the bifurcation.

Good news

This is simple to implement (given a detector).

Our ultimate goal is to switch branches at bifurcation points. In order to do this, we'll need to locate the bifurcation points precisely.

Idea A

Apply bisection to the detection algorithm.

In other words, you know two points on the branch that straddle the bifurcation. At each iteration, cut the interval between them in half and keep the subinterval that contains the bifurcation.

Good news

This is simple to implement (given a detector).

Bad news

This only converges linearly, so finding many digits will take forever.

Here is an idea that will let us quickly localise (some) bifurcations to high precision.

Here is an idea that will let us quickly localise (some) bifurcations to high precision.

By the IFT, we know that a *necessary* condition for a bifurcation is that

$$F_u(u, \lambda) \text{ is singular.}$$

Here is an idea that will let us quickly localise (some) bifurcations to high precision.

By the IFT, we know that a *necessary* condition for a bifurcation is that

$$F_u(u, \lambda) \text{ is singular.}$$

Idea B: Seydel–Moore–Spence

Find $(u, v, \lambda) \in X \times X \times \mathbb{R}$ such that

$$\begin{aligned} F(u, \lambda) &= 0, \\ F_u(u, \lambda)v &= 0, \\ \|v\|^2 &= 1. \end{aligned}$$

Rüdiger Seydel, 1947–

Gerald Moore, 1951–

Alistair Spence, 1948–

Comment

The Seydel–Moore–Spence system is highly nonlinear. However, it is easy to construct good initial guesses.

Comment

The Seydel–Moore–Spence system is highly nonlinear. However, it is easy to construct good initial guesses.

Good news

The Seydel–Moore–Spence system has nonsingular Fréchet derivative at a fold, so Newton–Kantorovich will converge quadratically.

Comment

The Seydel–Moore–Spence system is highly nonlinear. However, it is easy to construct good initial guesses.

Good news

The Seydel–Moore–Spence system has nonsingular Fréchet derivative at a fold, so Newton–Kantorovich will converge quadratically.

Bad news

The Fréchet derivative of the Seydel–Moore–Spence system is singular at other bifurcation points, so Newton–Kantorovich converges linearly.

Comment

The Seydel–Moore–Spence system is highly nonlinear. However, it is easy to construct good initial guesses.

Good news

The Seydel–Moore–Spence system has nonsingular Fréchet derivative at a fold, so Newton–Kantorovich will converge quadratically.

Bad news

The Fréchet derivative of the Seydel–Moore–Spence system is singular at other bifurcation points, so Newton–Kantorovich converges linearly.

Good news

It's possible to construct other augmented systems for other kinds of bifurcations. You have to know what you're looking for, though ...

One last comment: if you want to find out how a bifurcation point varies as you vary *another* parameter $\mu \in \mathbb{R}$, One last comment: if you want to find out how a bifurcation point varies as you vary *another* parameter $\mu \in \mathbb{R}$,

do pseudo-arclength continuation on the Seydel–Moore–Spence system!

Section 4

Branch switching

To learn how to switch branches at a bifurcation point, we need another

Great Theorem of Nonlinear Functional Analysis.

To learn how to switch branches at a bifurcation point, we need another

Great Theorem of Nonlinear Functional Analysis.

Lyapunov–Schmidt reduction (1906, 1908)

Let $F(u_0, \lambda_0) = 0$ with F_u singular. Let

$$d = \dim \ker F_u(u_0, \lambda_0).$$

Aleksandr Lyapunov, 1857–1918

Erhard Schmidt, 1876–1959

To learn how to switch branches at a bifurcation point, we need another

Great Theorem of Nonlinear Functional Analysis.

Lyapunov–Schmidt reduction (1906, 1908)

Let $F(u_0, \lambda_0) = 0$ with F_u singular. Let

$$d = \dim \ker F_u(u_0, \lambda_0).$$

Near the bifurcation point, we can relate

$$\text{solutions of } F \iff \text{solutions of } R$$

where R is a $d \times d$ algebraic system!

Aleksandr Lyapunov, 1857–1918

Erhard Schmidt, 1876–1959

For this section, we will make the following assumptions:

Essential assumptions

- $F(u_0, \lambda_0) = 0$;
- $A := F_u(u_0, \lambda_0) \in L(X, Y)$ is *Fredholm*:

$$\dim \ker(A) < \infty, \quad \operatorname{codim} \operatorname{range}(A) < \infty;$$

- $d = \dim \ker(A) > 0$.

For this section, we will make the following assumptions:

Essential assumptions

- $F(u_0, \lambda_0) = 0$;
- $A := F_u(u_0, \lambda_0) \in L(X, Y)$ is *Fredholm*:

$$\dim \ker(A) < \infty, \quad \operatorname{codim} \operatorname{range}(A) < \infty;$$

- $d = \dim \ker(A) > 0$.

Non-essential assumptions

- X and Y are Hilbert spaces;
- $\operatorname{ind}(A) := \dim \ker(A) - \operatorname{codim} \operatorname{range}(A) = 0$.

Let $A^* : Y \to X$ be the associated adjoint operator. Construct

$$\ker(A) = \mathrm{span}\{\phi_1, \dots, \phi_d\}, \quad \ker(A^*) = \mathrm{span}\{\psi_1, \dots, \psi_d\},$$

where $\{\phi_i\}_i$ and $\{\psi_i\}_i$ are orthonormal bases.

Let $A^* : Y \to X$ be the associated adjoint operator. Construct

$$\ker(A) = \operatorname{span}\{\phi_1, \ldots, \phi_d\}, \quad \ker(A^*) = \operatorname{span}\{\psi_1, \ldots, \psi_d\},$$

where $\{\phi_i\}_i$ and $\{\psi_i\}_i$ are orthonormal bases.

Then construct

$$Px \coloneqq \sum_{i=1}^{d} (\phi_i, x)_X \phi_i,$$

Let $A^* : Y \to X$ be the associated adjoint operator. Construct

$$\ker(A) = \operatorname{span}\{\phi_1, \dots, \phi_d\}, \quad \ker(A^*) = \operatorname{span}\{\psi_1, \dots, \psi_d\},$$

where $\{\phi_i\}_i$ and $\{\psi_i\}_i$ are orthonormal bases.

Then construct

$$Px \coloneqq \sum_{i=1}^{d} (\phi_i, x)_X \phi_i, \quad Qy \coloneqq \sum_{i=1}^{d} (\psi_i, y)_Y \psi_i.$$

Let $A^* : Y \to X$ be the associated adjoint operator. Construct

$$\ker(A) = \mathrm{span}\{\phi_1, \ldots, \phi_d\}, \quad \ker(A^*) = \mathrm{span}\{\psi_1, \ldots, \psi_d\},$$

where $\{\phi_i\}_i$ and $\{\psi_i\}_i$ are orthonormal bases.

Then construct

$$Px \coloneqq \sum_{i=1}^{d} (\phi_i, x)_X \phi_i, \quad Qy \coloneqq \sum_{i=1}^{d} (\psi_i, y)_Y \psi_i.$$

By construction,

$$\mathrm{range}(P) = \ker(A),$$

Let $A^* : Y \to X$ be the associated adjoint operator. Construct

$$\ker(A) = \mathrm{span}\{\phi_1, \ldots, \phi_d\}, \quad \ker(A^*) = \mathrm{span}\{\psi_1, \ldots, \psi_d\},$$

where $\{\phi_i\}_i$ and $\{\psi_i\}_i$ are orthonormal bases.

Then construct

$$Px \coloneqq \sum_{i=1}^{d} (\phi_i, x)_X \phi_i, \quad Qy \coloneqq \sum_{i=1}^{d} (\psi_i, y)_Y \psi_i.$$

By construction,

$$\mathrm{range}(P) = \ker(A), \quad \mathrm{range}(Q) = \ker(A^*)$$

Let $A^* : Y \to X$ be the associated adjoint operator. Construct

$$\ker(A) = \operatorname{span}\{\phi_1, \ldots, \phi_d\}, \quad \ker(A^*) = \operatorname{span}\{\psi_1, \ldots, \psi_d\},$$

where $\{\phi_i\}_i$ and $\{\psi_i\}_i$ are orthonormal bases.

Then construct

$$Px \coloneqq \sum_{i=1}^{d} (\phi_i, x)_X \phi_i, \quad Qy \coloneqq \sum_{i=1}^{d} (\psi_i, y)_Y \psi_i.$$

By construction,

$$\operatorname{range}(P) = \ker(A), \quad \operatorname{range}(Q) = \ker(A^*) = \operatorname{range}(A)^{\perp}.$$

Let $A^* : Y \to X$ be the associated adjoint operator. Construct

$$\ker(A) = \mathrm{span}\{\phi_1, \dots, \phi_d\}, \quad \ker(A^*) = \mathrm{span}\{\psi_1, \dots, \psi_d\},$$

where $\{\phi_i\}_i$ and $\{\psi_i\}_i$ are orthonormal bases.

Then construct

$$Px \coloneqq \sum_{i=1}^{d} (\phi_i, x)_X \phi_i, \quad Qy \coloneqq \sum_{i=1}^{d} (\psi_i, y)_Y \psi_i.$$

By construction,

$$\mathrm{range}(P) = \ker(A), \quad \mathrm{range}(Q) = \ker(A^*) = \mathrm{range}(A)^\perp.$$

Then we can decompose

$$X = \mathrm{range}(P) \oplus \mathrm{range}(I - P) \eqqcolon X_1 \oplus X_2,$$
$$Y = \mathrm{range}(Q) \oplus \mathrm{range}(I - Q) \eqqcolon Y_1 \oplus Y_2.$$

Write

$$u = Pu + (I - P)u =: v + w, \quad v \in X_1, w \in X_2.$$

Write

$$u = Pu + (I - P)u =: v + w, \quad v \in X_1, w \in X_2.$$

Then the system $F(u, \lambda) = 0$ is equivalent to

$$\begin{aligned}
\hat{F}(v, w, \lambda) &:= QF(v + w, \lambda) &&= 0 \in Y_1, \\
\bar{F}(v, w, \lambda) &:= (I - Q)F(v + w, \lambda) &&= 0 \in Y_2.
\end{aligned}$$

Write

$$u = Pu + (I - P)u =: v + w, \quad v \in X_1, w \in X_2.$$

Then the system $F(u, \lambda) = 0$ is equivalent to

$$\begin{aligned}
\hat{F}(v, w, \lambda) &:= QF(v + w, \lambda) &&= 0 \in Y_1, \\
\bar{F}(v, w, \lambda) &:= (I - Q)F(v + w, \lambda) &&= 0 \in Y_2.
\end{aligned}$$

The Fréchet derivative $\bar{F}_w$ is the restriction of A to

$$A : \ker(A)^{\perp} \to \mathrm{range}(A)$$

and is thus invertible.

Write

$$u = Pu + (I - P)u =: v + w, \quad v \in X_1, w \in X_2.$$

Then the system $F(u, \lambda) = 0$ is equivalent to

$$\begin{aligned} \hat{F}(v, w, \lambda) &:= QF(v + w, \lambda) &&= 0 \in Y_1, \\ \bar{F}(v, w, \lambda) &:= (I - Q)F(v + w, \lambda) &&= 0 \in Y_2. \end{aligned}$$

The Fréchet derivative $\bar{F}_w$ is the restriction of A to

$$A : \ker(A)^\perp \to \mathrm{range}(A)$$

and is thus invertible. So by the IFT we can locally write

$$w = H(v, \lambda).$$

We can thus write our reduced system

Reduced system

$$R(v,\lambda) := QF(v + H(v,\lambda),\lambda) = 0,$$
$$R : \ker(A) \times \mathbb{R} \to \mathrm{range}(A)^{\perp}.$$

We can thus write our reduced system

Reduced system

$$R(v, \lambda) := QF(v + H(v, \lambda), \lambda) = 0,$$
$$R : \ker(A) \times \mathbb{R} \to \mathrm{range}(A)^{\perp}.$$

This reduced system has the *same symmetries* and *same bifurcations* as the original problem, near (u_0, λ_0).

We can thus write our reduced system

Reduced system

$$R(v, \lambda) := QF(v + H(v, \lambda), \lambda) = 0,$$
$$R : \ker(A) \times \mathbb{R} \to \mathrm{range}(A)^{\perp}.$$

This reduced system has the *same symmetries* and *same bifurcations* as the original problem, near (u_0, λ_0).

This is an extremely useful theoretical result. It forms the basis of most analytical calculations of bifurcation structures.

Using our bases for $\ker(A)$ and $\mathrm{range}(A)^\perp$, let's explicitly write:

Reduced system (algebraic)

$$r_j(x,\lambda) := \big(\psi_j, R(x_1\phi_1 + \cdots + x_d\phi_d, \lambda)\big)_Y,$$
$$r : \mathbb{R}^d \times \mathbb{R} \to \mathbb{R}^d.$$

Using our bases for $\ker(A)$ and $\mathrm{range}(A)^{\perp}$, let's explicitly write:

Reduced system (algebraic)

$$r_j(x, \lambda) := \big(\psi_j, R(x_1\phi_1 + \cdots + x_d\phi_d, \lambda)\big)_Y,$$
$$r : \mathbb{R}^d \times \mathbb{R} \to \mathbb{R}^d.$$

In practice we can never get our hands on r, because we don't know H.

Using our bases for $\ker(A)$ and $\mathrm{range}(A)^\perp$, let's explicitly write:

Reduced system (algebraic)

$$r_j(x, \lambda) := \big(\psi_j, R(x_1\phi_1 + \cdots + x_d\phi_d, \lambda)\big)_Y,$$
$$r : \mathbb{R}^d \times \mathbb{R} \to \mathbb{R}^d.$$

In practice we can never get our hands on r, because we don't know H.

Instead, we compute a Taylor expansion (usually to third derivatives) of r.

Using our bases for $\ker(A)$ and $\mathrm{range}(A)^{\perp}$, let's explicitly write:

Reduced system (algebraic)

$$r_j(x, \lambda) \coloneqq \left(\psi_j, R(x_1\phi_1 + \cdots + x_d\phi_d, \lambda)\right)_Y,$$
$$r : \mathbb{R}^d \times \mathbb{R} \to \mathbb{R}^d.$$

In practice we can never get our hands on r, because we don't know H.

Instead, we compute a Taylor expansion (usually to third derivatives) of r.

The derivatives of r can be computed from derivatives of F, and require solving linear systems involving A ($d^2 + 1$ solves for third derivatives).

Challenge

For large d, the Taylor expansion of the reduced equations are not easy to solve. There are techniques from numerical algebraic geometry that can provably yield all solutions, but they are too slow to use in practice.

Challenge

For large d, the Taylor expansion of the reduced equations are not easy to solve. There are techniques from numerical algebraic geometry that can provably yield all solutions, but they are too slow to use in practice.

The pragmatic response taken is to brute-force the system with many, many initial guesses (e.g. as in `pde2path` and `BifurcationKit.jl`).

Example: Liouville–Bratu–Gelfand problem in 2D

$$\nabla^2 u - 10(u - \lambda e^u) = 0 \text{ on } \Omega := (0,1)^2, \quad \nabla u \cdot n = 0 \text{ on } \partial\Omega.$$

Example: Liouville–Bratu–Gelfand problem in 2D

$$\nabla^2 u - 10(u - \lambda e^u) = 0 \text{ on } \Omega := (0,1)^2, \quad \nabla u \cdot n = 0 \text{ on } \partial\Omega.$$

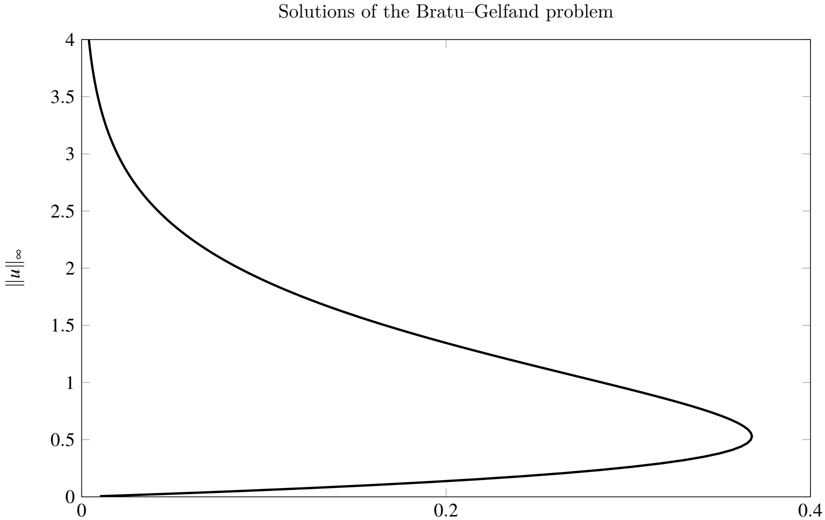
This is a famously intricate problem. I calculated the bifurcation diagram using `BifurcationKit.jl`. It was first computed successfully by Michiel Wouters.

Romain Veltz, 1982–

Michiel Wouters, ?–

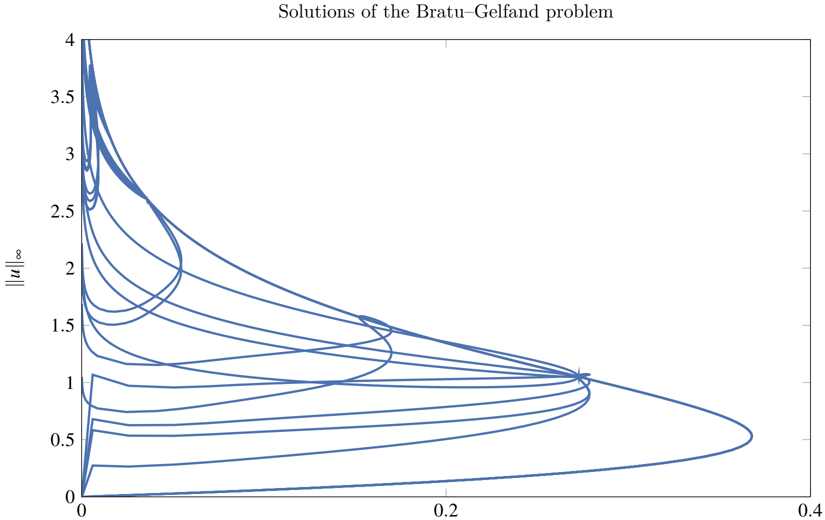
Example: Liouville–Bratu–Gelfand problem in 2D

$$\nabla^2 u - 10(u - \lambda e^u) = 0 \text{ on } \Omega := (0,1)^2, \quad \nabla u \cdot n = 0 \text{ on } \partial\Omega.$$



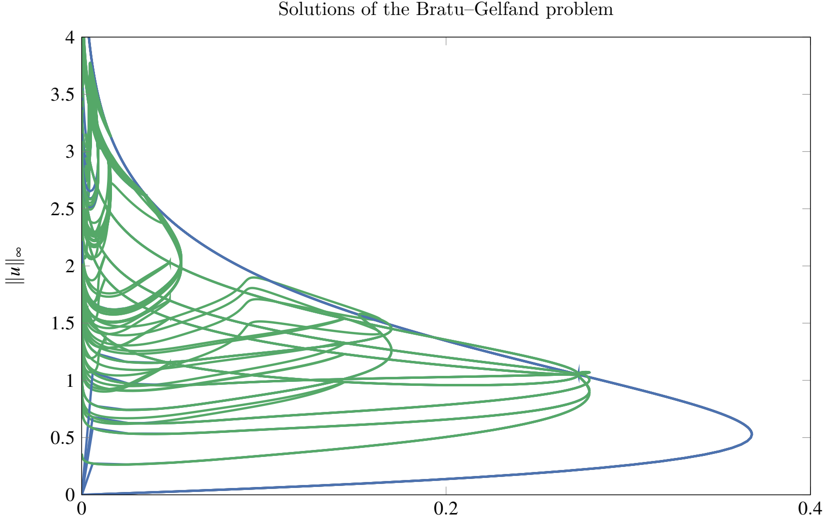
Example: Liouville–Bratu–Gelfand problem in 2D

$$\nabla^2 u - 10(u - \lambda e^u) = 0 \text{ on } \Omega := (0,1)^2, \quad \nabla u \cdot n = 0 \text{ on } \partial\Omega.$$



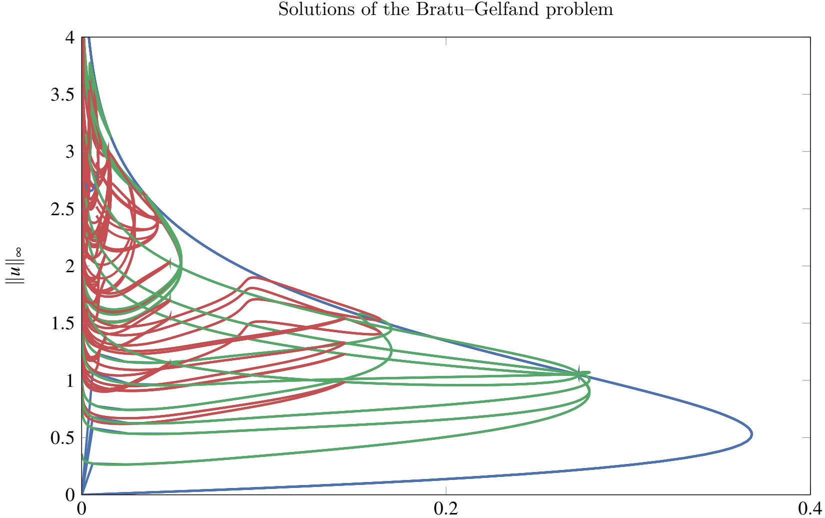
Example: Liouville–Bratu–Gelfand problem in 2D

$$\nabla^2 u - 10(u - \lambda e^u) = 0 \text{ on } \Omega := (0,1)^2, \quad \nabla u \cdot n = 0 \text{ on } \partial\Omega.$$



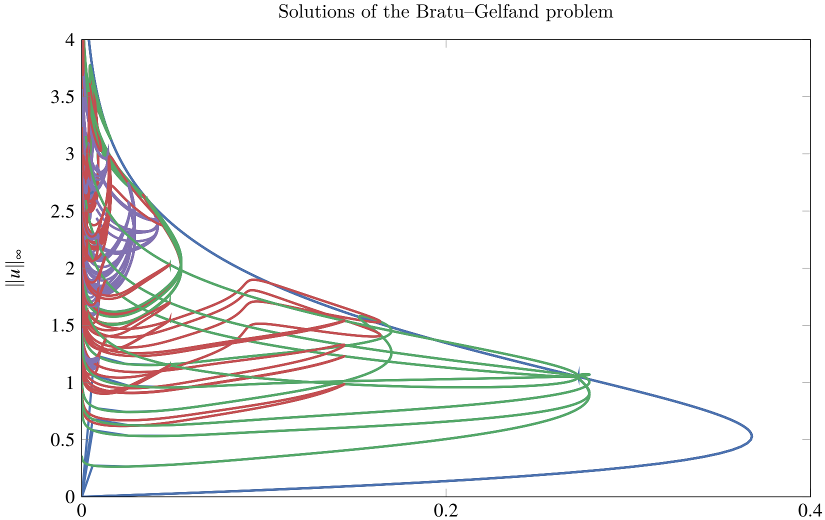
Example: Liouville–Bratu–Gelfand problem in 2D

$$\nabla^2 u - 10(u - \lambda e^u) = 0 \text{ on } \Omega := (0,1)^2, \quad \nabla u \cdot n = 0 \text{ on } \partial\Omega.$$



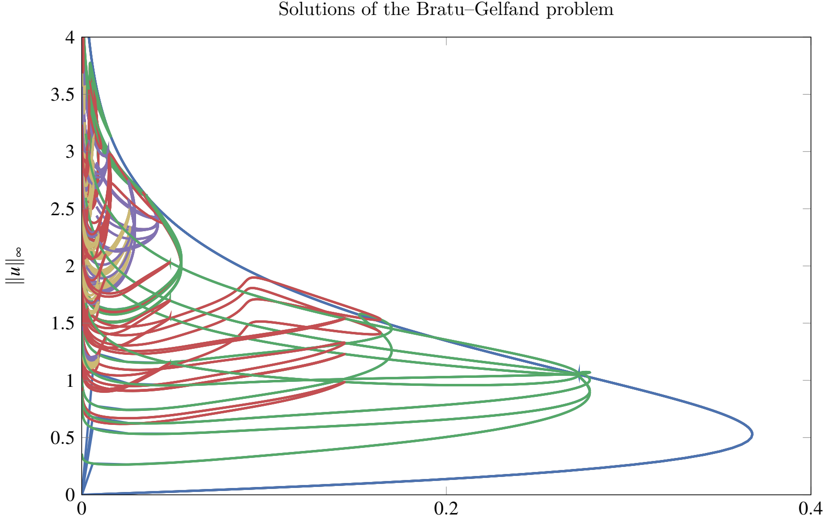
Example: Liouville–Bratu–Gelfand problem in 2D

$$\nabla^2 u - 10(u - \lambda e^u) = 0 \text{ on } \Omega := (0,1)^2, \quad \nabla u \cdot n = 0 \text{ on } \partial\Omega.$$



Example: Liouville–Bratu–Gelfand problem in 2D

$$\nabla^2 u - 10(u - \lambda e^u) = 0 \text{ on } \Omega := (0,1)^2, \quad \nabla u \cdot n = 0 \text{ on } \partial\Omega.$$



Lecture 3: Deflation algorithms for bifurcation analysis

Patrick E. Farrell

University of Oxford

June 1

Good news

The combination of continuation and branch switching is very powerful.

Good news

The combination of continuation and branch switching is very powerful.

Bad news

However, it has some disadvantages and weaknesses, too.

Downside A

You have to solve a lot of different problems.

Downside A

You have to solve a lot of different problems.

We work for years to develop a good solver for

$$F(u, \lambda) = 0 \ldots$$

Downside A

You have to solve a lot of different problems.

We work for years to develop a good solver for

$$F(u, \lambda) = 0 \ldots$$

but now we need to solve

$$\begin{bmatrix} F(u,\lambda) \\ p(u,\lambda,s) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \qquad F_u(u,\lambda)v = \lambda v \qquad \begin{bmatrix} F(u,\lambda) \\ F_u(u,\lambda)v \\ \|v\|^2 - 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Downside A

You have to solve a lot of different problems.

We work for years to develop a good solver for

$$F(u, \lambda) = 0 \ldots$$

but now we need to solve

$$\begin{bmatrix} F(u,\lambda) \\ p(u,\lambda,s) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \qquad F_u(u,\lambda)v = \lambda v \qquad \begin{bmatrix} F(u,\lambda) \\ F_u(u,\lambda)v \\ \|v\|^2 - 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Large-scale

This is OK when you can afford direct solvers, but it's hard at large scale.

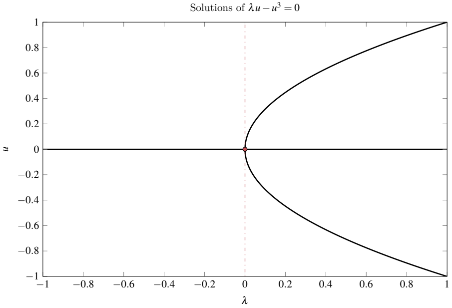
Downside B

We can only find branches *connected* to our initial data.

Downside B

We can only find branches *connected* to our initial data.

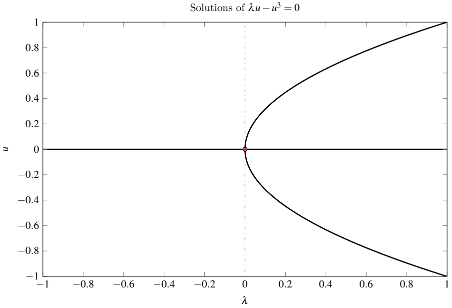
This works fine . . .



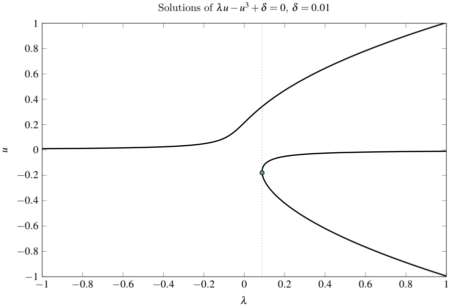
Downside B

We can only find branches *connected* to our initial data.

This works fine . . .



. . . but this does not.



The standard approach to deal with this is to

(a) modify the problem to restore connectedness;

The standard approach to deal with this is to

(a) modify the problem to restore connectedness;

(b) apply continuation + branch switching;

The standard approach to deal with this is to

(a) modify the problem to restore connectedness;

(b) apply continuation + branch switching;

(c) continue the branches you find back to the problem you care about.

The standard approach to deal with this is to

(a) modify the problem to restore connectedness;

(b) apply continuation + branch switching;

(c) continue the branches you find back to the problem you care about.

Problem A

You have to know to look for the missing branches.

The standard approach to deal with this is to

(a) modify the problem to restore connectedness;

(b) apply continuation + branch switching;

(c) continue the branches you find back to the problem you care about.

Problem A

You have to know to look for the missing branches.

Problem B

Executing this is manual and tedious.

The standard approach to deal with this is to

(a) modify the problem to restore connectedness;

(b) apply continuation + branch switching;

(c) continue the branches you find back to the problem you care about.

Problem A

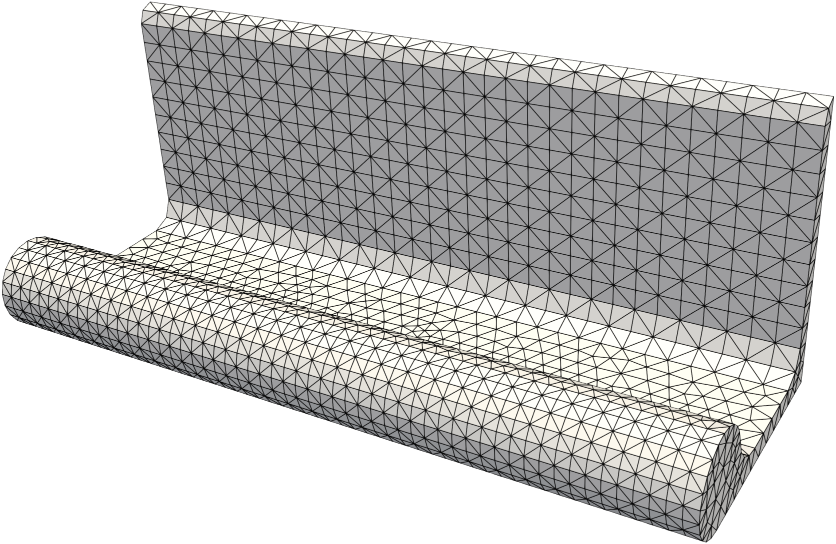
You have to know to look for the missing branches.

Problem B

Executing this is manual and tedious.

Problem C

Restoring connectedness is not always possible!



The connectedness is broken by non-symmetry of the domain.

Deflation offers a complementary approach.

Disconnected diagrams

An algorithm that can compute **disconnected bifurcation diagrams**.

Deflation offers a complementary approach.

Disconnected diagrams

An algorithm that can compute **disconnected bifurcation diagrams**.

Simplicity & scaling

The computational kernel is exactly the same as Newton's method: solve

$$F_u(u, \lambda)\delta u = -F(u, \lambda).$$

Section 2

Deflation

Deflation

Fix parameter λ. Given

- a Fréchet differentiable residual $F : X \to Y$
- a solution $u \in X$, $F(u) = 0$, $F_u(u)$ nonsingular

Deflation

Fix parameter λ. Given

- a Fréchet differentiable residual $F : X \to Y$
- a solution $u \in X$, $F(u) = 0$, $F_u(u)$ nonsingular

construct a **new nonlinear problem** $G : X \to Y$ such that:

Deflation

Fix parameter λ. Given

- a Fréchet differentiable residual $F : X \to Y$
- a solution $u \in X$, $F(u) = 0$, $F_u(u)$ nonsingular

construct a **new nonlinear problem** $G : X \to Y$ such that:

- (Preservation of solutions) $F(\tilde{u}) = 0 \iff G(\tilde{u}) = 0 \; \forall \; \tilde{u} \neq u$;

Deflation

Fix parameter λ. Given

- a Fréchet differentiable residual $F : X \to Y$
- a solution $u \in X$, $F(u) = 0$, $F_u(u)$ nonsingular

construct a **new nonlinear problem** $G : X \to Y$ such that:

- (Preservation of solutions) $F(\tilde{u}) = 0 \iff G(\tilde{u}) = 0 \ \forall \ \tilde{u} \neq u$;
- (Deflation property) Newton–Kantorovich applied to G will never converge to u again, starting from any initial guess.

Deflation

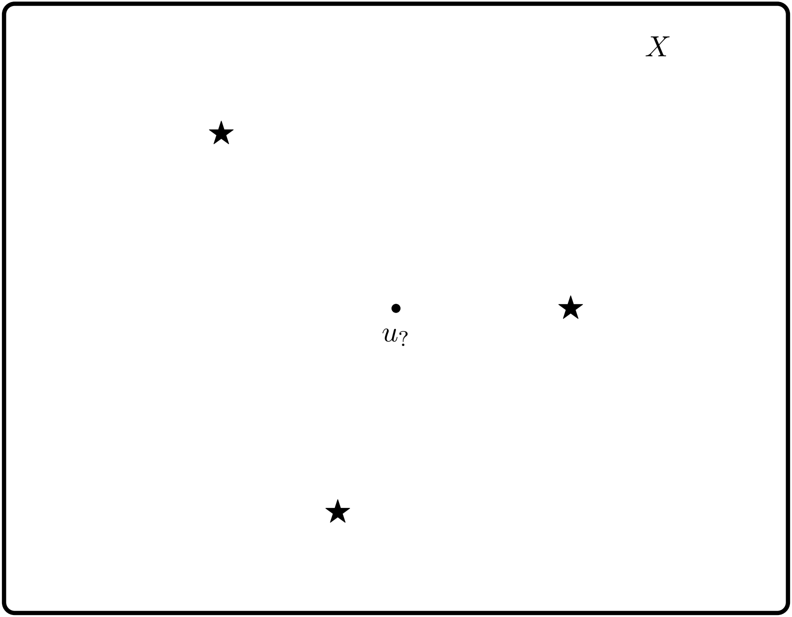
Fix parameter λ. Given

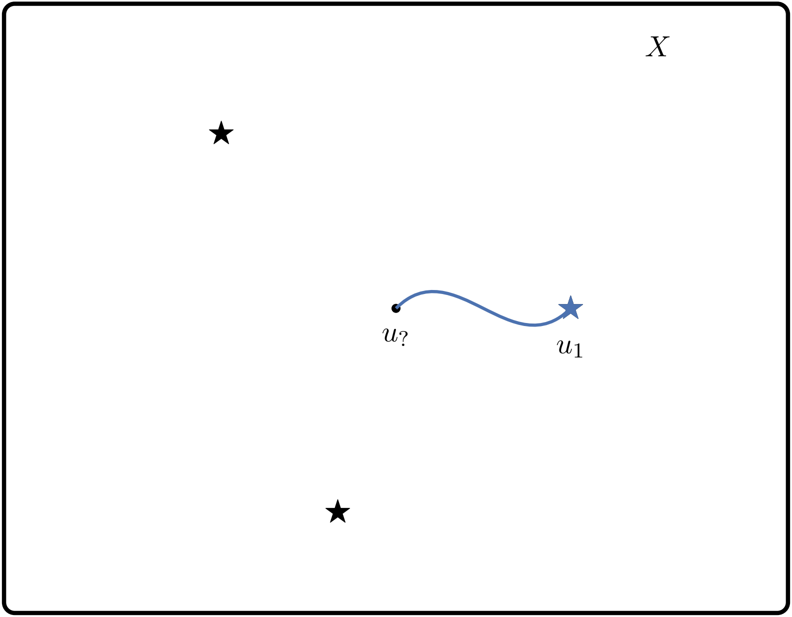
- a Fréchet differentiable residual $F : X \to Y$
- a solution $u \in X$, $F(u) = 0$, $F_u(u)$ nonsingular

construct a **new nonlinear problem** $G : X \to Y$ such that:

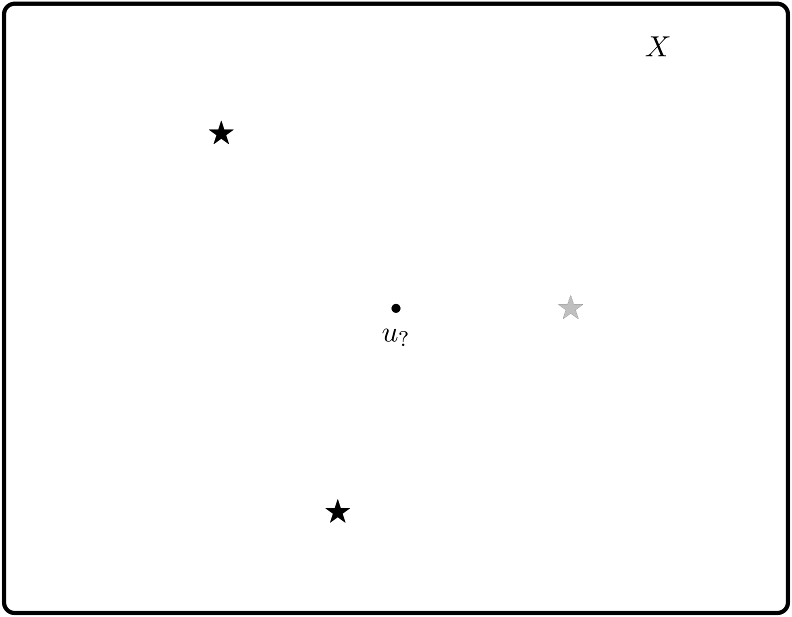
- (Preservation of solutions) $F(\tilde{u}) = 0 \iff G(\tilde{u}) = 0 \; \forall \; \tilde{u} \neq u$;
- (Deflation property) Newton–Kantorovich applied to G will never converge to u again, starting from any initial guess.

Find more solutions, starting from the same initial guess.

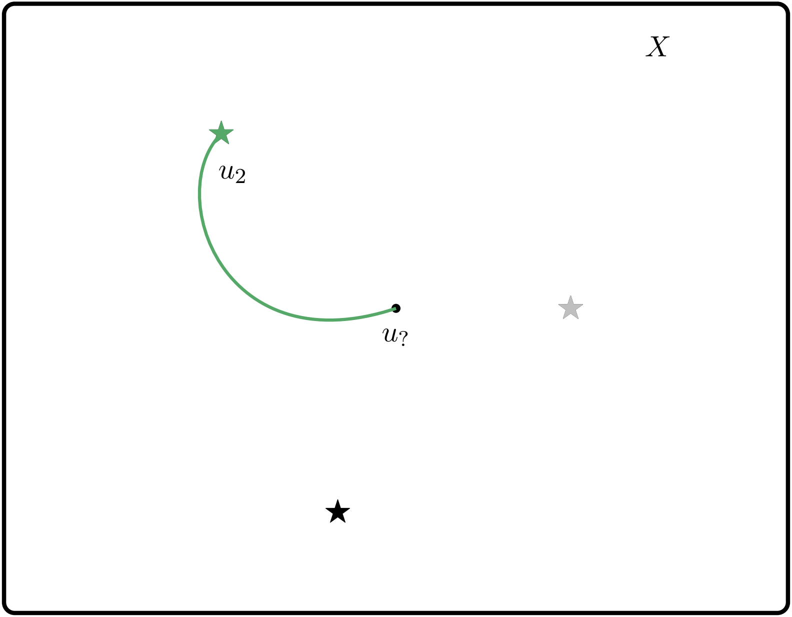

X
$u_?$



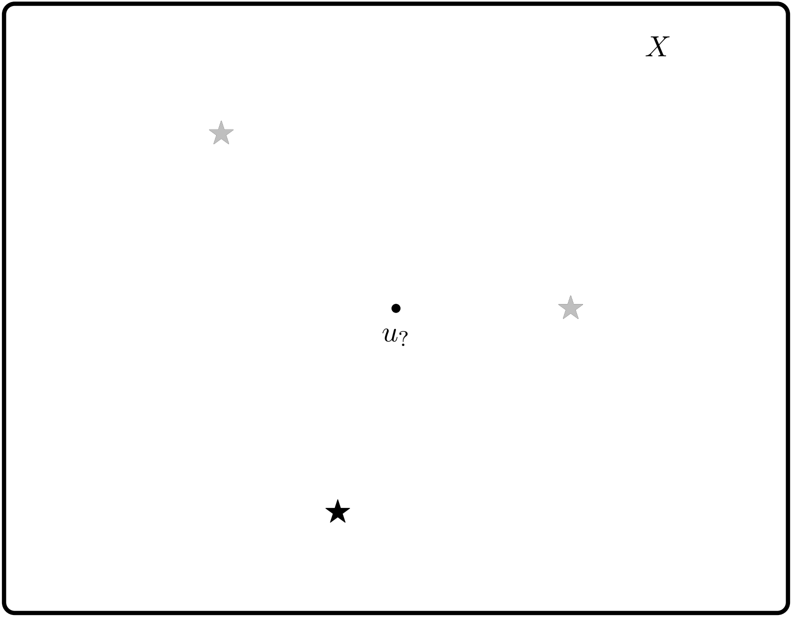
Newton from initial guess.



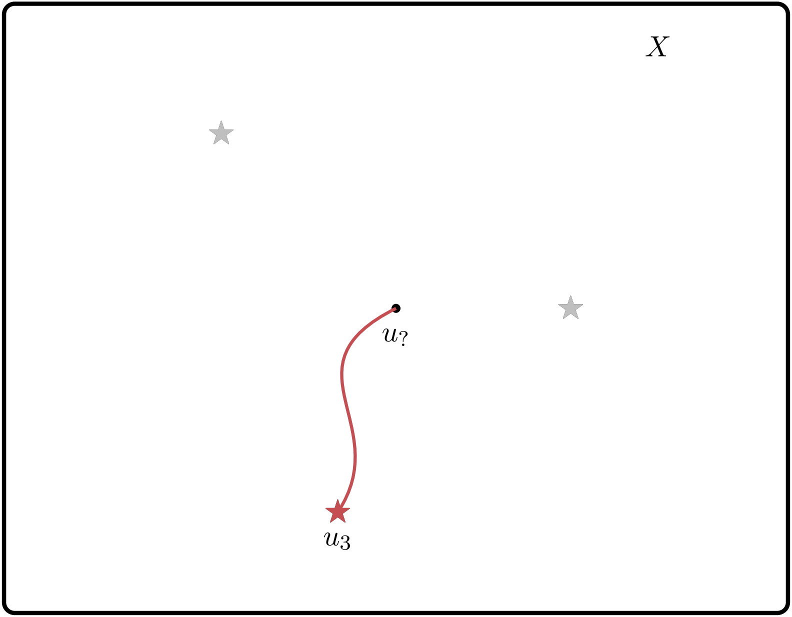
Deflate solution found.



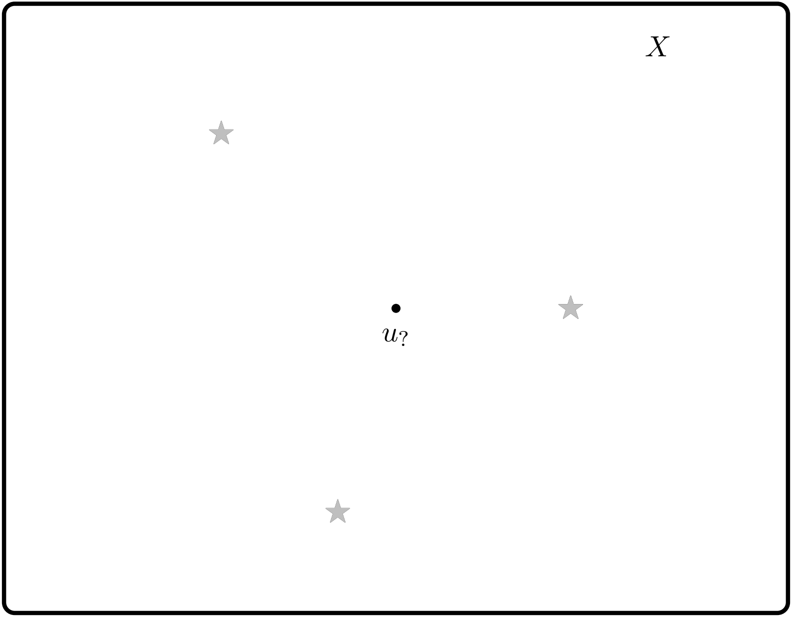
Newton from initial guess.



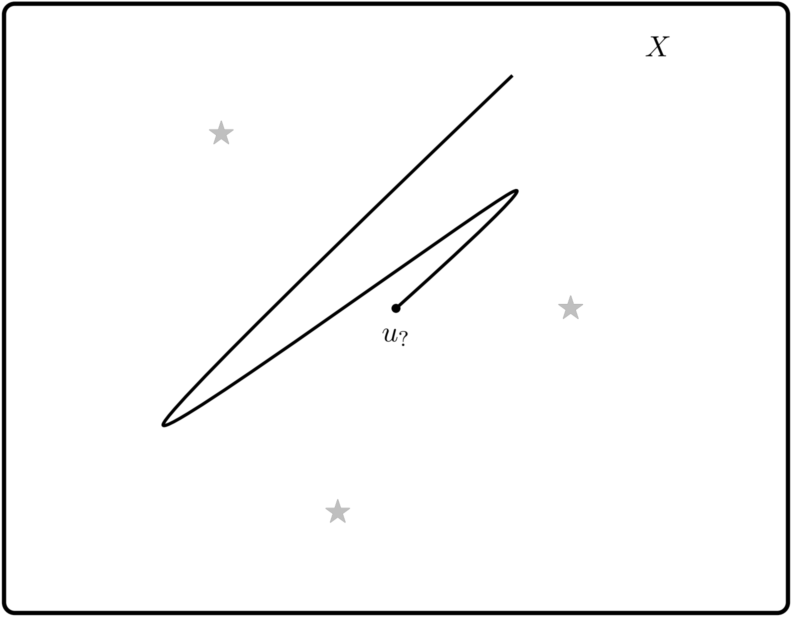
Deflate solution found.



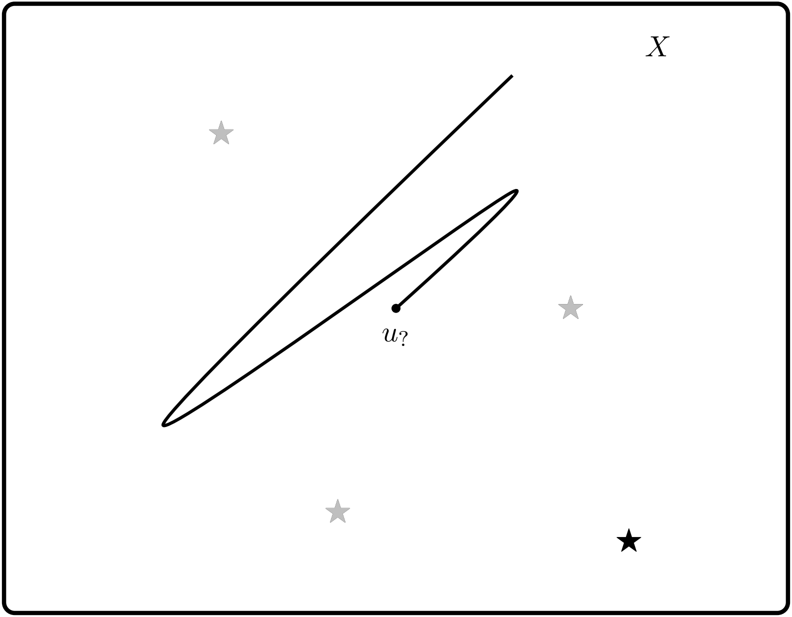
Newton from initial guess.



Deflate solution found.



Terminate on nonconvergence.



Terminate on nonconvergence.

Big if true. How can you do it?

Big if true. How can you do it?

Numer. Math. 16, 334–342 (1971)

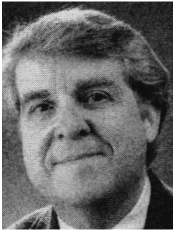
Deflation Techniques for the Calculation of Further Solutions of a Nonlinear System

KENNETH M. BROWN and WILLIAM B. GEARHART

Received March 10, 1970

Summary. This paper defines several classes of methods which can be used to find additional solutions of a nonlinear system of equations. A theory which embraces these classes is presented and the theory is extended to the multiple root problem. The techniques developed can also be used in avoiding previously found extreme points when performing function minimization. Results of computer experiments are presented.

Kenneth Brown, ?–?



Bill Gearhart, ?–

Brown & Gearhart's criterion

We say that $M(u;r)$ is a *deflation operator* if

$$\liminf_{u \to r} \|G(u)\| := \liminf_{u \to r} \|M(u;r)F(u)\| > 0.$$

Brown & Gearhart's criterion

We say that $M(u;r)$ is a *deflation operator* if

$$\liminf_{u \to r} \|G(u)\| := \liminf_{u \to r} \|M(u;r)F(u)\| > 0.$$

Brown & Gearhart's proposal

Choose

$$M(u;r) := \frac{1}{\|u - r\|}.$$

Note that $M(u,r) > 0$ always, so $G(u) = 0 \iff F(u) = 0$.

Brown & Gearhart's criterion

We say that $M(u;r)$ is a *deflation operator* if

$$\liminf_{u \to r} \|G(u)\| := \liminf_{u \to r} \|M(u;r)F(u)\| > 0.$$

Brown & Gearhart's proposal

Choose

$$M(u;r) := \frac{1}{\|u-r\|}.$$

Note that $M(u,r) > 0$ always, so $G(u) = 0 \iff F(u) = 0$.

Since $\|F(u)\| = O(\|u-r\|)$ as $u \to r$, this works.

. . . albeit sometimes not robustly.

. . . albeit sometimes not robustly.

Numerical experience with deflation has shown it is often a matter of seeming chance whether one obtains an additional solution.

(Allgower & Georg, 1990)

. . . albeit sometimes not robustly.

Numerical experience with deflation has shown it is often a matter of seeming chance whether one obtains an additional solution.

(Allgower & Georg, 1990)

[Deflation is] not . . . very reliable for larger problems.

(Kanzow, 2000)

... albeit sometimes not robustly.

> *Numerical experience with deflation has shown it is often a matter of seeming chance whether one obtains an additional solution.*
>
> (Allgower & Georg, 1990)

> *[Deflation is] not ... very reliable for larger problems.*
>
> (Kanzow, 2000)

Why?

. . . albeit sometimes not robustly.

> *Numerical experience with deflation has shown it is often a matter of seeming chance whether one obtains an additional solution.*
>
> (Allgower & Georg, 1990)

> *[Deflation is] not . . . very reliable for larger problems.*
>
> (Kanzow, 2000)

Why?

One problem: assuming F does not blow up as $\|u - r\| \to \infty$, then Newton discovers that it can achieve

$$\|G(u)\|_Y < \mathrm{tol}$$

for any tol, by taking $\|u - r\|$ large enough.

Our proposal

$$M_p(u; r) := \left(\frac{1}{\|u - r\|^p} + 1 \right), \quad p \geq 1.$$

Ásgeir Birkisson, 1985–

Simon Funke, 1983–

Our proposal

$$M_p(u;r) := \left(\frac{1}{\|u-r\|^p} + 1\right), \quad p \geq 1.$$

This has the right behaviour *both* as

$$\|u-r\| \to 0,$$
$$\|u-r\| \to \infty.$$

Ásgeir Birkisson, 1985–

Simon Funke, 1983–

Our proposal

$$M_p(u;r) := \left(\frac{1}{\|u-r\|^p} + 1\right), \quad p \geq 1.$$

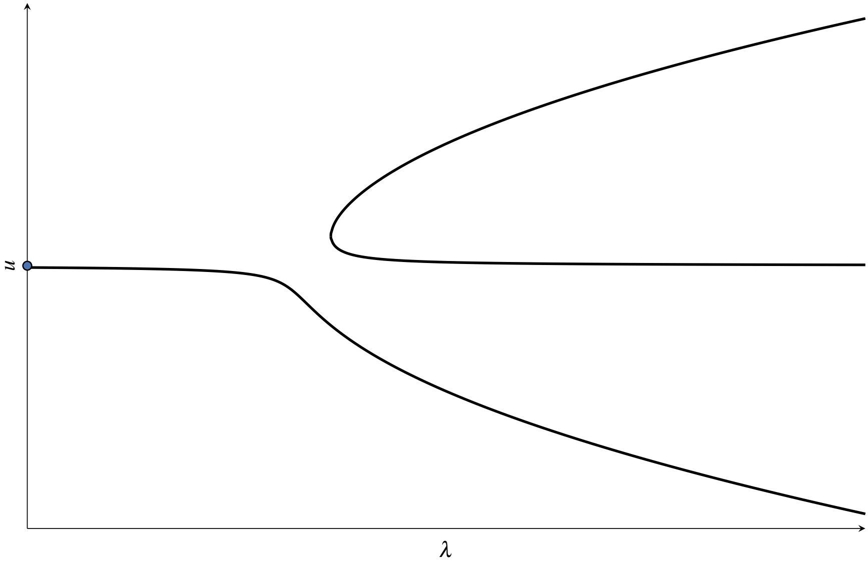
This has the right behaviour *both* as

$$\begin{aligned}\|u-r\| &\to 0,\\ \|u-r\| &\to \infty.\end{aligned}$$

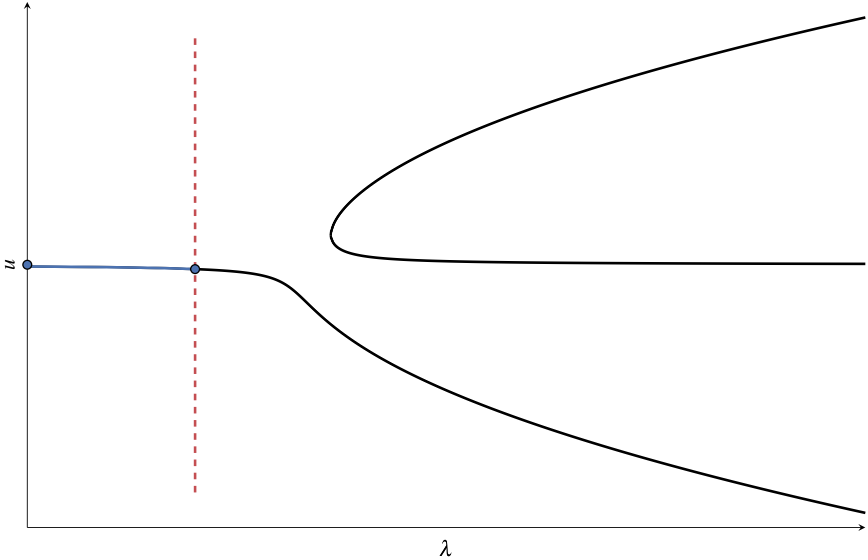
This makes the procedure much more reliable.

Ásgeir Birkisson, 1985–

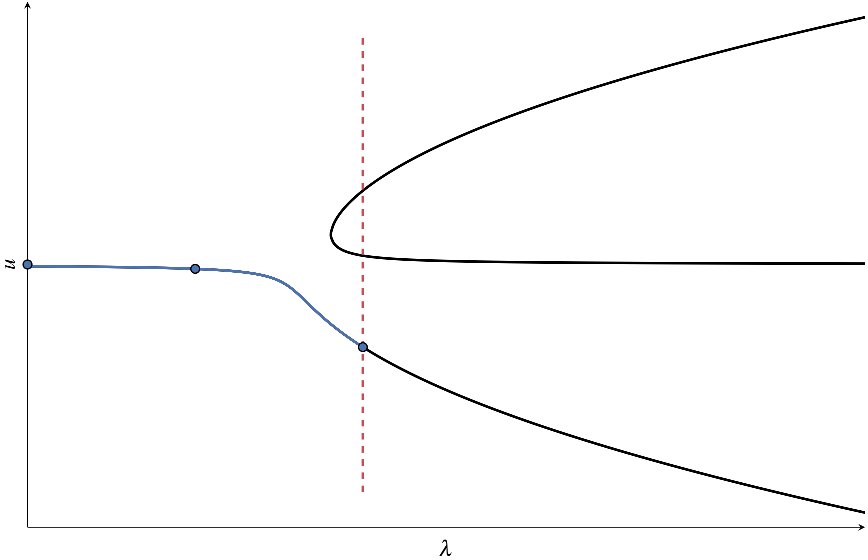
Simon Funke, 1983–



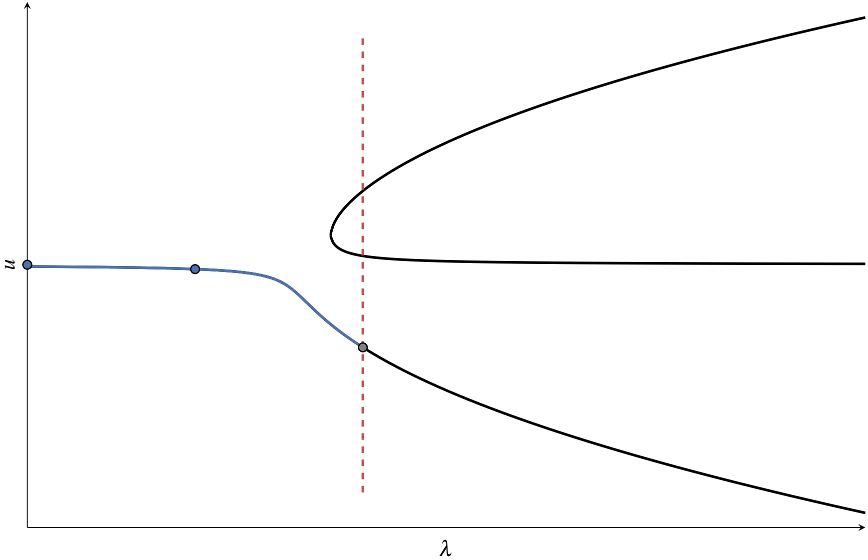
Start with (u_0, λ_0).



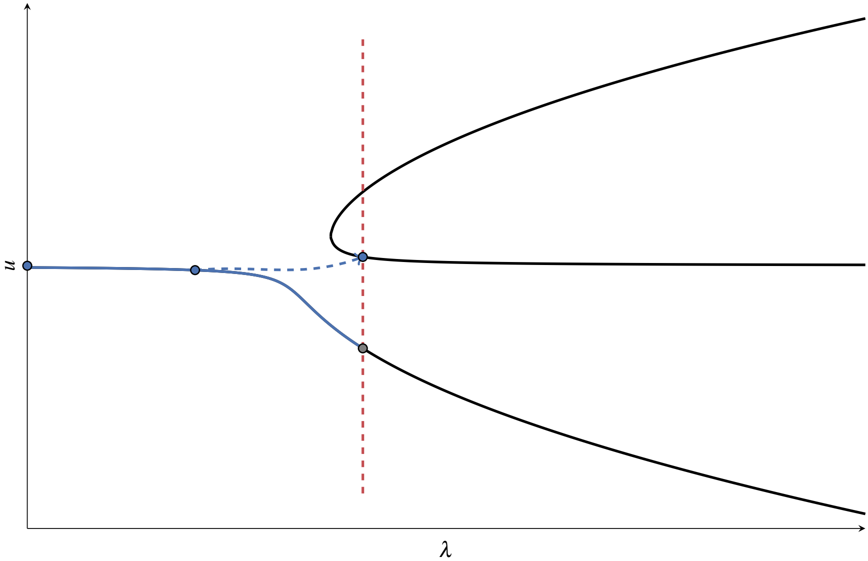
Perform a continuation step.



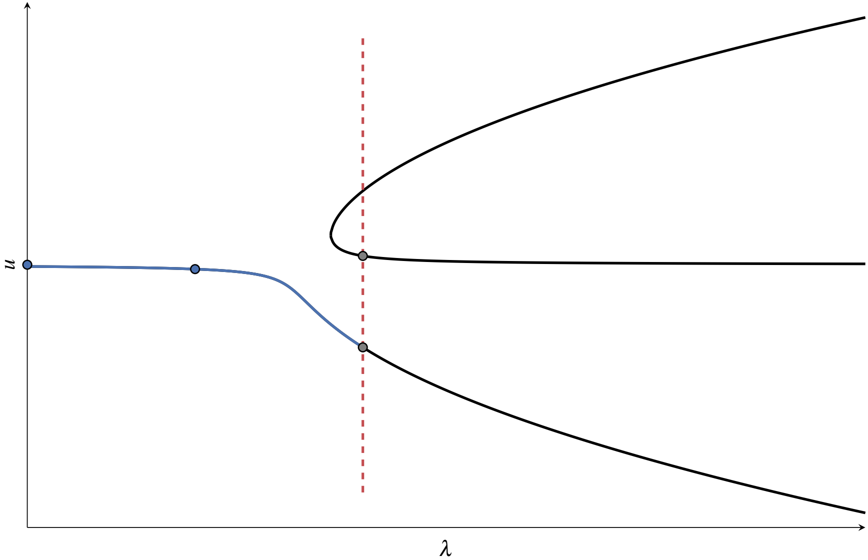
Perform another continuation step.



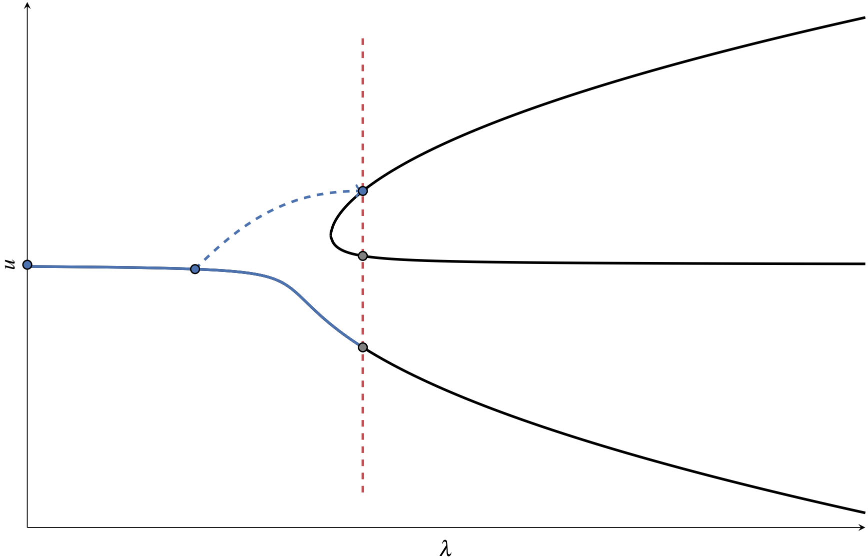
Deflate the solution found.



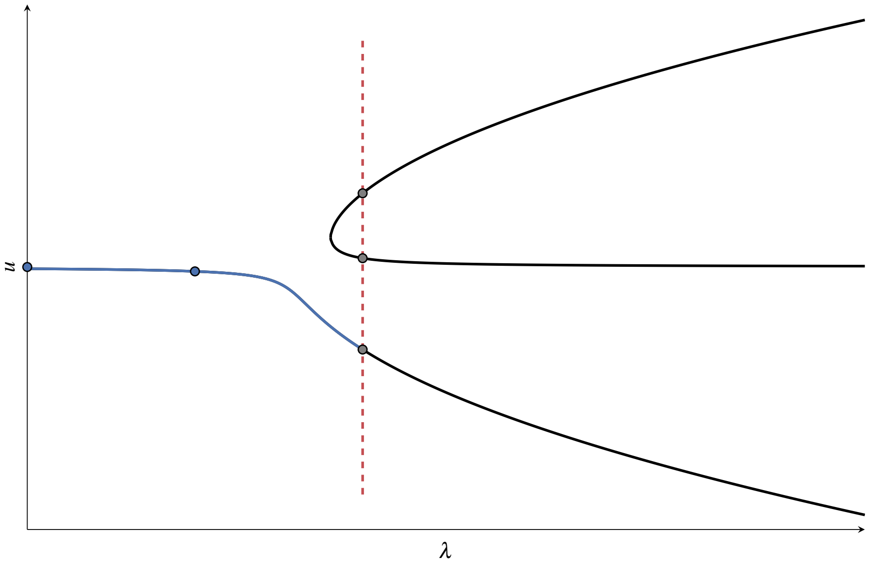
Solve again.



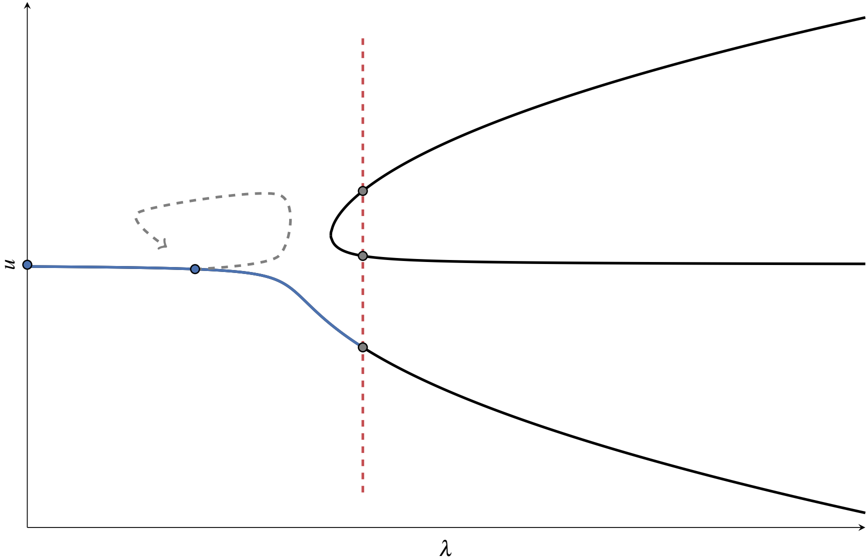
Deflate the solution found.



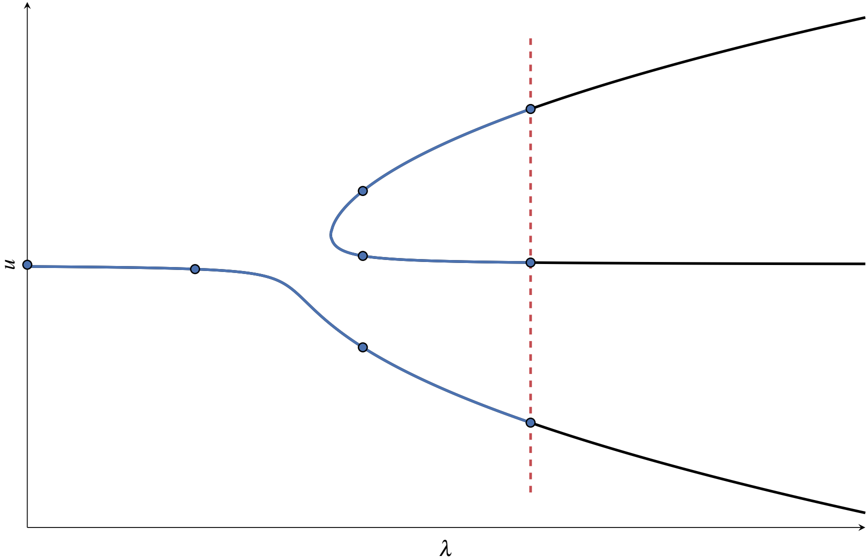
Solve again.



Deflate the solution found.



Search again, unsuccessfully.



Repeat.

Good news

Deflation lets us discover disconnected branches!

Section 3

Solving the deflated problem

We assume we have a good solver for our discretised Newton step

$$F_u(u, \lambda)\delta u_F = -F(u, \lambda), \quad F \in C^1(\mathbb{R}^N \times \mathbb{R}, \mathbb{R}^N).$$

We assume we have a good solver for our discretised Newton step

$$F_u(u,\lambda)\delta u_F = -F(u,\lambda), \quad F \in C^1(\mathbb{R}^N \times \mathbb{R}, \mathbb{R}^N).$$

We now want to solve

$$G_u(u,\lambda)\delta u_G = -G(u,\lambda)$$

where

$$G(u,\lambda) = M(u;u_1)M(u;u_2)\cdots M(u;u_n)F(u,\lambda) =: M(u)F(u,\lambda).$$

We assume we have a good solver for our discretised Newton step

$$F_u(u, \lambda)\delta u_F = -F(u, \lambda), \quad F \in C^1(\mathbb{R}^N \times \mathbb{R}, \mathbb{R}^N).$$

We now want to solve

$$G_u(u, \lambda)\delta u_G = -G(u, \lambda)$$

where

$$G(u, \lambda) = M(u; u_1)M(u; u_2) \cdots M(u; u_n)F(u, \lambda) =: M(u)F(u, \lambda).$$

Good news

You can compute δu_G easily from δu_F!

By the product rule,

$$G_u(u, \lambda) = M(u) F_u(u, \lambda) + F(u, \lambda) M_u^\top.$$

By the product rule,

$$G_u(u, \lambda) = M(u)F_u(u, \lambda) + F(u, \lambda)M_u^\top.$$

By the product rule,

$$G_u(u, \lambda) = M(u)F_u(u, \lambda) + F(u, \lambda)M_u^\top.$$

At first this looks bad. The deflated Jacobian is dense, as it is a rank-one update of a sparse matrix.

By the product rule,

$$G_u(u, \lambda) = M(u)F_u(u, \lambda) + F(u, \lambda)M_u^\top.$$

At first this looks bad. The deflated Jacobian is dense, as it is a rank-one update of a sparse matrix.

Sherman–Morrison–Woodbury formula

$$\left(A + uv^\top\right)^{-1} = A^{-1} - \left(\frac{A^{-1}uv^\top A^{-1}}{1 + v^\top A^{-1}u}\right).$$

Maurice Bartlett, 1910–2002

By the product rule,

$$G_u(u, \lambda) = M(u) F_u(u, \lambda) + F(u, \lambda) M_u^\top.$$

At first this looks bad. The deflated Jacobian is dense, as it is a rank-one update of a sparse matrix.

Sherman–Morrison–Woodbury formula

$$\left(A + uv^\top\right)^{-1} = A^{-1} - \left(\frac{A^{-1} u v^\top A^{-1}}{1 + v^\top A^{-1} u}\right).$$

Maurice Bartlett, 1910–2002

At first it looks like applying this to a vector w requires two solves with A: $A^{-1}u$ and $A^{-1}w$. But something magical happens . . .

Applying the Sherman–Morrison–Woodbury formula, we have

$$\delta u_G = -[G_u]^{-1} G = -\left(M F_u + F M_u^\top \right)^{-1} (MF)$$

Applying the Sherman–Morrison–Woodbury formula, we have

$$
\begin{aligned}
\delta u_G = -[G_u]^{-1} G &= -\left(M F_u + F M_u^\top \right)^{-1} (MF) \\
&= -\left[M^{-1} F_u^{-1} - \frac{M^{-1} F_u^{-1} F M_u^\top M^{-1} F_u^{-1}}{1 + M_u^\top M^{-1} F_u^{-1} F} \right] (MF)
\end{aligned}
$$

Applying the Sherman–Morrison–Woodbury formula, we have

$$\begin{aligned}
\delta u_G = -[G_u]^{-1} G &= -\left(M F_u + F M_u^\top\right)^{-1} (MF) \\
&= -\left[M^{-1} F_u^{-1} - \frac{M^{-1} F_u^{-1} F M_u^\top M^{-1} F_u^{-1}}{1 + M_u^\top M^{-1} F_u^{-1} F}\right] (MF) \\
&= -F_u^{-1} F + \frac{F_u^{-1} F M_u^\top M^{-1} F_u^{-1} F}{1 + M_u^\top M^{-1} F_u^{-1} F}
\end{aligned}$$

Applying the Sherman–Morrison–Woodbury formula, we have

$$
\begin{aligned}
\delta u_G = -[G_u]^{-1} G &= -\left(M F_u + F M_u^\top \right)^{-1} (MF) \\
&= -\left[M^{-1} F_u^{-1} - \frac{M^{-1} F_u^{-1} F M_u^\top M^{-1} F_u^{-1}}{1 + M_u^\top M^{-1} F_u^{-1} F} \right] (MF) \\
&= -F_u^{-1} F + \frac{F_u^{-1} F M_u^\top M^{-1} F_u^{-1} F}{1 + M_u^\top M^{-1} F_u^{-1} F} \\
&= \left(1 - \frac{M^{-1} M_u^\top F_u^{-1} F}{1 + M_u^\top M^{-1} F_u^{-1} F} \right) \left(-F_u^{-1} F \right)
\end{aligned}
$$

Applying the Sherman–Morrison–Woodbury formula, we have

$$
\begin{aligned}
\delta u_G = -[G_u]^{-1} G &= -\left(M F_u + F M_u^\top\right)^{-1} (MF) \\
&= -\left[M^{-1} F_u^{-1} - \frac{M^{-1} F_u^{-1} F M_u^\top M^{-1} F_u^{-1}}{1 + M_u^\top M^{-1} F_u^{-1} F}\right] (MF) \\
&= -F_u^{-1} F + \frac{F_u^{-1} F M_u^\top M^{-1} F_u^{-1} F}{1 + M_u^\top M^{-1} F_u^{-1} F} \\
&= \left(1 - \frac{M^{-1} M_u^\top F_u^{-1} F}{1 + M_u^\top M^{-1} F_u^{-1} F}\right) \left(-F_u^{-1} F\right) \\
&= \left(1 + \frac{M^{-1} M_u^\top \delta u_F}{1 - M^{-1} M_u^\top \delta u_F}\right) \delta u_F.
\end{aligned}
$$

Applying the Sherman–Morrison–Woodbury formula, we have

$$\begin{aligned}
\delta u_G = -[G_u]^{-1}G &= -\left(MF_u + FM_u^\top\right)^{-1}(MF) \\
&= -\left[M^{-1}F_u^{-1} - \frac{M^{-1}F_u^{-1}FM_u^\top M^{-1}F_u^{-1}}{1 + M_u^\top M^{-1}F_u^{-1}F}\right](MF) \\
&= -F_u^{-1}F + \frac{F_u^{-1}FM_u^\top M^{-1}F_u^{-1}F}{1 + M_u^\top M^{-1}F_u^{-1}F} \\
&= \left(1 - \frac{M^{-1}M_u^\top F_u^{-1}F}{1 + M_u^\top M^{-1}F_u^{-1}F}\right)\left(-F_u^{-1}F\right) \\
&= \left(1 + \frac{M^{-1}M_u^\top \delta u_F}{1 - M^{-1}M_u^\top \delta u_F}\right)\delta u_F.
\end{aligned}$$

So we only need to solve one system with F_u!

Solving the deflated problem

To solve

$$G_u \delta u_G = -G,$$

do the following:

Solving the deflated problem

To solve

$$G_u \delta u_G = -G,$$

do the following:

1. Solve

$$F_u \delta u_F = -F.$$

Solving the deflated problem

To solve

$$G_u \delta u_G = -G,$$

do the following:

1. Solve
$$F_u \delta u_F = -F.$$
2. Evaluate
$$p = M_u^\top \delta u_F.$$

Solving the deflated problem

To solve

$$G_u \delta u_G = -G,$$

do the following:

1. Solve
$$F_u \delta u_F = -F.$$
2. Evaluate
$$p = M_u^\top \delta u_F.$$
3. Evaluate
$$\tau = 1 + \frac{M^{-1} p}{1 - M^{-1} p}.$$

Solving the deflated problem

To solve

$$G_u \delta u_G = -G,$$

do the following:

1. Solve
$$F_u \delta u_F = -F.$$
2. Evaluate
$$p = M_u^\top \delta u_F.$$
3. Evaluate
$$\tau = 1 + \frac{M^{-1} p}{1 - M^{-1} p}.$$
4. Return
$$\delta u_G = \tau \delta u_F.$$

Good news

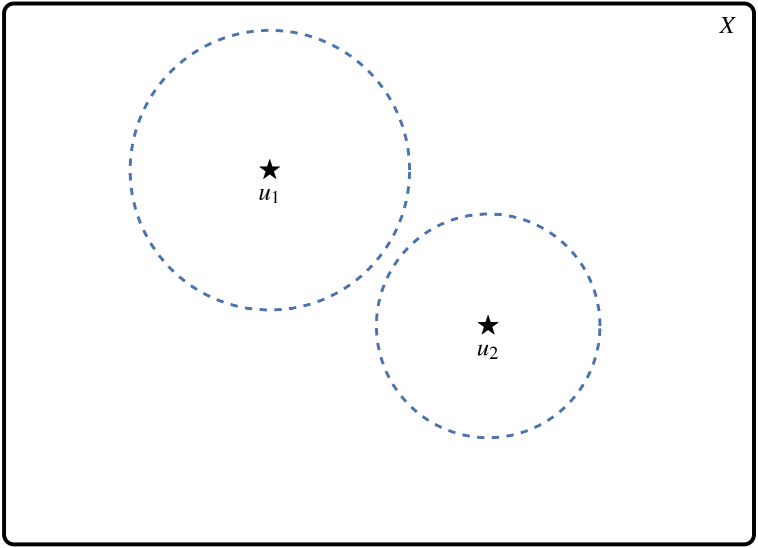
You can apply deflation to massive discretisations.

Section 4

Convergence of deflation

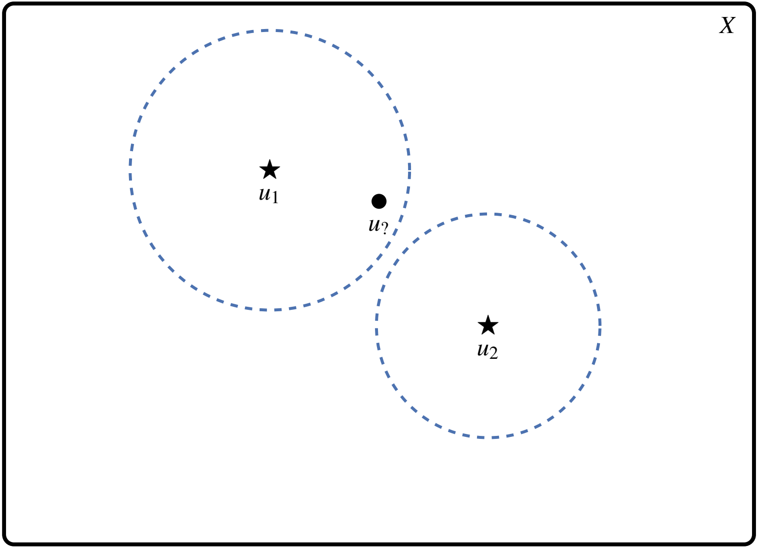
It is possible to give sufficient conditions for deflation to find two roots.

It is possible to give sufficient conditions for deflation to find two roots.



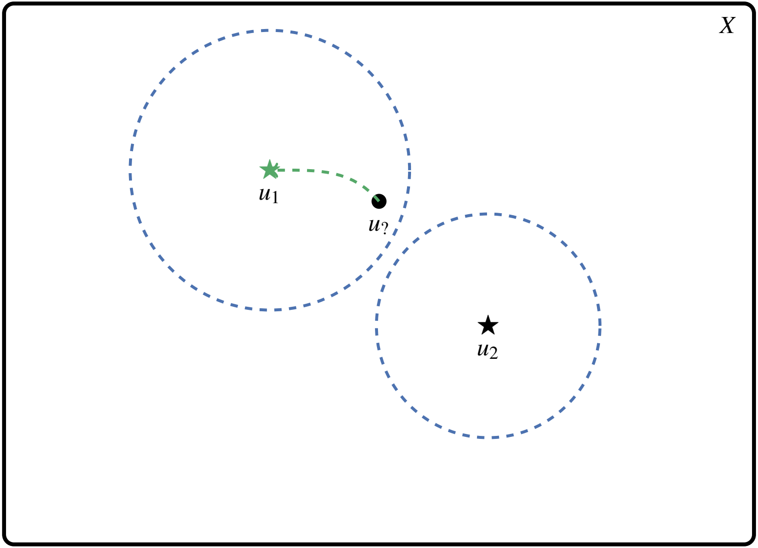
Two solutions, with Rall–Rheinboldt balls.

It is possible to give sufficient conditions for deflation to find two roots.



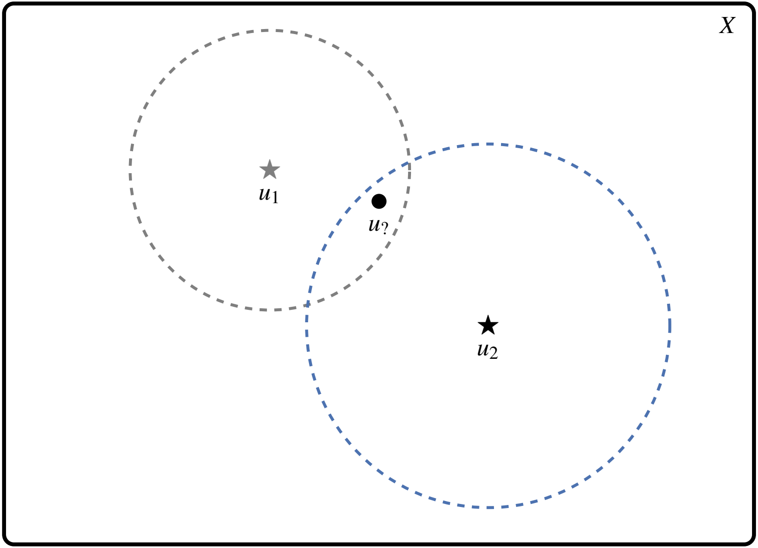
Start with an initial guess within a ball.

It is possible to give sufficient conditions for deflation to find two roots.



Converge to that solution.

It is possible to give sufficient conditions for deflation to find two roots.



Deflate that solution; the other Rall–Rheinboldt ball expands.

Section 5

Examples

Allen–Cahn equation

$$F(u, \lambda) = -\lambda^2 \nabla^2 u + u^3 - u = 0, \quad u = g \text{ on } \partial\Omega.$$

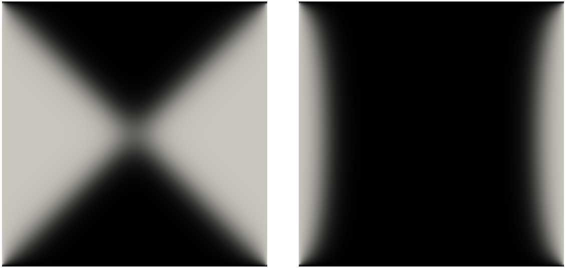
Allen–Cahn equation

$$F(u, \lambda) = -\lambda^2 \nabla^2 u + u^3 - u = 0, \quad u = g \text{ on } \partial\Omega.$$

Solutions found starting from $u = 0$ for $\lambda = 0.04$.

Allen–Cahn equation

$$F(u, \lambda) = -\lambda^2 \nabla^2 u + u^3 - u = 0, \quad u = g \text{ on } \partial\Omega.$$



Solutions found starting from $u = 0$ for $\lambda = 0.04$.

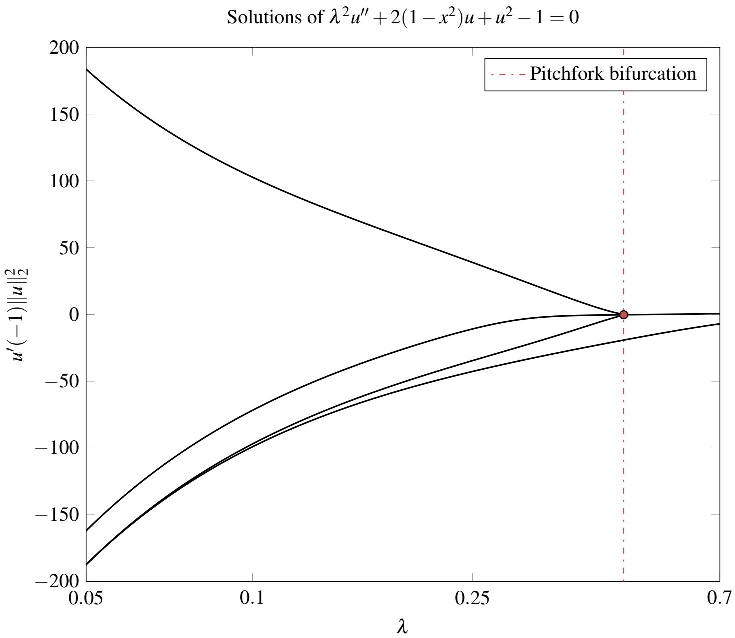
Allen–Cahn equation

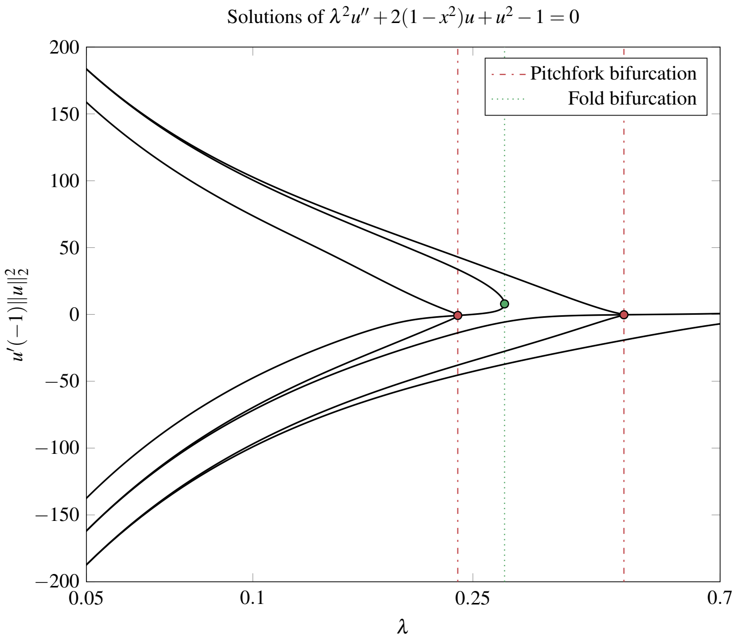
$$F(u,\lambda) = -\lambda^2\nabla^2 u + u^3 - u = 0, \quad u = g \text{ on } \partial\Omega.$$

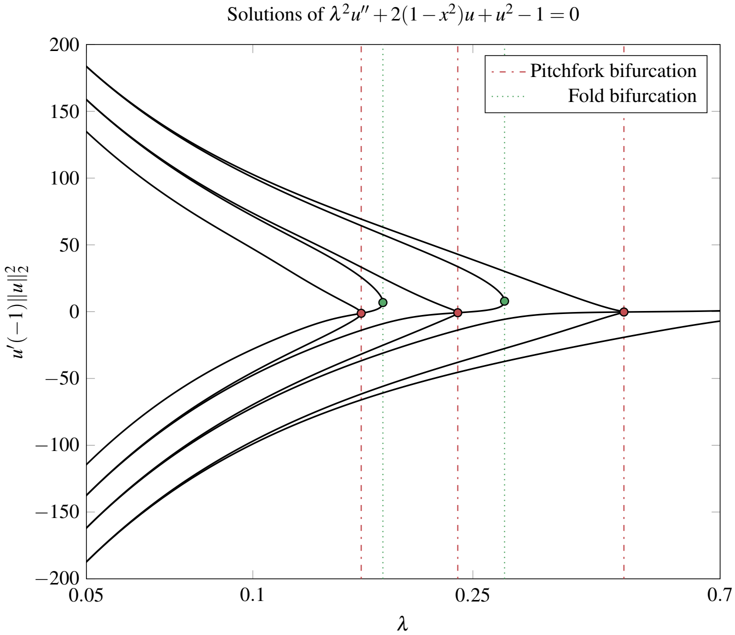
Solutions found starting from $u = 0$ for $\lambda = 0.04$.

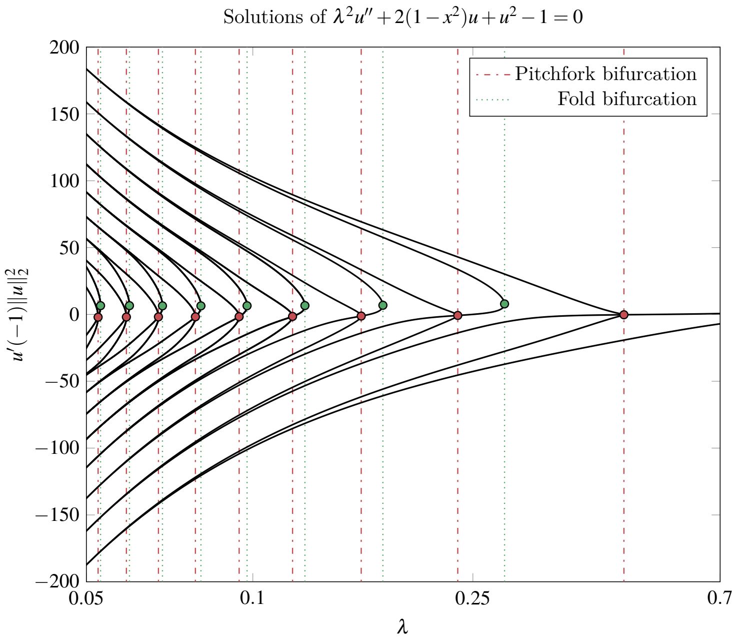
Carrier's equation

$$F(u,\lambda) = \lambda^2 u'' + 2(1-x^2)u + u^2 - 1 = 0, \quad u(-1) = 0 = u(1).$$


Solutions of $\lambda^2 u'' + 2(1-x^2)u + u^2 - 1 = 0$
Pitchfork bifurcation
200
150
100
50
0
−50
−100
−150
−200
$u'(-1)\|u\|_2^2$
0.05
0.1
0.25
0.7
λ


Solutions of $\lambda^2 u'' + 2(1-x^2)u + u^2 - 1 = 0$
Pitchfork bifurcation
Fold bifurcation
200
150
100
50
0
−50
−100
−150
−200
$u'(-1)\|u\|_2^2$
0.05
0.1
0.25
0.7
λ


Solutions of $\lambda^2 u'' + 2(1-x^2)u + u^2 - 1 = 0$
Pitchfork bifurcation
Fold bifurcation
200
150
100
50
0
−50
−100
−150
−200
0.05
0.1
0.25
0.7
$u'(-1)\|u\|_2^2$
λ

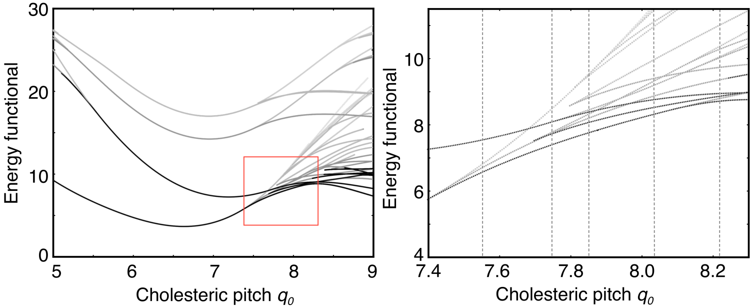

Solutions of $\lambda^2 u'' + 2(1-x^2)u + u^2 - 1 = 0$
Pitchfork bifurcation
Fold bifurcation
$u'(-1)\|u\|_2^2$
200
150
100
50
0
−50
−100
−150
−200
0.05
0.1
0.25
0.7
λ

Oseen–Frank

$$\min\ J = \int_\Omega K_1(\nabla \cdot u)^2 + K_2(u \cdot \nabla \times u + q_0)^2 + K_3 \left| u \times \nabla \times u \right|, \quad u \cdot u = 1.$$

Oseen–Frank

$$\min\ J = \int_{\Omega} K_1(\nabla \cdot u)^2 + K_2(u \cdot \nabla \times u + q_0)^2 + K_3 \left| u \times \nabla \times u \right|, \quad u \cdot u = 1.$$



Section 6

Symmetries

Symmetries

What if the equation has a continuous symmetry group?

Symmetries

What if the equation has a continuous symmetry group?

Philosophy

The fundamental structures are the distinct **orbits** of solutions.

Symmetries

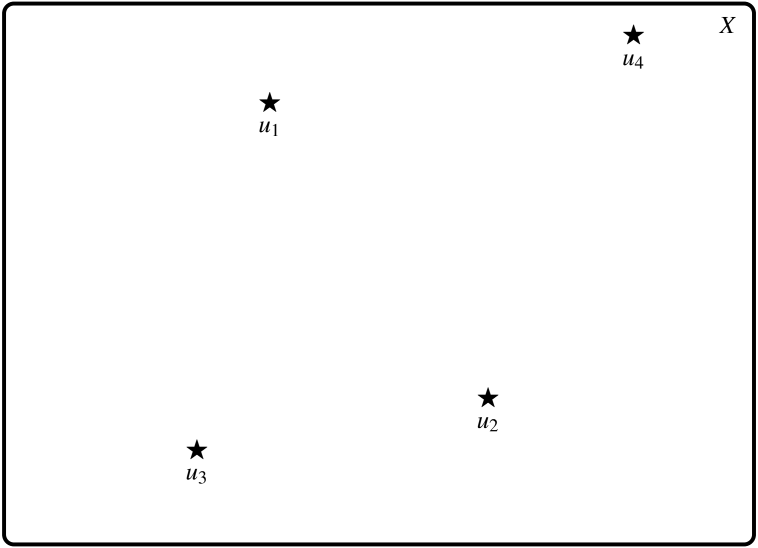
What if the equation has a continuous symmetry group?

Philosophy

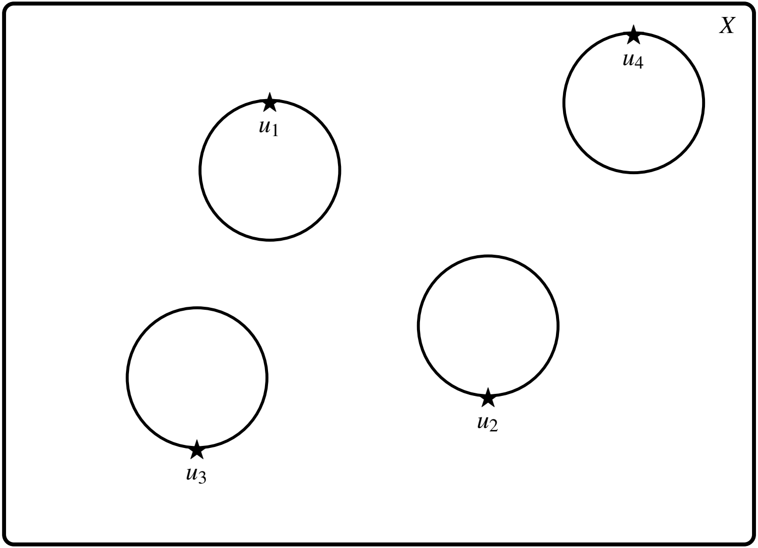
The fundamental structures are the distinct **orbits** of solutions.

Key idea

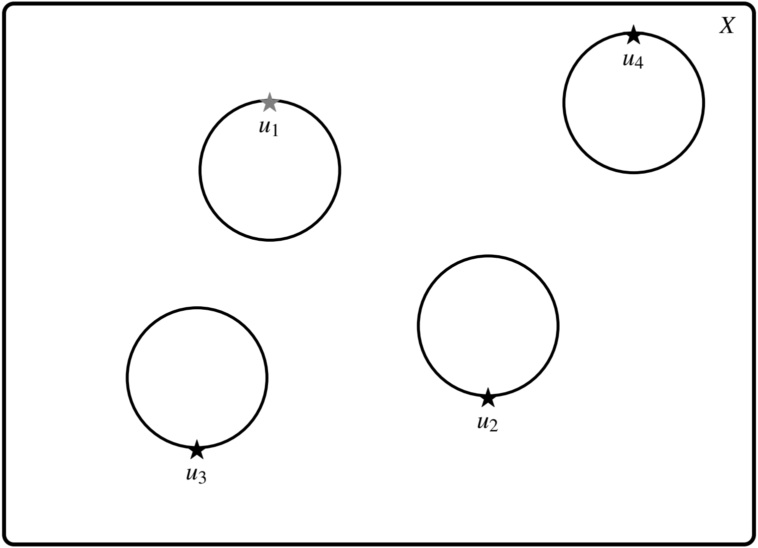
Construct a deflation operator invariant under the action of the Lie group.



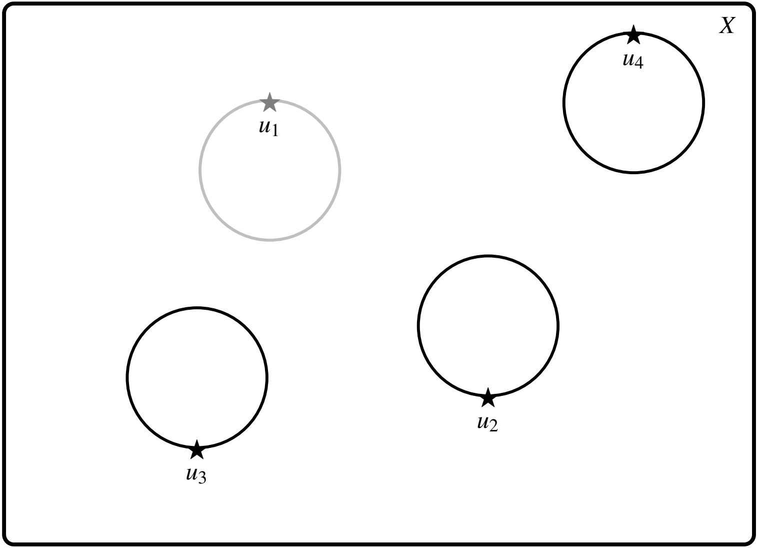
Four solutions, not related by the symmetry group.



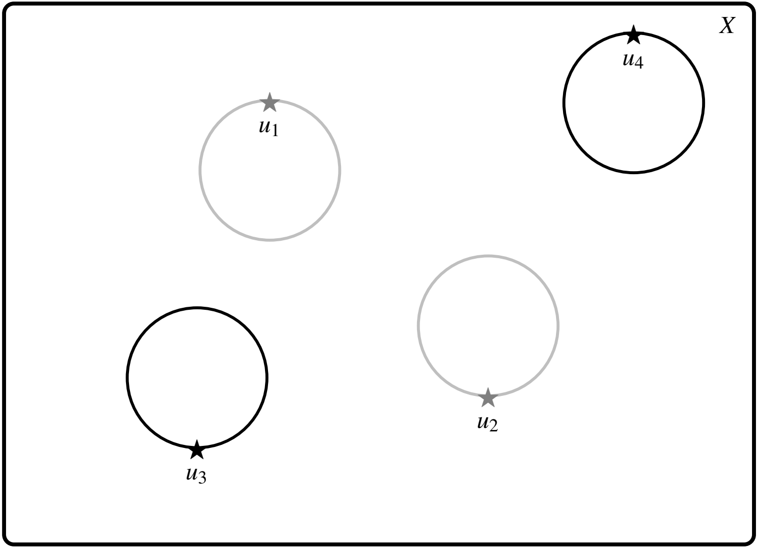
Each solution induces a *group orbit* of solutions, related by symmetry.



Not enough to deflate the solution—must deflate the entire orbit.



Design a deflation operator that deflates the entire orbit.



Design a deflation operator that deflates the entire orbit.

Gross–Pitaevskii equation

$$-\frac{1}{2}\Delta u + \frac{x^2 + y^2 + z^2}{2} u - \mu u + |u|^2 u = 0, \qquad u|_{\partial\Omega} = 0.$$

Gross–Pitaevskii equation

$$-\frac{1}{2}\Delta u + \frac{x^2 + y^2 + z^2}{2} u - \mu u + |u|^2 u = 0, \qquad u|_{\partial\Omega} = 0.$$

First symmetry group $\mathrm{SO}(2)$: phase shifts

$$u(\vec{x}) \mapsto e^{i\theta} u(\vec{x}), \quad \theta \in \mathbb{R}.$$

Gross–Pitaevskii equation

$$-\frac{1}{2}\Delta u + \frac{x^2 + y^2 + z^2}{2} u - \mu u + |u|^2 u = 0, \qquad u|_{\partial\Omega} = 0.$$

First symmetry group $\mathrm{SO}(2)$: phase shifts

$$u(\vec{x}) \mapsto e^{i\theta} u(\vec{x}), \quad \theta \in \mathbb{R}.$$

Invariant deflation operator

$$M(u; r) = \left\| |u|^2 - |r|^2 \right\|^{-2} + 1.$$

Gross–Pitaevskii equation

$$-\frac{1}{2}\Delta u + \frac{x^2 + y^2 + z^2}{2} u - \mu u + |u|^2 u = 0, \qquad u|_{\partial\Omega} = 0.$$

Second symmetry group $\mathrm{SO}(3)$: spatial rotations

$$u(\vec{x}) \mapsto u(R\vec{x}), \quad R^{-1} = R^T, \ \det(R) = 1.$$

Gross–Pitaevskii equation

$$-\frac{1}{2}\Delta u + \frac{x^2+y^2+z^2}{2}u - \mu u + |u|^2 u = 0, \qquad u|_{\partial\Omega} = 0.$$

Second symmetry group $\mathrm{SO}(3)$: spatial rotations

$$u(\vec{x}) \mapsto u(R\vec{x}), \quad R^{-1} = R^T, \ \det(R) = 1.$$

Invariant deflation operator

$$M(u;r) = \|\bar{u} - \bar{r}\|^{-2} + 1,$$

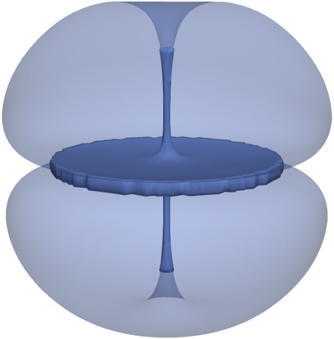
where

$\bar{u}(r, \theta, \psi)$ averages u over the sphere of radius r.

Gross–Pitaevskii equation

$$-\frac{1}{2}\Delta u + \frac{x^2+y^2+z^2}{2}u - \mu u + |u|^2 u = 0, \qquad u|_{\partial\Omega} = 0.$$

Solutions for $\mu = 6$.

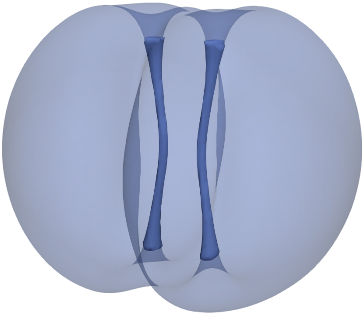


A vortex line and a planar dark soliton.

Gross–Pitaevskii equation

$$-\frac{1}{2}\Delta u + \frac{x^2 + y^2 + z^2}{2} u - \mu u + |u|^2 u = 0, \qquad u|_{\partial\Omega} = 0.$$

Solutions for $\mu = 6$.

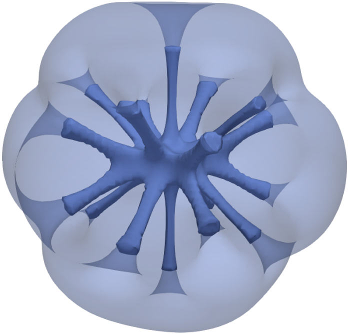


A pair of vortex lines.

Gross–Pitaevskii equation

$$-\frac{1}{2}\Delta u + \frac{x^2+y^2+z^2}{2}u - \mu u + |u|^2 u = 0, \qquad u|_{\partial\Omega} = 0.$$

Solutions for $\mu = 6$.

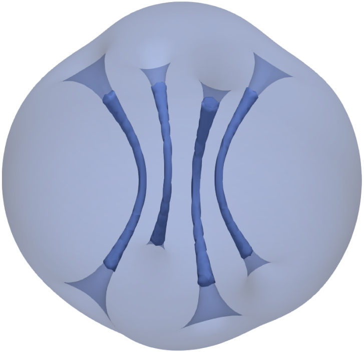


A vortex star.

Gross–Pitaevskii equation

$$-\frac{1}{2}\Delta u + \frac{x^2 + y^2 + z^2}{2}u - \mu u + |u|^2 u = 0, \qquad u|_{\partial\Omega} = 0.$$

Solutions for $\mu = 6$.

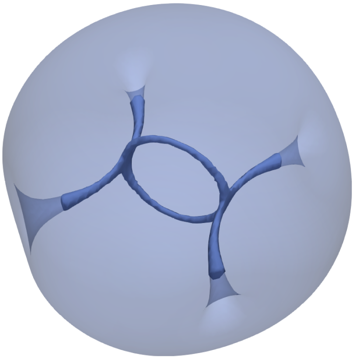


Four vortex lines of alternating charge.

Gross–Pitaevskii equation

$$-\frac{1}{2}\Delta u + \frac{x^2+y^2+z^2}{2}u - \mu u + |u|^2 u = 0, \qquad u|_{\partial\Omega} = 0.$$

Solutions for $\mu = 6$.

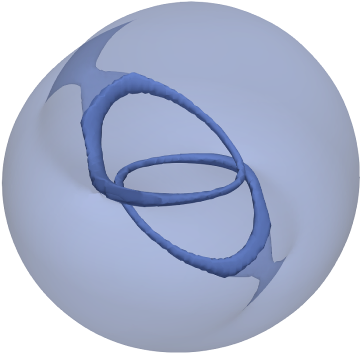


A vortex ring with two "handles".

Gross–Pitaevskii equation

$$-\frac{1}{2}\Delta u + \frac{x^2+y^2+z^2}{2}u - \mu u + |u|^2 u = 0, \qquad u|_{\partial\Omega} = 0.$$

Solutions for $\mu = 6$.

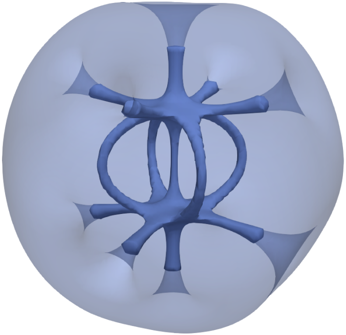


Two bent vortex rings?

Gross–Pitaevskii equation

$$-\frac{1}{2}\Delta u + \frac{x^2+y^2+z^2}{2}u - \mu u + |u|^2 u = 0, \qquad u|_{\partial\Omega} = 0.$$

Solutions for $\mu = 6$.

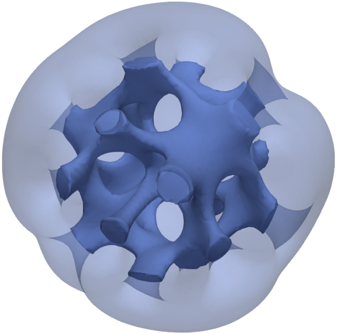


Two vortex rings and five lines?

Gross–Pitaevskii equation

$$-\frac{1}{2}\Delta u + \frac{x^2+y^2+z^2}{2}u - \mu u + |u|^2 u = 0, \qquad u|_{\partial\Omega} = 0.$$

Solutions for $\mu = 6$.



A vortex ring cage?

Section 7

Semismooth problems

Many problems feature inequality constraints.

Many problems feature inequality constraints.

The natural language for formulating these is as a *variational inequality*.

$\mathrm{VI}(Q, K)$

Let X be a real reflexive Banach space, $K \subset X$ a closed convex subset, and $Q : K \to X^*$. The task is to

find $u^\star \in K$ such that $\langle Q(u^\star), v - u^\star \rangle \geq 0$ for all $v \in K$.

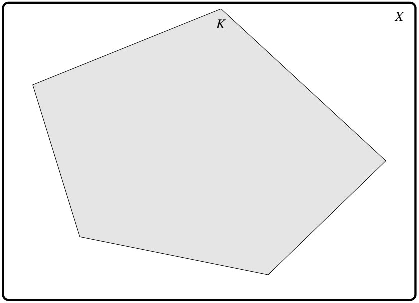
Many problems feature inequality constraints.

The natural language for formulating these is as a *variational inequality*.

$\mathrm{VI}(Q, K)$

Let X be a real reflexive Banach space, $K \subset X$ a closed convex subset, and $Q : K \to X^*$. The task is to

$$\text{find } u^\star \in K \text{ such that } \langle Q(u^\star), v - u^\star \rangle \geq 0 \text{ for all } v \in K.$$



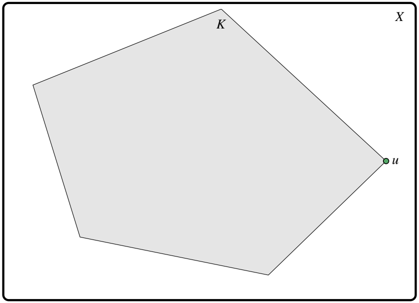
Many problems feature inequality constraints.

The natural language for formulating these is as a *variational inequality*.

$\mathrm{VI}(Q, K)$

Let X be a real reflexive Banach space, $K \subset X$ a closed convex subset, and $Q : K \to X^*$. The task is to

$$\text{find } u^\star \in K \text{ such that } \langle Q(u^\star), v - u^\star \rangle \geq 0 \text{ for all } v \in K.$$



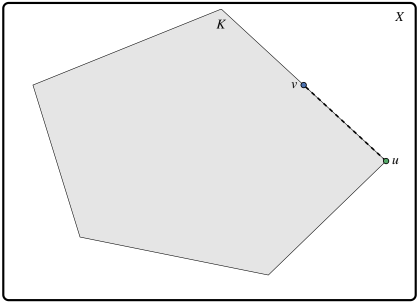
Many problems feature inequality constraints.

The natural language for formulating these is as a *variational inequality*.

$\mathrm{VI}(Q, K)$

Let X be a real reflexive Banach space, $K \subset X$ a closed convex subset, and $Q : K \to X^*$. The task is to

find $u^\star \in K$ such that $\langle Q(u^\star), v - u^\star \rangle \geq 0$ for all $v \in K$.



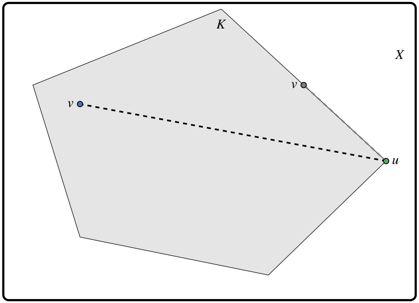
Many problems feature inequality constraints.

The natural language for formulating these is as a *variational inequality*.

$\mathrm{VI}(Q, K)$

Let X be a real reflexive Banach space, $K \subset X$ a closed convex subset, and $Q : K \to X^*$. The task is to

$$\text{find } u^\star \in K \text{ such that } \langle Q(u^\star), v - u^\star \rangle \geq 0 \text{ for all } v \in K.$$



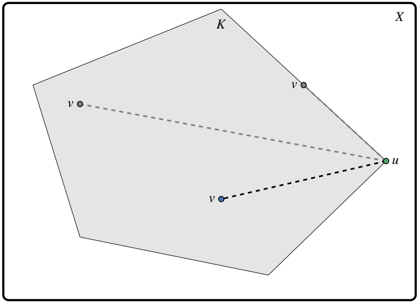
Many problems feature inequality constraints.

The natural language for formulating these is as a *variational inequality*.

$\mathrm{VI}(Q, K)$

Let X be a real reflexive Banach space, $K \subset X$ a closed convex subset, and $Q : K \to X^*$. The task is to

$$\text{find } u^\star \in K \text{ such that } \langle Q(u^\star), v - u^\star \rangle \geq 0 \text{ for all } v \in K.$$



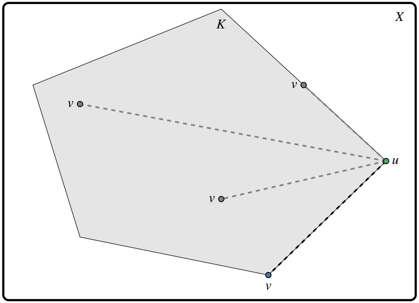
Many problems feature inequality constraints.

The natural language for formulating these is as a *variational inequality*.

$\mathrm{VI}(Q, K)$

Let X be a real reflexive Banach space, $K \subset X$ a closed convex subset, and $Q : K \to X^*$. The task is to

$$\text{find } u^\star \in K \text{ such that } \langle Q(u^\star), v - u^\star\rangle \geq 0 \text{ for all } v \in K.$$



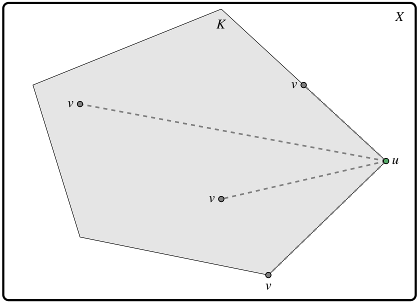
Many problems feature inequality constraints.

The natural language for formulating these is as a *variational inequality*.

$\mathrm{VI}(Q, K)$

Let X be a real reflexive Banach space, $K \subset X$ a closed convex subset, and $Q : K \to X^*$. The task is to

$$\text{find } u^\star \in K \text{ such that } \langle Q(u^\star), v - u^\star \rangle \geq 0 \text{ for all } v \in K.$$

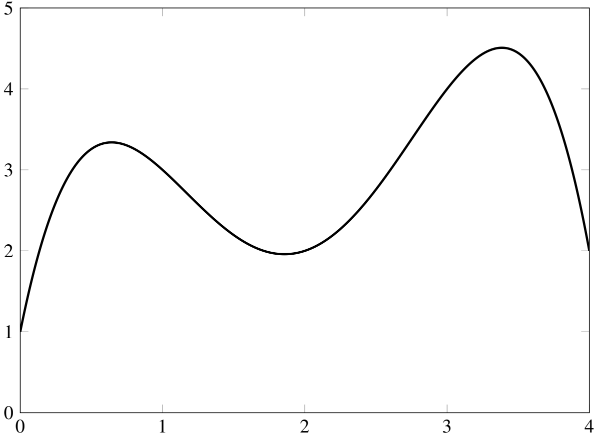


For example, if you want to minimise $f \in C^1(\mathbb{R}, \mathbb{R})$ over a closed interval $I \subset \mathbb{R}$, the necessary optimality condition is

$$\mathrm{VI}(f', I).$$

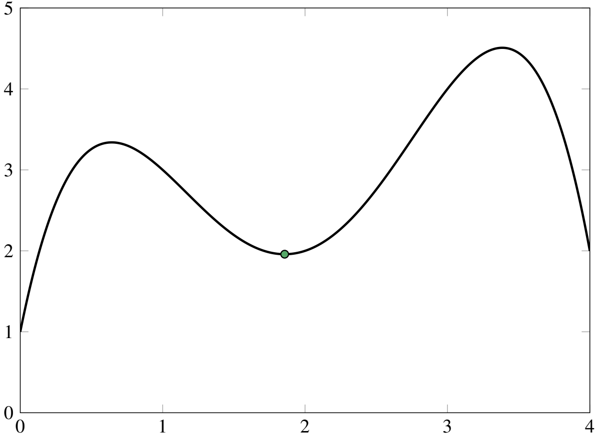
For example, if you want to minimise $f \in C^1(\mathbb{R}, \mathbb{R})$ over a closed interval $I \subset \mathbb{R}$, the necessary optimality condition is

$$\mathrm{VI}(f', I).$$



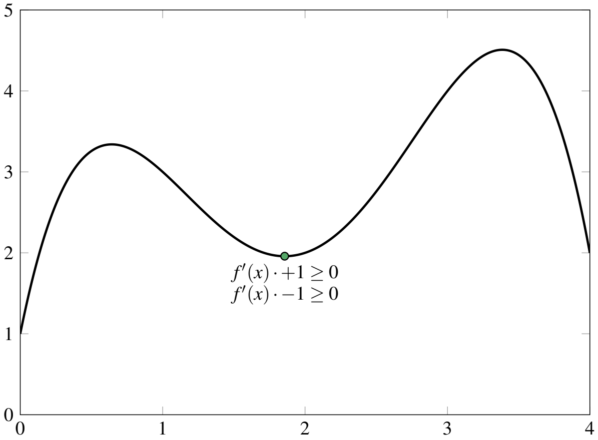
For example, if you want to minimise $f \in C^1(\mathbb{R}, \mathbb{R})$ over a closed interval $I \subset \mathbb{R}$, the necessary optimality condition is

$$\mathrm{VI}(f', I).$$



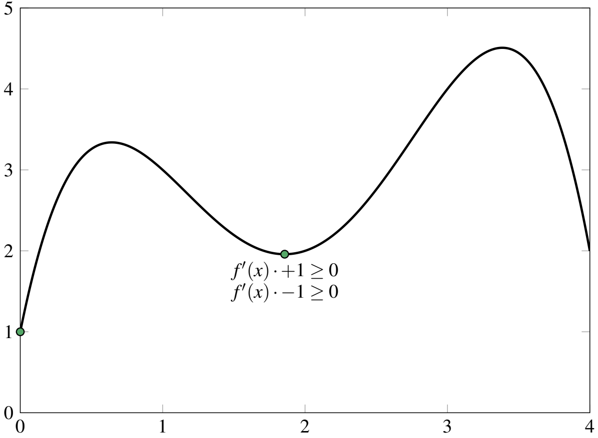
For example, if you want to minimise $f \in C^1(\mathbb{R}, \mathbb{R})$ over a closed interval $I \subset \mathbb{R}$, the necessary optimality condition is

$$\mathrm{VI}(f', I).$$



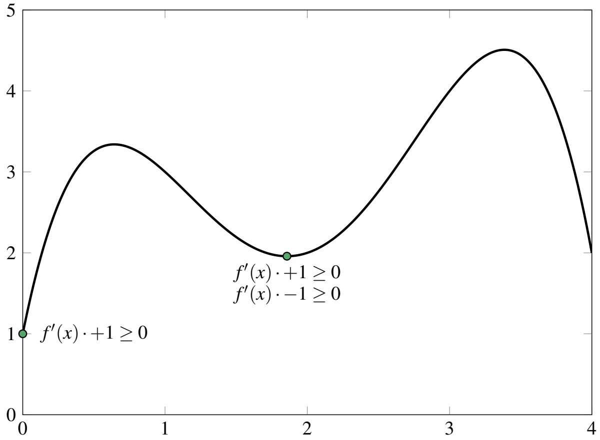
For example, if you want to minimise $f \in C^1(\mathbb{R}, \mathbb{R})$ over a closed interval $I \subset \mathbb{R}$, the necessary optimality condition is

$$\mathrm{VI}(f', I).$$



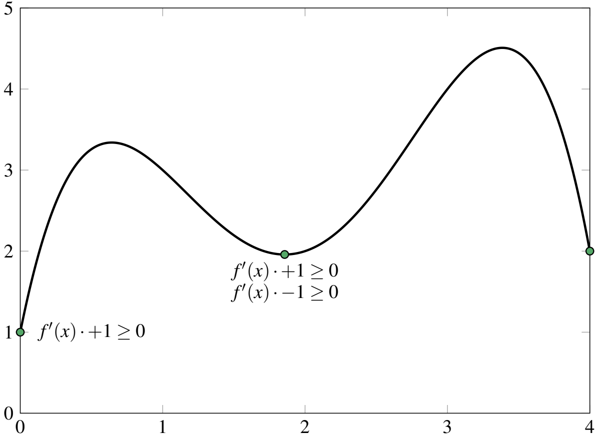
For example, if you want to minimise $f \in C^1(\mathbb{R}, \mathbb{R})$ over a closed interval $I \subset \mathbb{R}$, the necessary optimality condition is

$$\mathrm{VI}(f', I).$$



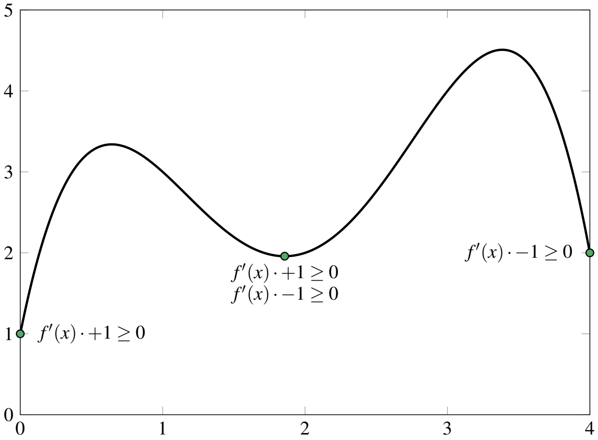
For example, if you want to minimise $f \in C^1(\mathbb{R}, \mathbb{R})$ over a closed interval $I \subset \mathbb{R}$, the necessary optimality condition is

$$\mathrm{VI}(f', I).$$



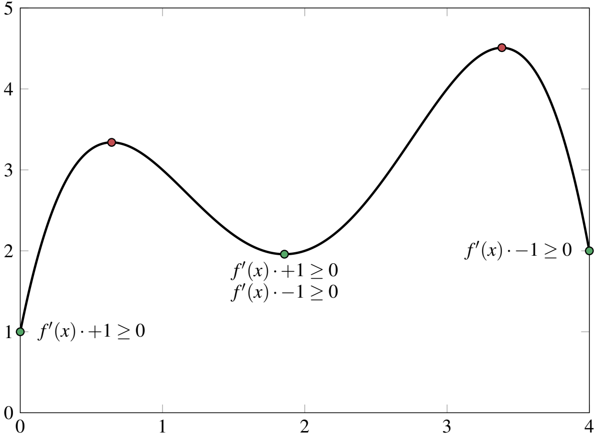
For example, if you want to minimise $f \in C^1(\mathbb{R}, \mathbb{R})$ over a closed interval $I \subset \mathbb{R}$, the necessary optimality condition is

$$\mathrm{VI}(f', I).$$



For example, if you want to minimise $f \in C^1(\mathbb{R}, \mathbb{R})$ over a closed interval $I \subset \mathbb{R}$, the necessary optimality condition is

$$\mathrm{VI}(f', I).$$



The main way of solving variational inequalities is to reformulate them as a system of equations.

The main way of solving variational inequalities is to reformulate them as a system of equations.

For example, $\mathrm{VI}(Q, K)$ with

$$K = \{x \in \mathbb{R} : x \geq 0\}$$

is equivalent to

$$S(x) \coloneqq \sqrt{x^2 + [Q(x)]^2} - x - Q(x) = 0.$$

The main way of solving variational inequalities is to reformulate them as a system of equations.

For example, $\mathrm{VI}(Q, K)$ with

$$K = \{x \in \mathbb{R} : x \geq 0\}$$

is equivalent to

$$S(x) \coloneqq \sqrt{x^2 + [Q(x)]^2} - x - Q(x) = 0.$$

The price we pay ...

... is that S is not smooth.

Good news

S is *just smooth enough* to define a Newton-type method with superlinear convergence.

Michael Hintermüller, 1970–

Michael Ulbrich, 1967–

Good news

S is *just smooth enough* to define a Newton-type method with superlinear convergence.

Semismoothness

Let X and Y be Banach spaces. Let $S : \Omega \subset X \to Y$, where Ω is an open subset of X. S is semismooth at $u \in \Omega$ if it is locally Lipschitz continuous at u and there exists an open neighbourhood $N \subset \Omega$ containing u with a *Newton derivative*, i.e. a mapping $H : \Omega \to L(X, Y)$ with the property that

$$S(u + h) - S(u) - H(u + h)h = o(h)$$

for all u in N.

Michael Hintermüller, 1970–

Michael Ulbrich, 1967–

Good news

S is *just smooth enough* to define a Newton-type method with superlinear convergence.

Semismooth Newton works just like normal:

$$u_{i+1} = u_i - [H(u_i)]^{-1} S(u_i),$$

where H is the Newton derivative.

This algorithm usually converges superlinearly.

Michael Hintermüller, 1970–

Michael Ulbrich, 1967–

Good news

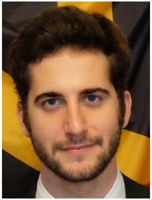
Deflation works for semismooth problems.

Good news

Deflation works for semismooth problems.

Theorem (F., Croci, Surowiec, 2020)

Under the same assumptions that are required for superlinear convergence of semismooth Newton, deflation works the same.



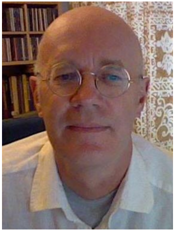
Matteo Croci, 1992–

Thomas Surowiec, 1982–

Gould gives an example where the central path is ill-behaved:

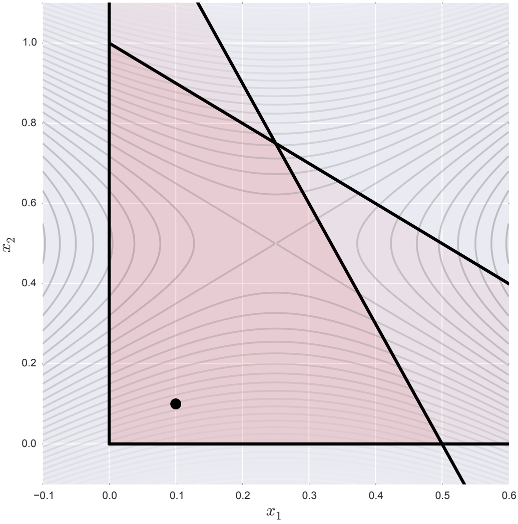
Nonconvex quadratic programming problem

$$
\begin{aligned}
\underset{x \in \mathbb{R}^2}{\text{minimise}} \quad & -2(x_1 - 0.25)^2 + 2(x_2 - 0.5)^2 \\
\text{subject to} \quad & x_1 + x_2 \leq 1 \\
& 3x_1 + x_2 \leq 1.5 \\
& x_1 \geq 0 \\
& x_2 \geq 0
\end{aligned}
$$



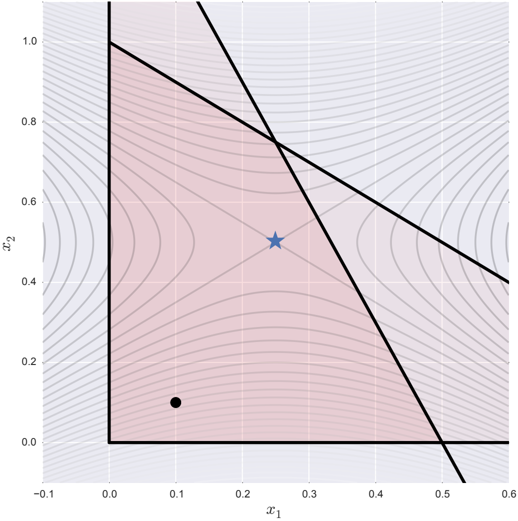
Nick Gould, 1957–

Gould gives an example where the central path is ill-behaved:



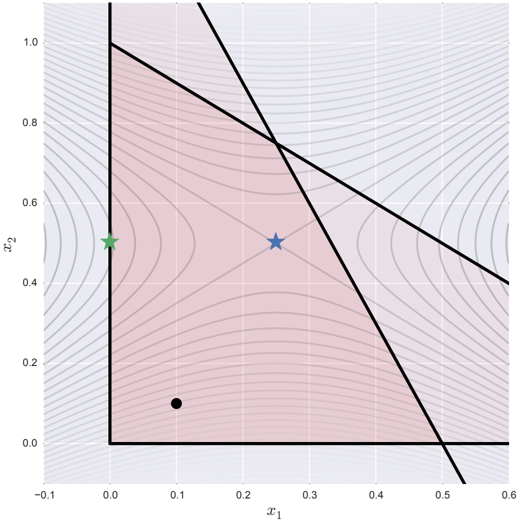
Deflation finds both minima and the saddle point.

Gould gives an example where the central path is ill-behaved:



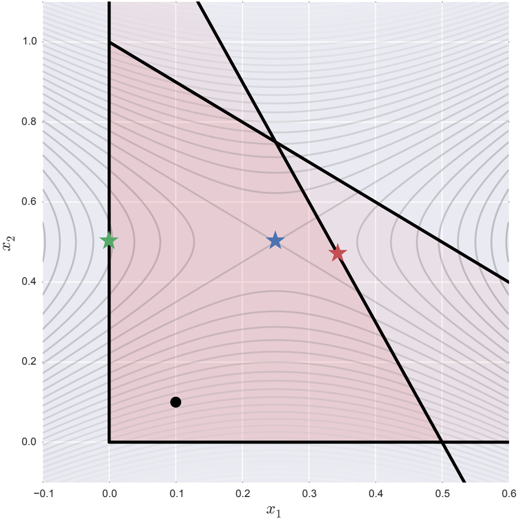
Deflation finds both minima and the saddle point.

Gould gives an example where the central path is ill-behaved:



Deflation finds both minima and the saddle point.

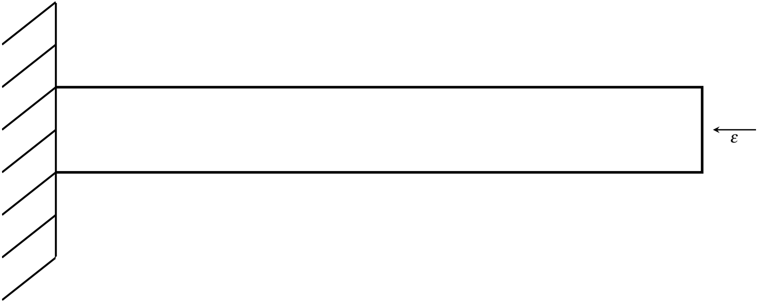
Gould gives an example where the central path is ill-behaved:



Deflation finds both minima and the saddle point.

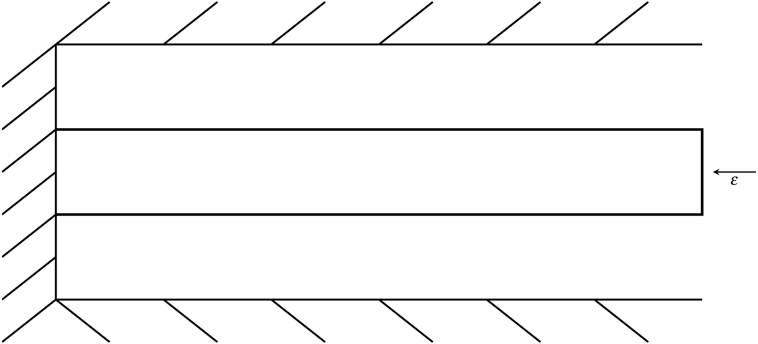
Buckling of a hyperelastic beam with contact constraints

$$\begin{aligned} \underset{u \in H^1(\Omega;\mathbb{R}^2)}{\text{minimise}} \quad & \Pi(u) = \int_\Omega \psi(u)\ \mathrm{d}x - \int_\Omega B \cdot u\ \mathrm{d}x \\ \text{subject to} \quad & u|_{\text{left}} = (0,0),\ \ u|_{\text{right}} = (-\varepsilon, 0), \\ & \mathrm{tr}(u_y) \in [a,b] \text{ a.e. in } \Gamma_{\text{top}}, \Gamma_{\text{bottom}}. \end{aligned}$$



Buckling of a hyperelastic beam with contact constraints

$$\begin{aligned} \underset{u \in H^1(\Omega;\mathbb{R}^2)}{\text{minimise}} \quad & \Pi(u) = \int_\Omega \psi(u)\ \mathrm{d}x - \int_\Omega B \cdot u \ \mathrm{d}x \\ \text{subject to} \quad & u|_{\text{left}} = (0,0),\ \ u|_{\text{right}} = (-\varepsilon, 0), \\ & \operatorname{tr}(u_y) \in [a,b] \text{ a.e. in } \Gamma_{\text{top}}, \Gamma_{\text{bottom}}. \end{aligned}$$



Buckling of a hyperelastic beam with contact constraints

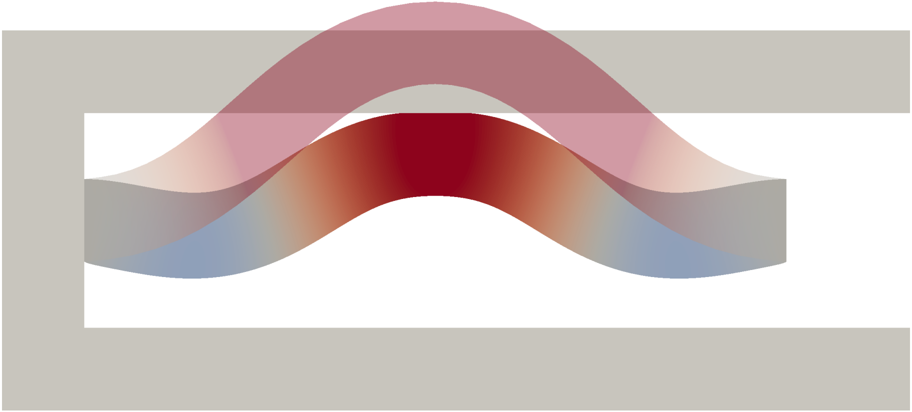
$$\begin{aligned} \underset{u \in H^1(\Omega;\mathbb{R}^2)}{\text{minimise}} \quad & \Pi(u) = \int_\Omega \psi(u)\ \mathrm{d}x - \int_\Omega B \cdot u\ \mathrm{d}x \\ \text{subject to} \quad & u|_{\text{left}} = (0,0),\ \ u|_{\text{right}} = (-\varepsilon, 0), \\ & \operatorname{tr}(u_y) \in [a,b] \text{ a.e. in } \Gamma_{\text{top}}, \Gamma_{\text{bottom}}. \end{aligned}$$

Neo–Hookean strain energy density

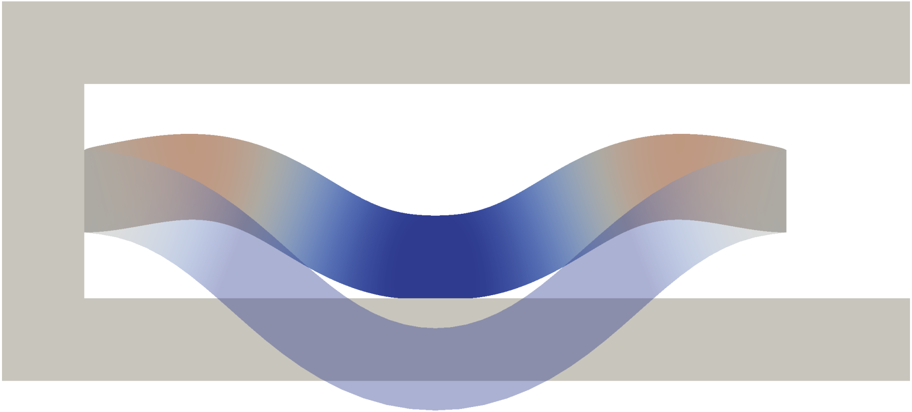
$$\psi(u) = \frac{\mu}{2}(\operatorname{tr}(C) - 2) - \mu \log(\det(C)) + \frac{\lambda}{2}\log(\det(C))^2,$$

where

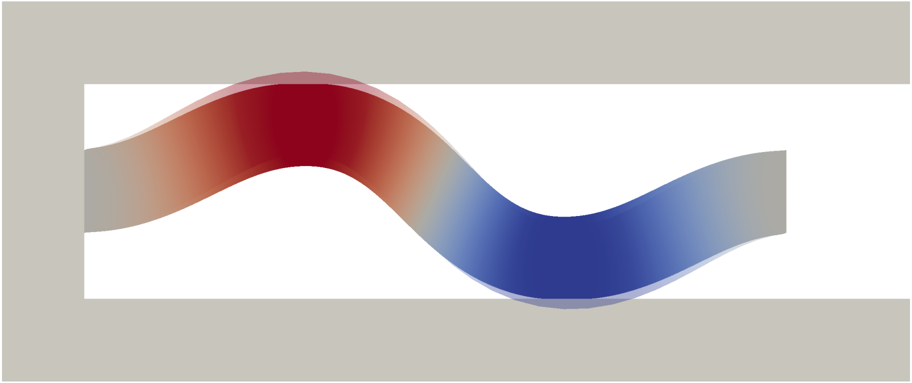
$$C = (I + \nabla u)^\top (I + \nabla u).$$



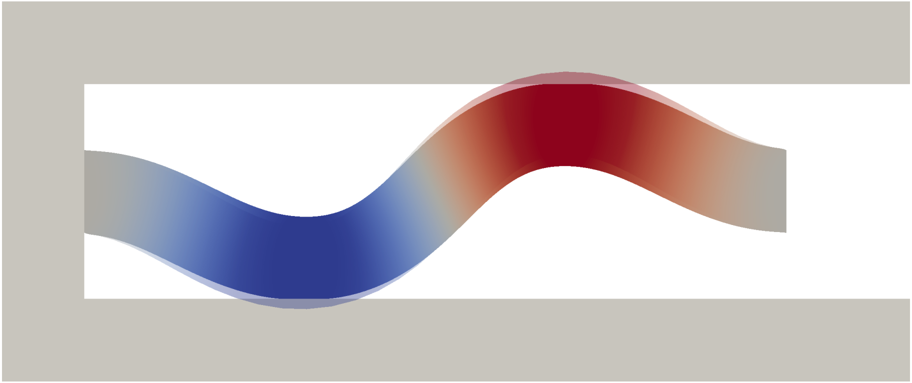
Multiple solutions of the beam with contact constraints.



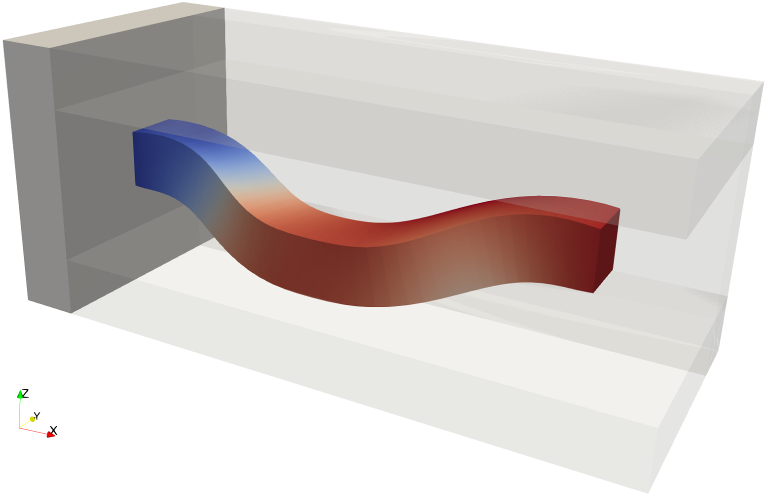
Multiple solutions of the beam with contact constraints.



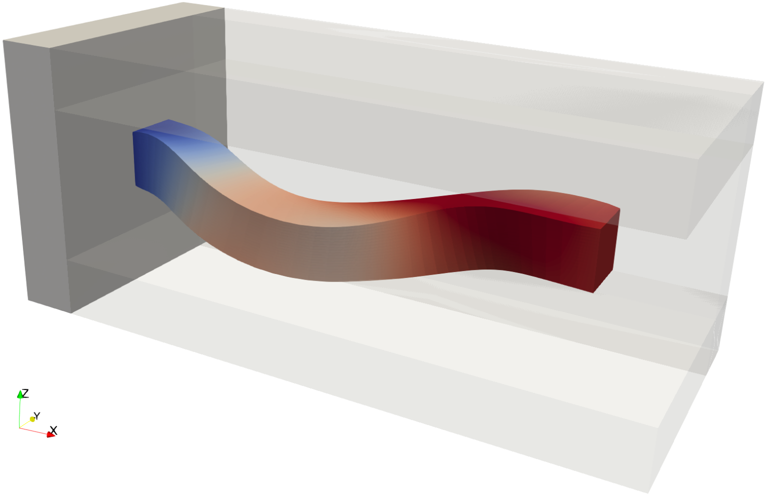
Multiple solutions of the beam with contact constraints.



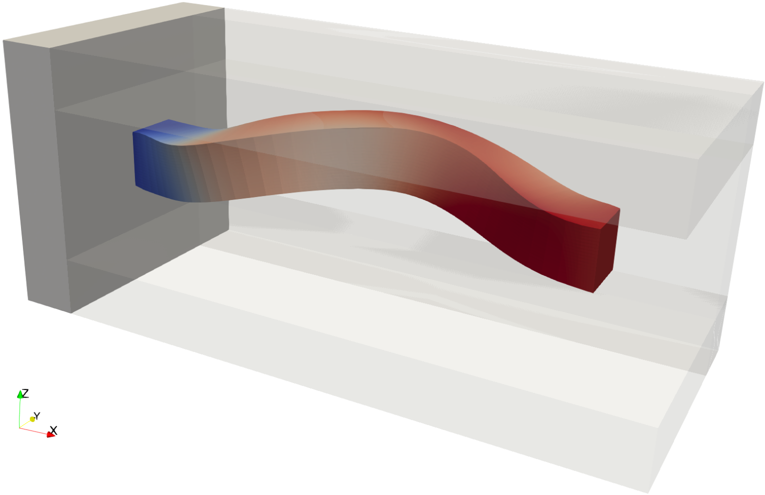
Multiple solutions of the beam with contact constraints.



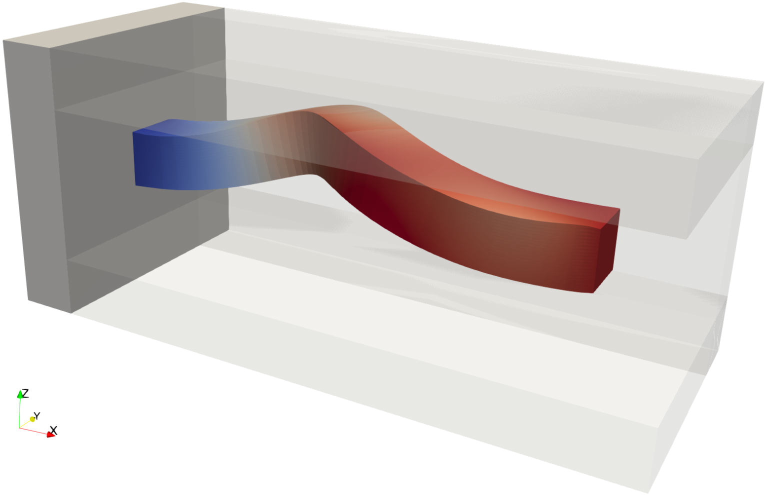
Multiple solutions of the beam with contact constraints.



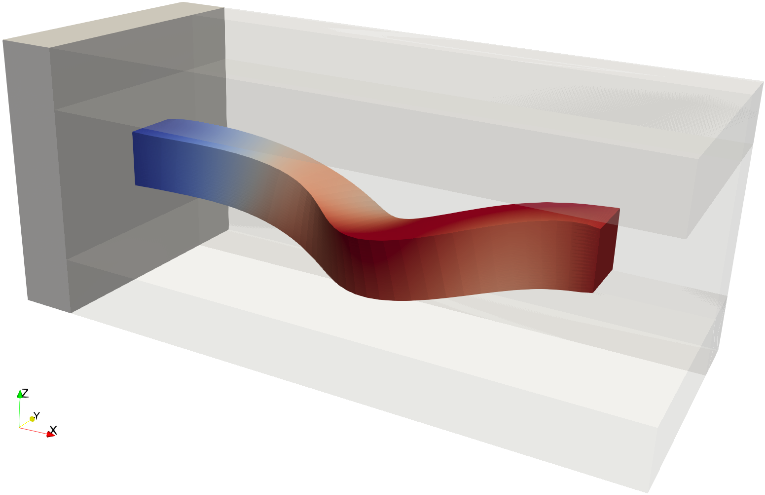
Multiple solutions of the beam with contact constraints.



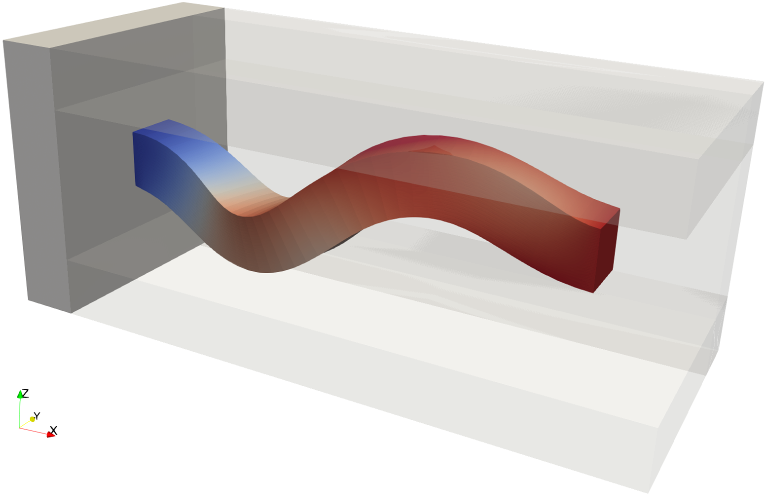
Multiple solutions of the beam with contact constraints.



Multiple solutions of the beam with contact constraints.



Multiple solutions of the beam with contact constraints.



Multiple solutions of the beam with contact constraints.

Conclusions!

Conclusions!

Main message

When solving nonlinear problems, think about multiple solutions!

Conclusions!

Main message

When solving nonlinear problems, think about multiple solutions!

Algorithms

We now have very powerful algorithms for numerical bifurcation analysis.

Open questions!

Open questions!

How do we apply classical algorithms at very large scale?

Open questions!

How do we apply classical algorithms at very large scale?

How should we best combine deflation and classical algorithms?

Open questions!

How do we apply classical algorithms at very large scale?

How should we best combine deflation and classical algorithms?

What does bifurcation analysis for nonsmooth systems look like?

Open questions!

How do we apply classical algorithms at very large scale?

How should we best combine deflation and classical algorithms?

What does bifurcation analysis for nonsmooth systems look like?

How can we robustly deal with general symmetry groups?

Open questions!

How do we apply classical algorithms at very large scale?

How should we best combine deflation and classical algorithms?

What does bifurcation analysis for nonsmooth systems look like?

How can we robustly deal with general symmetry groups?

Thank you

to Josef, the organisers, and all the participants!