Bifurcation analysis via neural networks

Prepared by Zinkin Zakhar
Academic Supervisor: Tomaschuk Korney

Definition

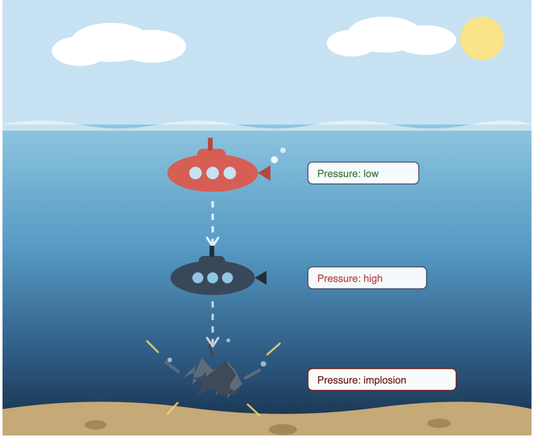
A **bifurcation points** is a value of the parameter λ of the system at which a small smooth change in λ produces a qualitative change in the system's behaviour.

Problem statement

A submarine's shell is described as a solution to some PDE.

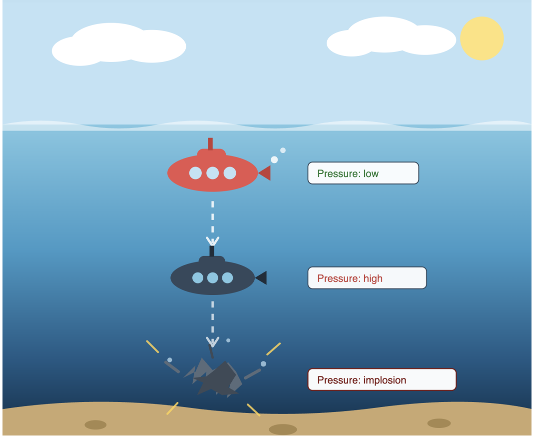
Problem statement

A submarine's shell is described as a solution to some PDE.



Problem statement

A submarine's shell is described as a solution to some PDE.



How do small changes of the parameter (e.g. pressure) affect the solution?

Problem statement

Given a differential equation (ODE or PDE):

$$\begin{cases} u'' = \lambda e^{u} \\ u(0) = u(1) = 0 \end{cases}$$

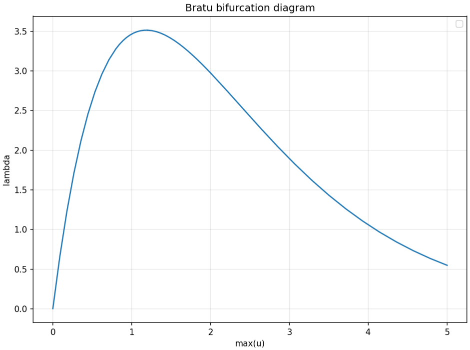
We want to graph $(\lambda, ||u||)$ for all λ.

Problem statement

Given a differential equation (ODE or PDE):

$$\begin{cases} u'' = \lambda e^u \\ u(0) = u(1) = 0 \end{cases}$$

We want to graph $(\lambda, ||u||)$ for all λ.



Problem statement

How to solve the problem?

Problem statement

How to solve the problem?

1. Solve DE for a fixed λ.

Problem statement

How to solve the problem?

1. Solve DE for a fixed λ.
2. Analyze bifurcation points.

Problem statement

How to solve the problem?

1. Solve DE for a fixed λ. (Numeric or Neural Network)
2. Analyze bifurcation points. (Theory)

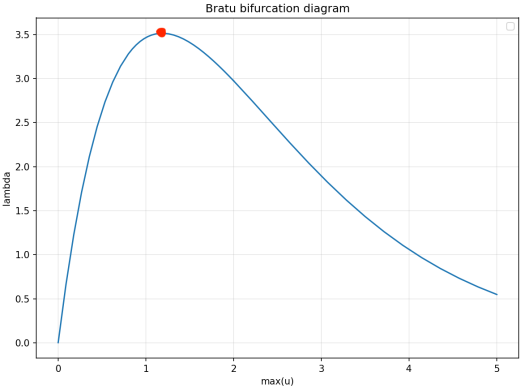
How to solve the problem?

- Apply **spatial discretization**, get $F(u, \lambda) = 0,\ u \in \mathbb{R}^N,\ \lambda \in \mathbb{R}$. Solve using **Newton's method**:

$$u^{(k+1)} = u^{(k)} - J_F\left(u^{(k)}, \lambda\right)^{-1} F\left(u^{(k)}, \lambda\right),$$

- **Parameter continuation**: given $(u^{[k-1]}, \lambda^{[k-1]})$, set $\lambda^{[k]} = \lambda^{[k-1]} + \Delta\lambda$, Newton's method with $u^{[k]}$ as init.

Works well until a bifurcation point.



How to solve the problem?

- To detect bifurcation points:

$$\tau(u, \lambda) = \det J_F(u, \lambda).$$

  Sign changes or $\approx 0 \implies$ bifurcation point.
- Two types of bifurcation points: **fold**, **transcritical** and **pitchfork**.

Why switch to Neural Networks?

Why switch? The curse of dimensionality

Discretization gives a system of size

$$N = O(M^d).$$

- d — spatial dimension, M — grid points per axis.
- Finer mesh $\Rightarrow$ N grows exponentially in d.

Why switch? The curse of dimensionality

Discretization gives a system of size

$$N = O(M^d).$$

- d — spatial dimension, M — grid points per axis.
- Finer mesh $\Rightarrow$ N grows exponentially in d.

Neural networks are mesh-free.

PINN

Sample $\{x_f^i\}$ interior points on Ω, $\{x_u^i\}$ on $\partial\Omega$, approximate the solution by a network u_θ.

PINN

Sample $\{x_f^i\}$ interior points on Ω, $\{x_u^i\}$ on $\partial\Omega$, approximate the solution by a network u_θ.

$$\mathcal{L} = \mathcal{L}_{\mathrm{DE}} + \mathcal{L}_{\mathrm{BC}}$$

PINN

Sample $\{x_f^i\}$ interior points on Ω, $\{x_u^i\}$ on $\partial\Omega$, approximate the solution by a network u_θ.

$$\mathcal{L} = \mathcal{L}_{\mathrm{DE}} + \mathcal{L}_{\mathrm{BC}}$$

$$\mathcal{L}_{\mathrm{DE}} = \frac{1}{N_f} \sum_{i=1}^{N_f} \left(u_\theta(x_f^i)\right)^2, \qquad \mathcal{L}_{\mathrm{BC}} = \frac{1}{N_u} \sum_{i=1}^{N_u} \left(u_\theta(x_u^i) - u^i\right)^2$$

Pseudo Arc-Length Continuation PINN

Existing approach:

Pseudo Arc-Length Continuation PINN

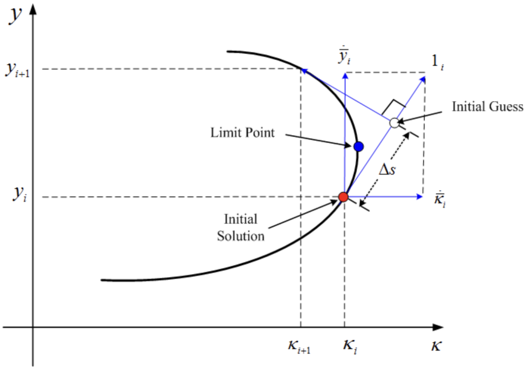
Existing approach:

$$\mathcal{L} = \mathcal{L}_{\mathrm{DE}} + \mathcal{L}_{\mathrm{BC}} + \mu\sqrt{[\beta_1(\|u^{[k]}\| - \|u^{[k-1]}\| - \gamma)]^2 + [\beta_2(\lambda^{[k]} - \lambda^{[k-1]})]^2} - \delta$$

Pseudo Arc-Length Continuation PINN

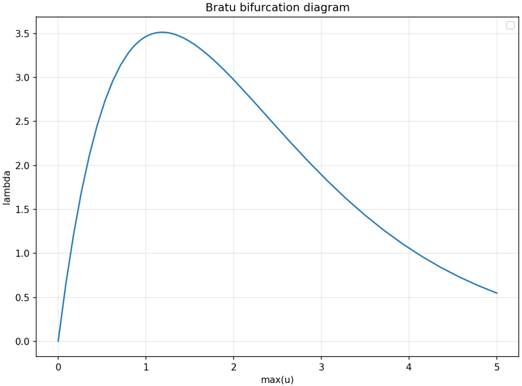
Existing approach:

$$\mathcal{L} = \mathcal{L}_{\mathrm{DE}} + \mathcal{L}_{\mathrm{BC}} + \mu\sqrt{[\beta_1(\|u^{[k]}\| - \|u^{[k-1]}\| - \gamma)]^2 + [\beta_2(\lambda^{[k]} - \lambda^{[k-1]})]^2} - \delta$$



Two new methods.

Method 1. Forced Solution Norm PINN



Method 1. Forced Solution Norm PINN

Penalize for not reaching a known norm.

Method 1. Forced Solution Norm PINN

Penalize for not reaching a known norm.

$$\mathcal{L} = \mathcal{L}_{\mathrm{DE}} + \mathcal{L}_{\mathrm{BC}} + \mu \, \mathrm{ReLU}(\mathrm{threshold} - \|u\|)^2$$

Method 1. Forced Solution Norm PINN

Penalize for not reaching a known norm.

$$\mathcal{L} = \mathcal{L}_{\mathrm{DE}} + \mathcal{L}_{\mathrm{BC}} + \mu\, \mathrm{ReLU}(\mathrm{threshold} - \|u\|)^2$$

To cross the point: $\mathrm{threshold} = C \max_i \|u_i\|, \quad C > 1.$

Method 2. Solution Norm as Input PINN

Method 2. Solution Norm as Input PINN

The map $\lambda \mapsto u$ is not single-valued.

Method 2. Solution Norm as Input PINN

The map $\lambda \mapsto u$ is not single-valued.

The map $\|u\| \mapsto (u, \lambda)$ is single-valued (on a part of a diagram).

Method 2. Solution Norm as Input PINN

Target norm T:

$$\begin{cases} N[u;\lambda] = 0 \\ B[u] = 0 \\ \|u\| = T \end{cases}$$

Method 2. Solution Norm as Input PINN

Target norm T:

$$\begin{cases} N[u;\lambda] = 0 \\ B[u] = 0 \\ \|u\| = T \end{cases}$$

$$T \mapsto (u, \lambda)$$

Method 2. Solution Norm as Input PINN

Target norm T:

$$\begin{cases} N[u;\lambda] = 0 \\ B[u] = 0 \\ \|u\| = T \end{cases}$$

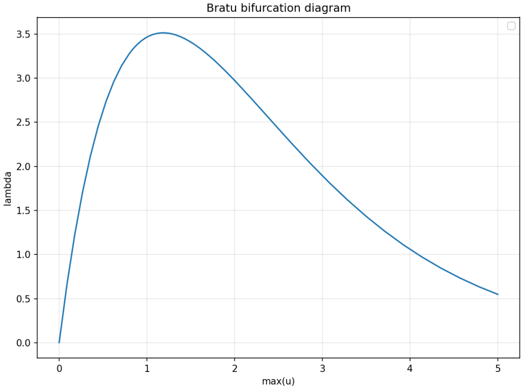
$$T \mapsto (u, \lambda)$$

$$\mathcal{L} = \mathcal{L}_{\mathrm{DE}} + \mathcal{L}_{\mathrm{BC}} + \mu\big(\|u_\theta\| - T\big)^2$$

Results: 1D Bratu

$$\begin{cases} u'' + \lambda e^u = 0, \\ u(0) = u(1) = 0. \end{cases}$$

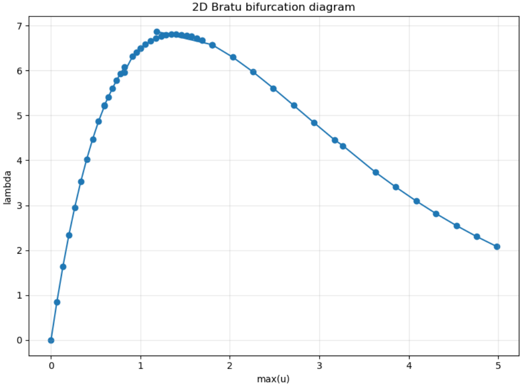
Arc-Length	Forced Norm	Norm as Input
1/10	✓	✓



Results: 1D Bratu

$$\begin{cases} \nabla^2 u + \lambda e^u = 0, \\ u(0, y) = u(x, 0) = 0. \end{cases}$$

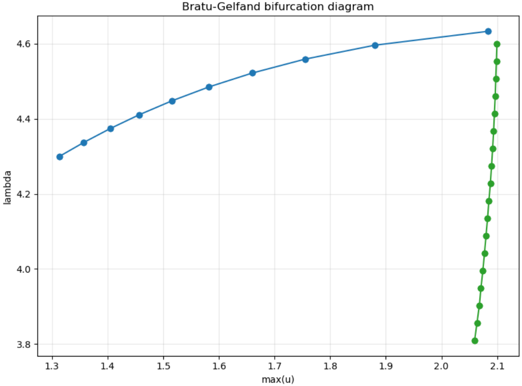
Arc-Length	Forced Norm	Norm as Input
$\times$	$\checkmark$	$\checkmark$



Results: 1D Gelfand-Bratu

$$\begin{cases} u'' + \lambda \exp\left(\dfrac{u}{1+\varepsilon u}\right) = 0, \\ u(0) = u(1) = 0, \end{cases}$$

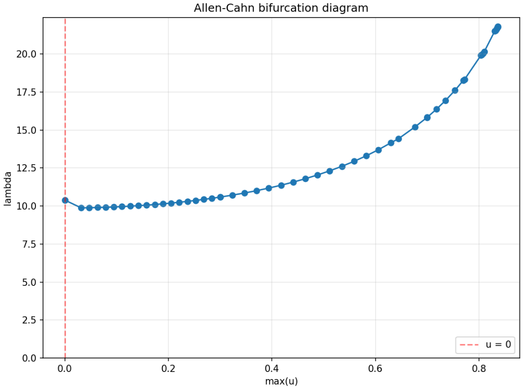
Forced Norm	Norm as Input
$\times$	$\checkmark$



Results: 1D Allen-Cahn

$$\begin{cases} u'' + \lambda(u - u^3) = 0, \\ u(0) = u(1) = 0. \end{cases}$$

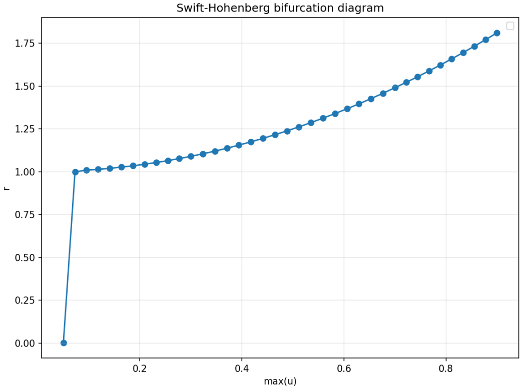
Norm as Input
✓



Results: 1D SwiftHohenberg

$$\begin{cases} -(1+\partial_x^2)^2 u + ru - u^3 = 0, \\ u \text{ periodic on } \partial\Omega. \end{cases}$$

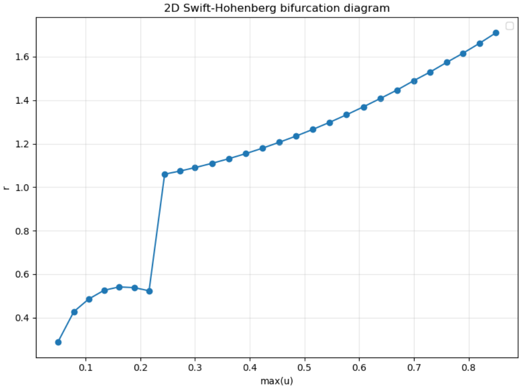
Norm as Input
✓



Results: 2D SwiftHohenberg

$$\begin{cases} -(1+\nabla^2)^2 u + ru - u^3 = 0, & x \in \Omega, \\ u \text{ periodic on } \partial\Omega. \end{cases}$$

Norm as Input
+-



The goal: von Kármán Cylindrical Shell

Kármán equations for a cylindrical shell (Donnell form):

$$D\nabla^4 w = \varphi_{yy} w_{xx} - 2\varphi_{xy} w_{xy} + \varphi_{xx} w_{yy} - \gamma\, \varphi_{xx} - \lambda,$$
$$\nabla^4 \varphi = -\left(w_{xy}^2 - w_{xx} w_{yy}\right) + \gamma\, w_{xx},$$

$$D = \frac{1}{12(1-\mu^2)}, \quad \nabla^4 = \partial_x^4 + 2\partial_x^2\partial_y^2 + \partial_y^4, \quad \gamma = \tfrac{R}{H}.$$

$\lambda =$ const — bifurcation parameter (pressure). First bifurcation $\lambda_{\mathrm{crit}} \approx 550$.

von Kármán: outcome

- No architecture converged to a good solution.
- Hypothesis (no extra physical input needed) is false.

Conclusion

- Summary of required theory for bifurcation analysis.

Conclusion

- Summary of required theory for bifurcation analysis.
- Discussion of existing methods with cited bibliography for further reading.

Conclusion

- Summary of required theory for bifurcation analysis.
- Discussion of existing methods with cited bibliography for further reading.
- Two new PINN methods for bifurcation diagrams.

Conclusion

- Summary of required theory for bifurcation analysis.
- Discussion of existing methods with cited bibliography for further reading.
- Two new PINN methods for bifurcation diagrams.
- Norm as Input — robust on many problems.

Conclusion

- Summary of required theory for bifurcation analysis.
- Discussion of existing methods with cited bibliography for further reading.
- Two new PINN methods for bifurcation diagrams.
- Norm as Input — robust on many problems.
- von Kármán goal not constructed: documented why and what was tried.