TOWSON UNIVERSITY

SWIMMING WITH DEEP LEARNING

Kristin Lloyd, Garrett Hauser, Jazmin Sharp,
Samuel Armstrong, Dante Buhl

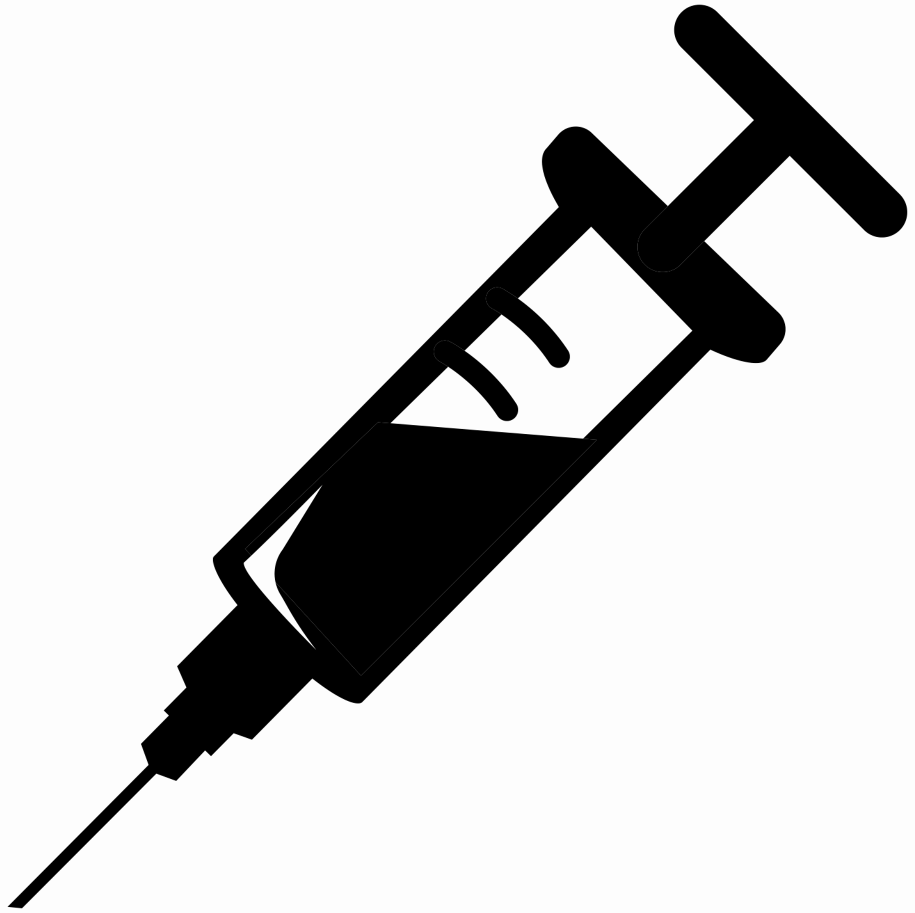
Research Experience for Undergraduates

MOTIVATION

“Artificial microswimmers offer exciting opportunities for biomedical applications. The goal of these synthetics is to swim like natural microorganisms through biological environments and perform complex tasks such as drug delivery”

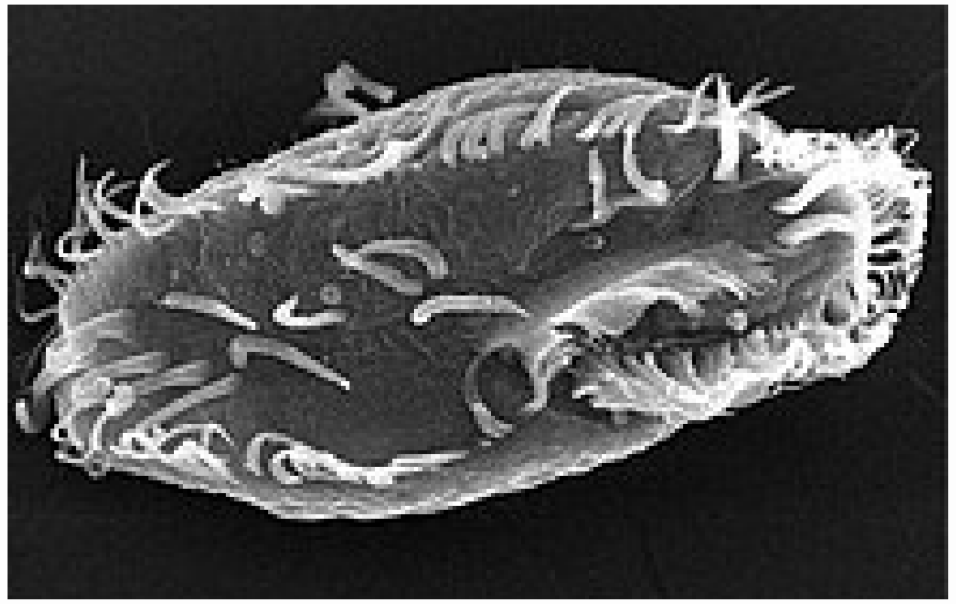
Roads to Smart Artificial Microswimmers

Alan C. H. Tsang, Ebru Demir, Yang Ding,*
and On Shun Pak*

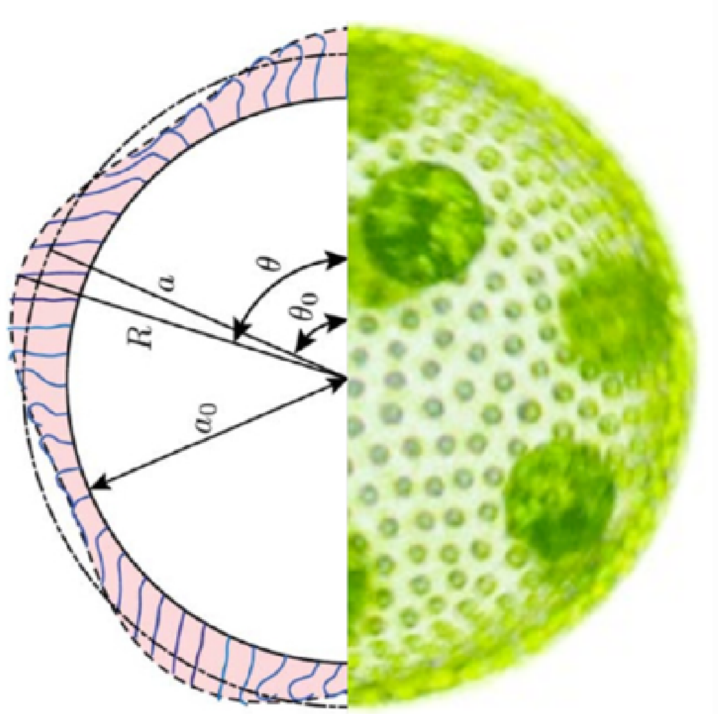


MOTIVATION

- We investigate the motion of ciliated microorganisms in Newtonian fluids
- We use physics-informed neural networks (PINNs) to simulate these microorganism propulsion mechanisms
- Ultimately, PINNs will be used to investigate the propulsion of microorganisms in more complex fluids, where exact solutions do not exist



PROBLEM SET UP



Surface Velocity Distribution

$$\vec{u} = \sum_{n=0}^{\infty} A_n P_n(\cos\theta)\,\vec{e}_r + \sum_{n=1}^{\infty} \frac{-2B_n}{n(n+1)} P_n^1(\cos\theta)\,\vec{e}_\theta$$

Lighthill, *Comm.Pure Appl. Math.* (1952)

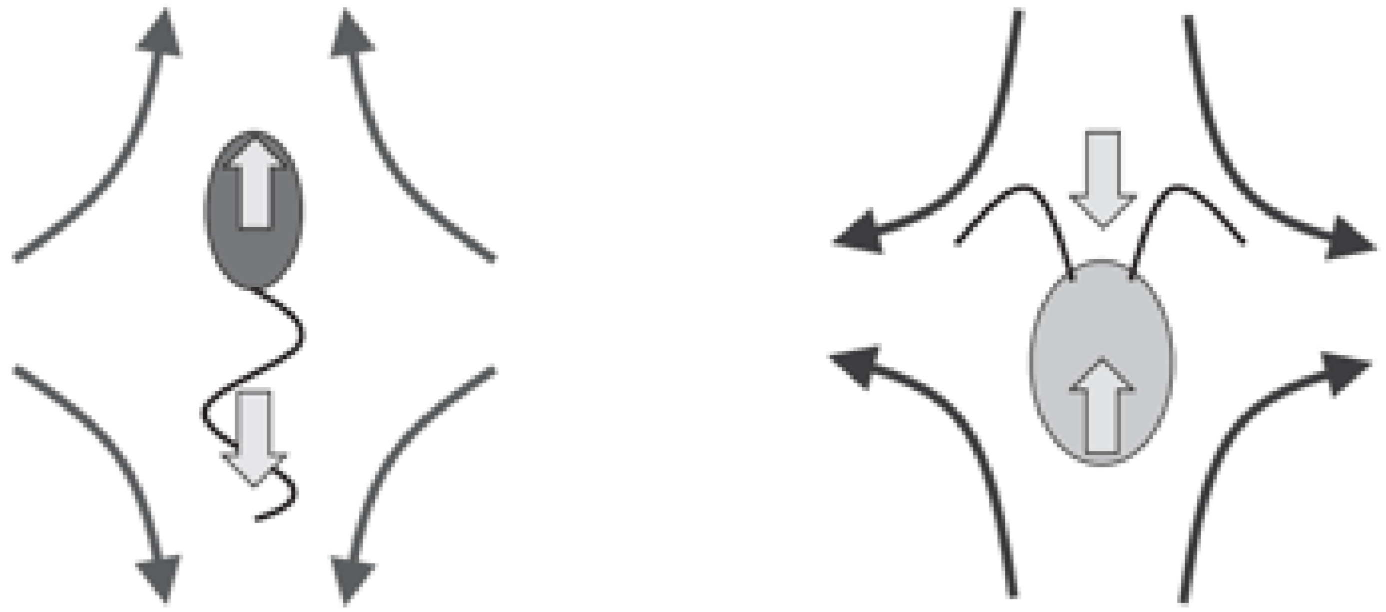
Blake, *J. Fluid Mech.* (1971)

$\alpha = B_2/B_1$

$\alpha = 0 \implies$ Neutral squirmer

- An and Bn are swimming modes
- Pn and Pn1 are Legendre and associated Legendre polynomials

$\alpha < 0 \implies$ Pusher $\alpha > 0 \implies$ Puller



E.g. *E coli* E.g. *Chlamydomonas*

Ishikawa, *J. R. Soc. Interface* (2009)

GOVERNING EQUATIONS

- Ciliated microorganisms navigate through environments characterized by low Reynolds numbers, which essentially means they move through highly viscous fluids

Stokes Equation

$$-\nabla p + \mu \nabla^2 \vec{v} = \vec{0}$$

Continuity Equation

$$\nabla \cdot \vec{u} = 0$$

- To describe our problem's geometry, we use spherical coordinates (r, θ, Φ)

GOVERNING EQUATIONS

The boundary conditions are given by

$$\vec{\mathrm{u}}(r=1) = \vec{\mathrm{u}}_{sq} + \vec{U}$$
$$\vec{\mathrm{u}}(r \to \infty) = \vec{0}$$

where the propulsion speed U is obtained by using the force free condition

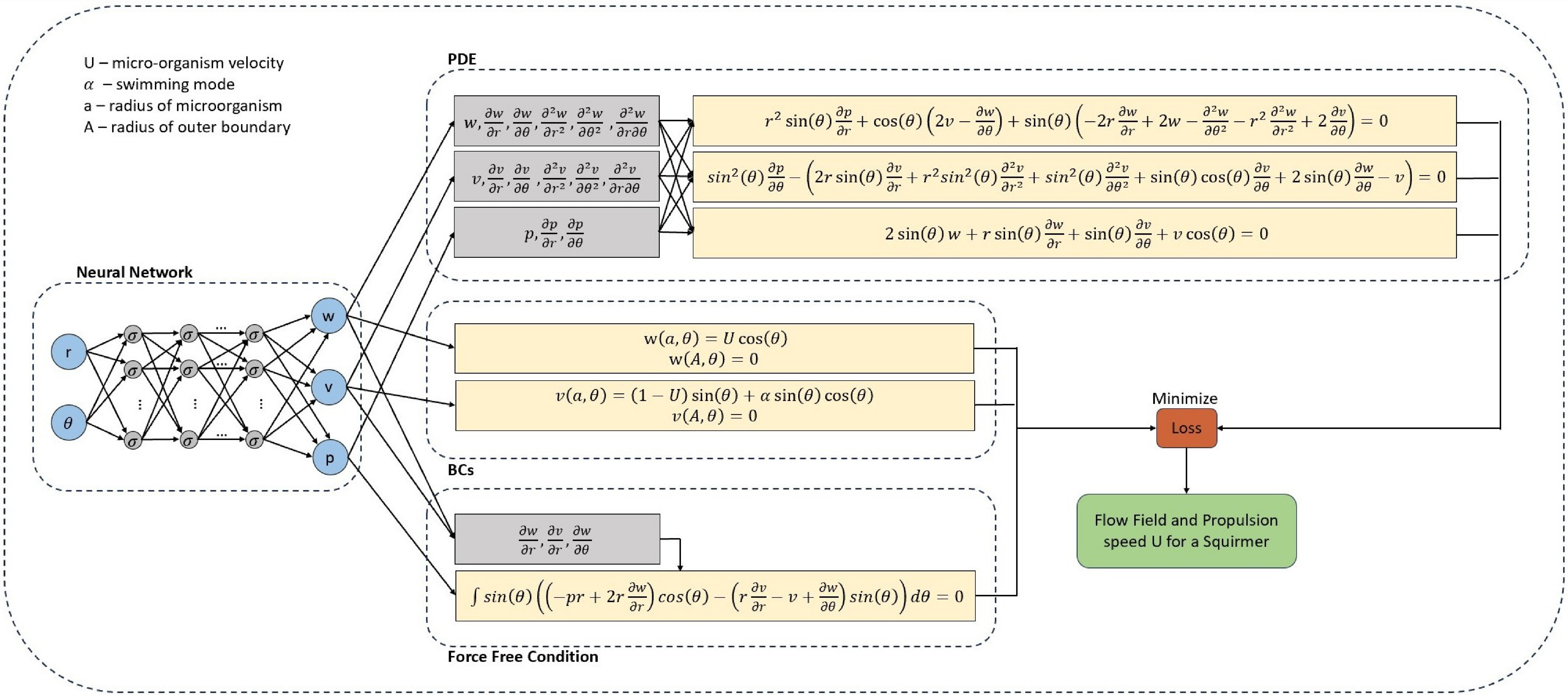
$$\int_S \sigma \cdot \boldsymbol{n} \, \mathrm{d}S = 0$$

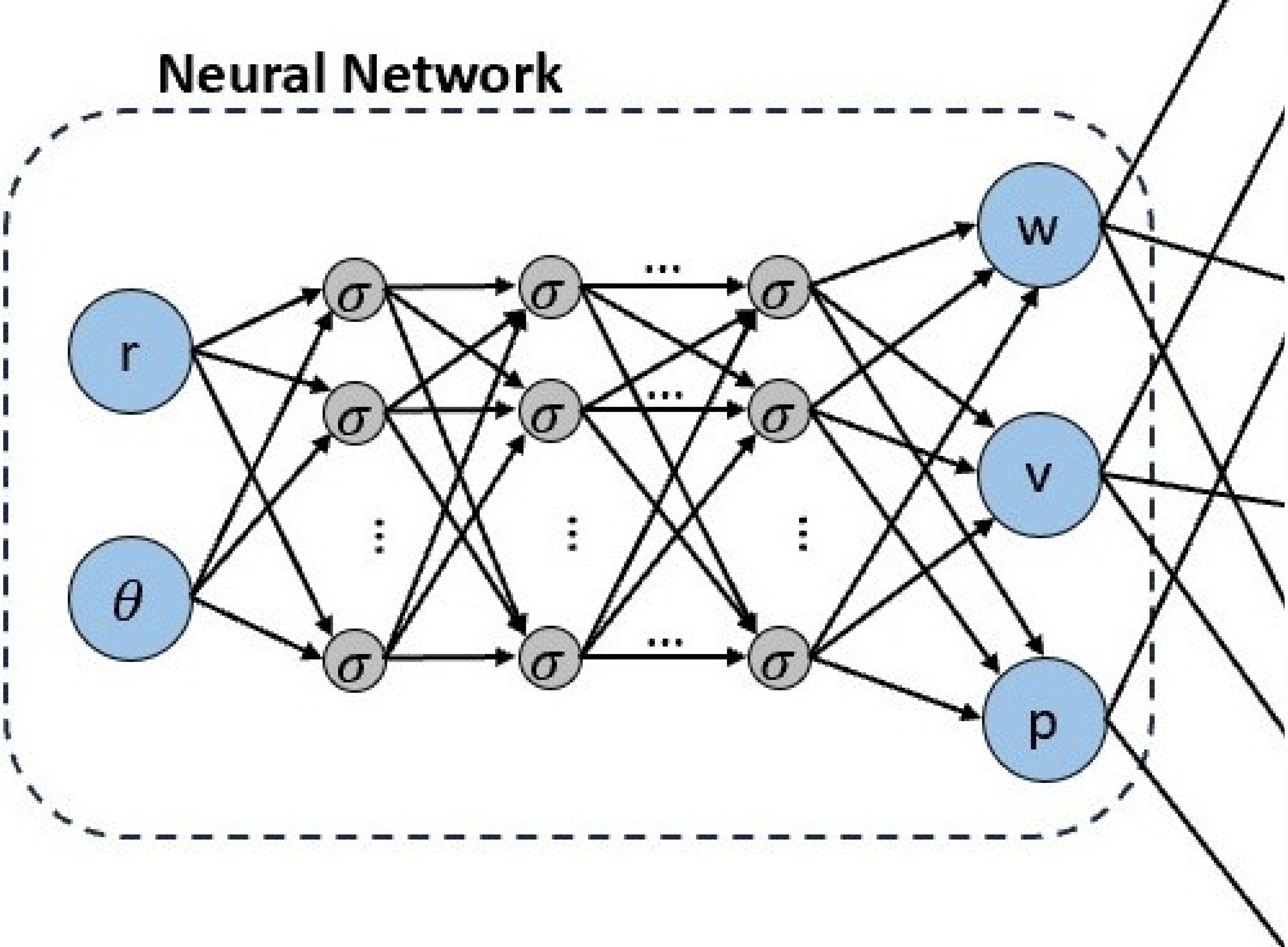
METHOD

DeepXDE

- A package in python that uses deep learning to solve differential equations
- Trained by exposing system to lots of data, and once trained, can make predictions/decisions based on new data it hasn't seen before
- Our training data consists of sample points in our domain where the differential equations are evaluated

PINN DIAGRAM




Neural Network
r
θ
σ
w
v
p

$w, \frac{\partial w}{\partial r}, \frac{\partial w}{\partial \theta}, \frac{\partial^2 w}{\partial r^2}, \frac{\partial^2 w}{\partial \theta^2}, \frac{\partial^2 w}{\partial r \partial \theta}$
$v, \frac{\partial v}{\partial r}, \frac{\partial v}{\partial \theta}, \frac{\partial^2 v}{\partial r^2}, \frac{\partial^2 v}{\partial \theta^2}, \frac{\partial^2 v}{\partial r \partial \theta}$
$p, \frac{\partial p}{\partial r}, \frac{\partial p}{\partial \theta}$
$\frac{\partial w}{\partial r}, \frac{\partial v}{\partial r}, \frac{\partial w}{\partial \theta}$

$$r^2 \sin(\theta)\frac{\partial p}{\partial r} + \cos(\theta)\left(2v - \frac{\partial w}{\partial \theta}\right) + \sin(\theta)\left(-2r\frac{\partial w}{\partial r} + 2w - \frac{\partial^2 w}{\partial \theta^2} - r^2\frac{\partial^2 w}{\partial r^2} + 2\frac{\partial v}{\partial \theta}\right) = 0$$

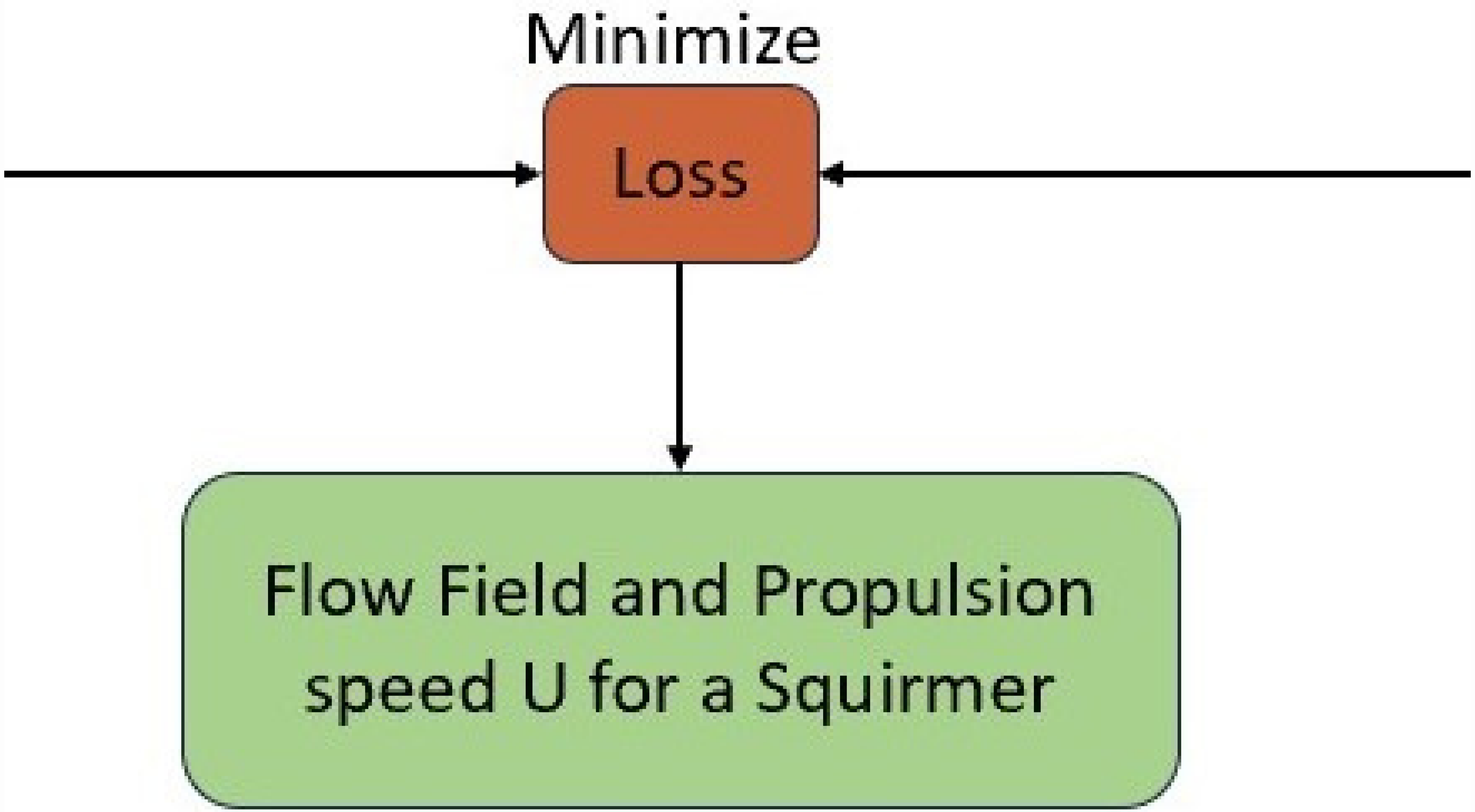
$$sin^2(\theta)\frac{\partial p}{\partial \theta} - \left(2r\sin(\theta)\frac{\partial v}{\partial r} + r^2 sin^2(\theta)\frac{\partial^2 v}{\partial r^2} + sin^2(\theta)\frac{\partial^2 v}{\partial \theta^2} + \sin(\theta)\cos(\theta)\frac{\partial v}{\partial \theta} + 2\sin(\theta)\frac{\partial w}{\partial \theta} - v\right) = 0$$

$$2\sin(\theta)\, w + r\sin(\theta)\frac{\partial w}{\partial r} + \sin(\theta)\frac{\partial v}{\partial \theta} + v\cos(\theta) = 0$$

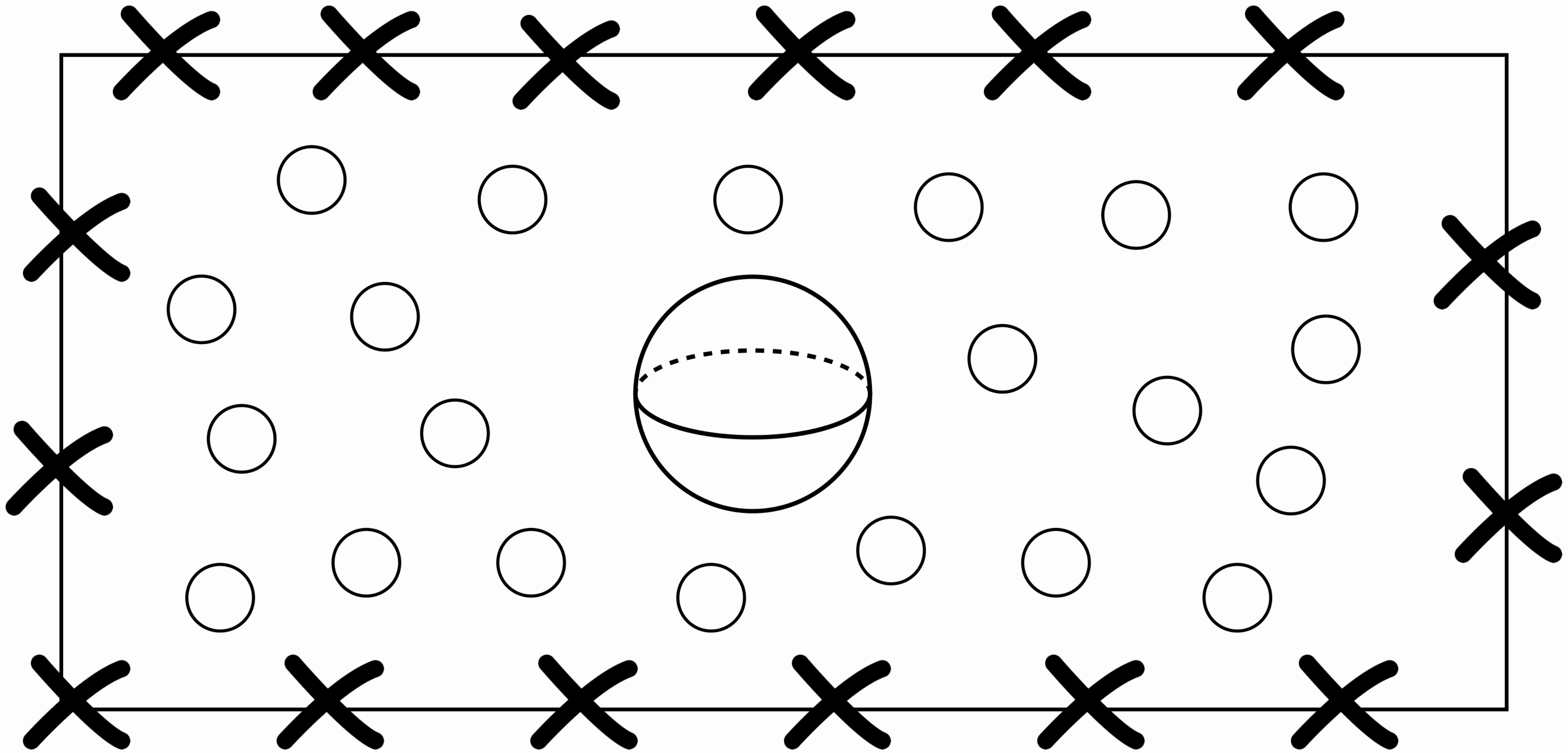
$$\mathrm{w}(a,\theta) = U\cos(\theta)$$
$$\mathrm{w}(A,\theta) = 0$$

$$v(a,\theta) = (1-U)\sin(\theta) + \alpha\sin(\theta)\cos(\theta)$$
$$v(A,\theta) = 0$$

$$\int sin(\theta)\left(\left(-pr + 2r\frac{\partial w}{\partial r}\right)cos(\theta) - \left(r\frac{\partial v}{\partial r} - v + \frac{\partial w}{\partial \theta}\right)sin(\theta)\right)d\theta = 0$$


Minimize
Loss
Flow Field and Propulsion speed U for a Squirmer

HYPERPARAMETERS



Points in Domain: 1000

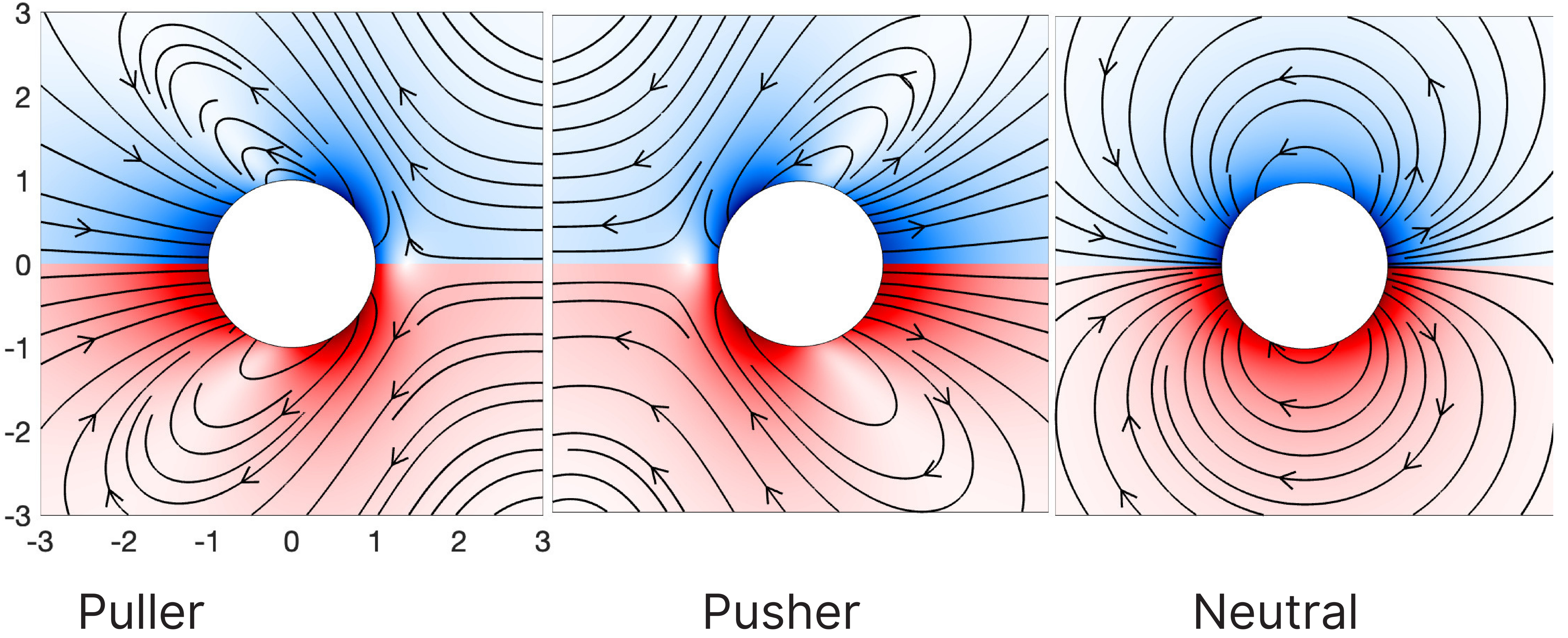
Points on Boundary: 300

Activation Function: Tanh

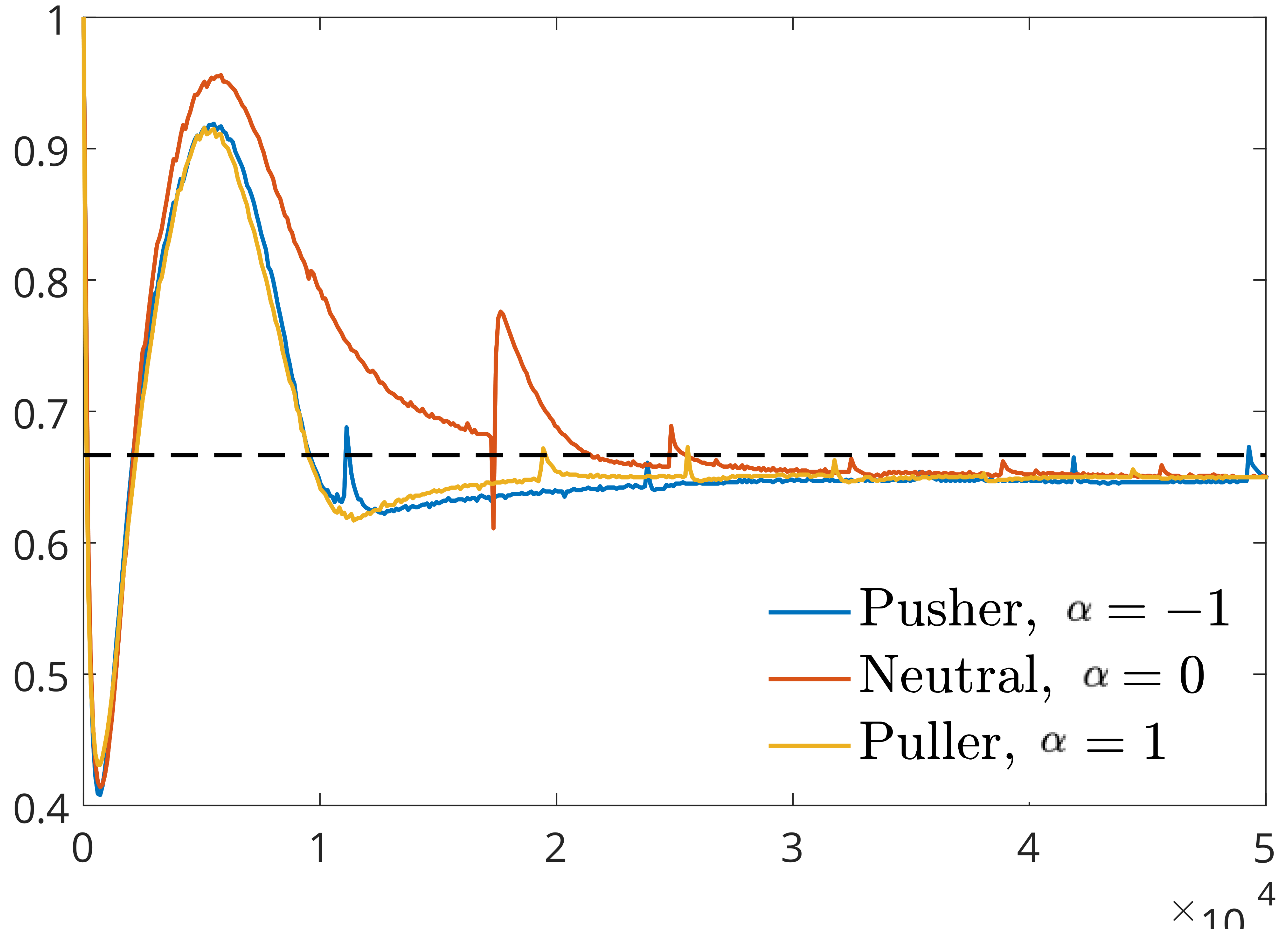
Learning Rate: 3x10^-3

3 Hidden Layers x 100 Nodes

RESULTS



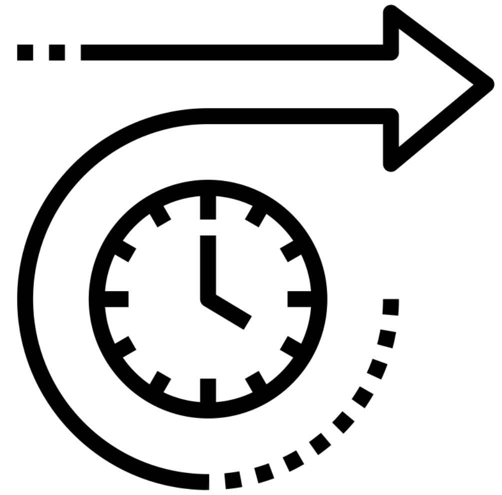
RESULTS



1
0.9
0.8
0.7
0.6
0.5
0.4
0
1
2
3
4
5
$\times 10^4$
Pusher, $\alpha = -1$
Neutral, $\alpha = 0$
Puller, $\alpha = 1$

SUMMARY/FUTURE WORKS

- The future work is to get the % of error as low as it can possibly go, which is being done by editing the hyper parameters
- The ongoing research is extending our PINN implementation to different fluids, including shear thinning and Brinkman
- The hope is to eventually apply these methods to model drug delivery in the human body



THANK YOU!

Grant No. DMS-2149865

Mentors:

Hervé Nganguia

Jia-An Yan