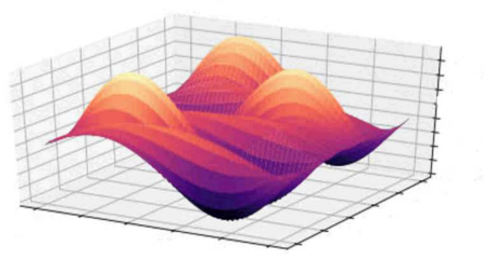
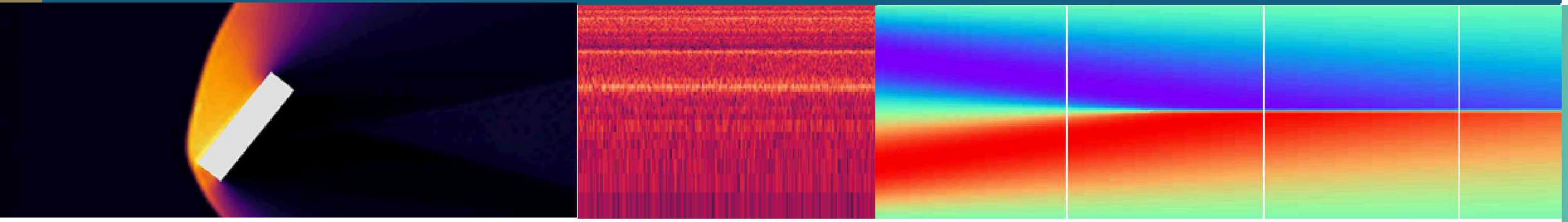
SAND2019-9664C

Modeling of Multi-Physics and Multi-Component Dynamics with Neural Networks





PRESENTED BY

Carianne Martinez

David Najera-Flores

ATA Engineering

Kevin M. Potter, Peter D. Yeh, Carianne Martinez, Adam Brink

Sandia National Laboratories

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

SAND2019-xxxx

Outline

Motivation and Hypothesis

Introduction to Deep Learning

Deep Learning for Fragment Characterization

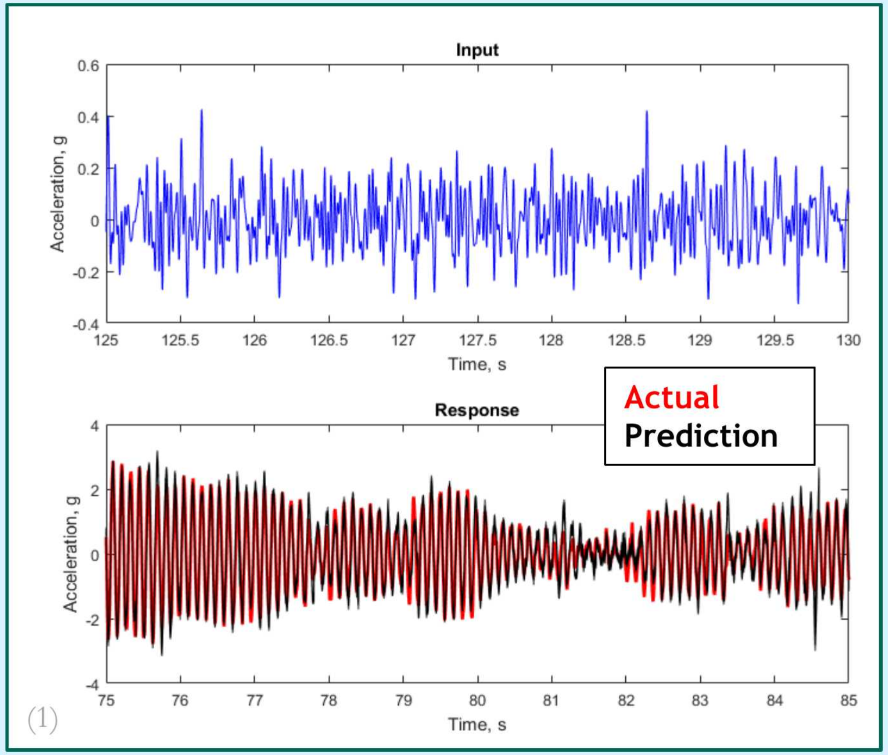
Network to Network Communication

- Iterative Neural Network Solutions
- One way coupled FSI

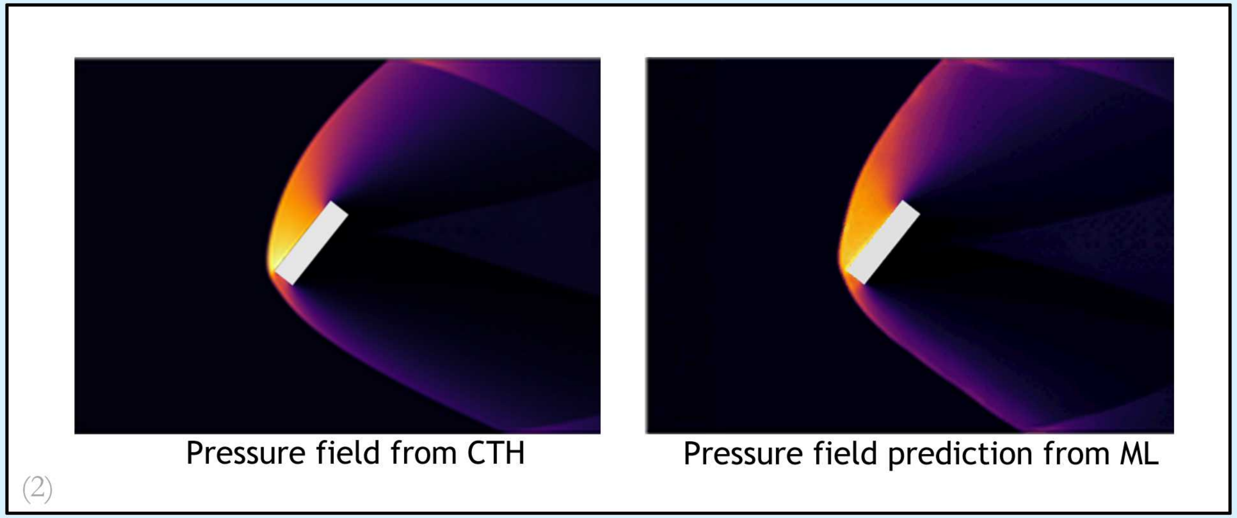
Future Work

Neural Networks can approximate complex functions

Our hypothesis: Deep learning can be used to emulate physics as a data driven model.



Signal of nonlinear system extended with ML (1)

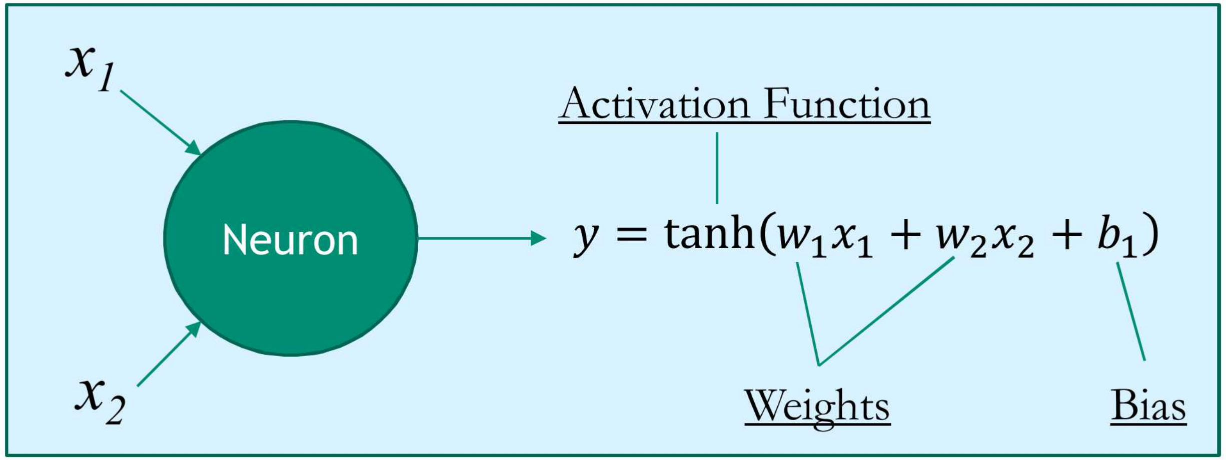


Pressure contour predicted by ML compared to simulated data (2)

(1) D. A. Najera-Flores and A. R. Brink, “Efficient random vibration analysis of nonlinear systems with long short-term memory networks for uncertainty quantification,” *Proceedings of ISMA*, Oct. 2018.

(2) P. D. Yeh and et al., “Physics-informed deep learning model for predicting ballistic coefficients of explosively driven fragments,” *1st Annual Meeting of the APS Division of Fluid Dynamics* , Nov. 2018.

The disruptive paradigm that is machine learning



Simple Single Neuron Example

At the neuron:

- Dot product of the weights with the inputs.
- Addition of bias term.
- Activation of function introduces nonlinearity.

Training Neural Networks:

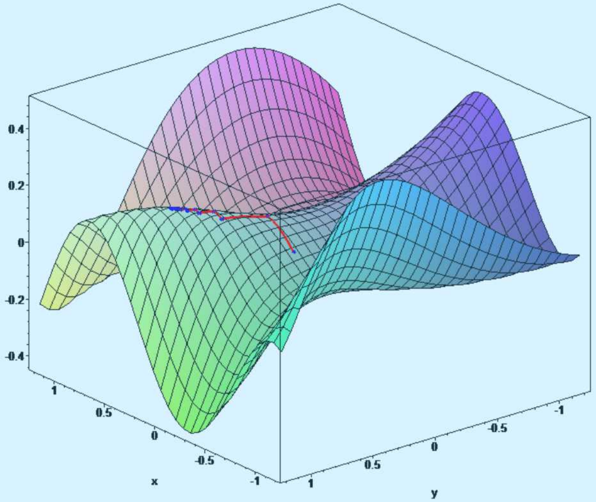
- Establish a ground truth dataset.
- Optimize weights to minimize the loss function.
- Gradient descent is a common technique.

Loss Function

$$L(y, \hat{y}) = \frac{1}{m} \sum_{i=1}^{m} (y_i - \hat{y}_i)^2$$

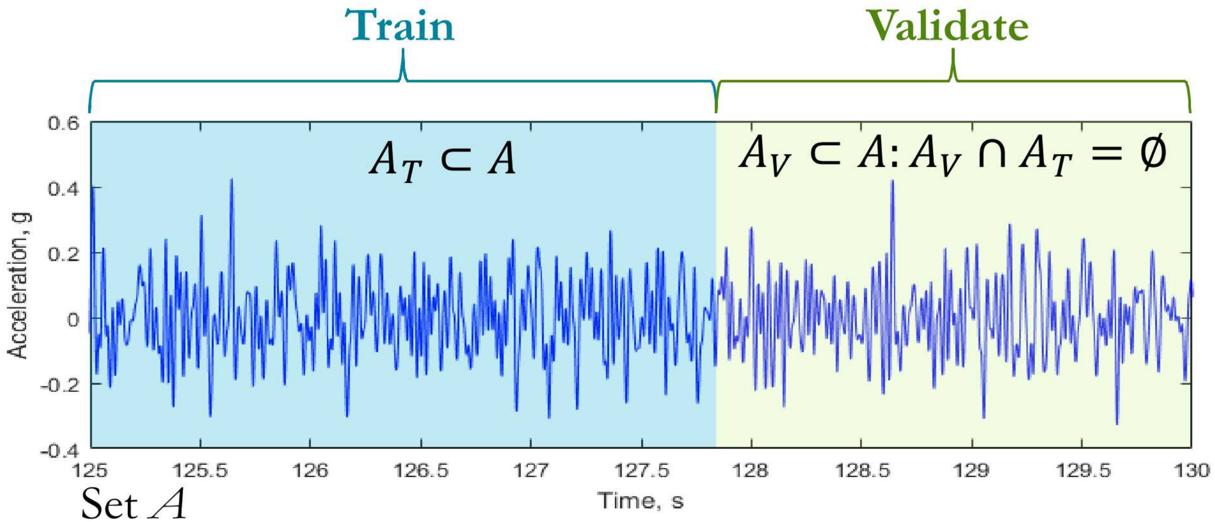
Gradient Descent

$$w_j^{i+1} = w_j^i - lr \frac{\partial L}{\partial w_j}$$



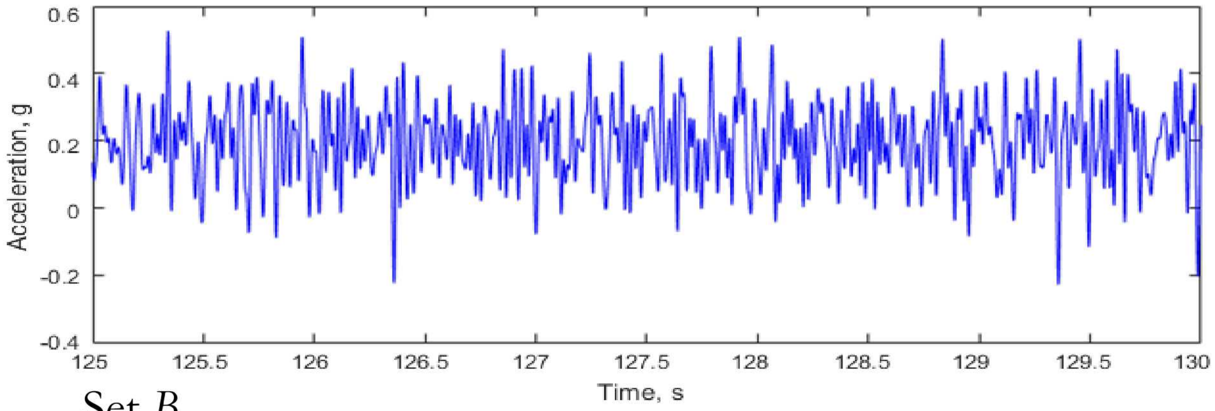
Optimization of Weight Functions

Neural networks are developed and used in three distinct stages



Set A

Infer



Set B

Train: Using a subset A_T of data from set A to optimize an error function by adjusting neuron weights.

Validate: Using a different, non-overlapping subset of data A_V from set A to show the trained network can replicate the output.

Inference: Using the trained and validated network to predict output from a completely new and unknown dataset B.

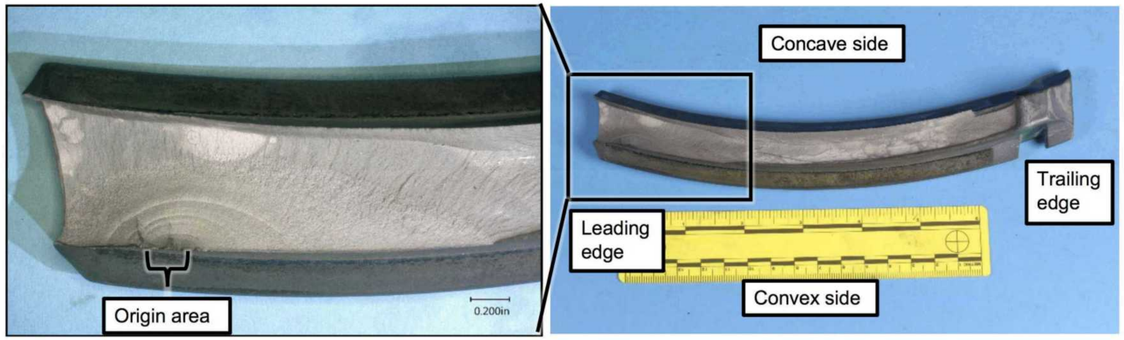
Training is the most costly part. Inference is cheap once the network is trained.

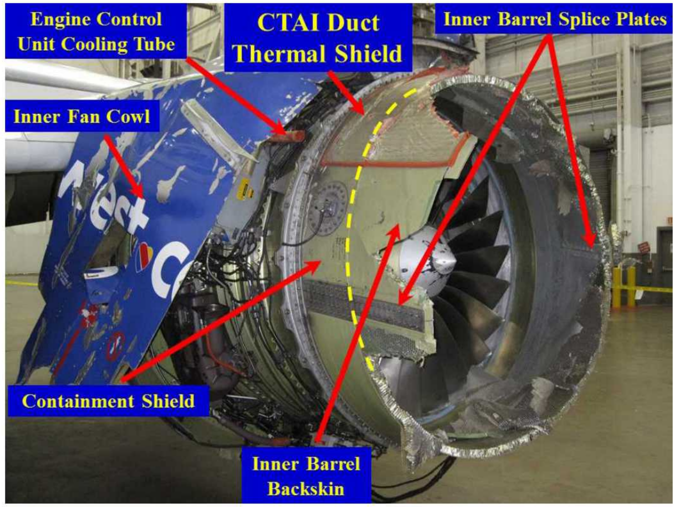
Deep Learning for Fragment Characterization

Understanding explosive fragment flight

Case Study: Southwest Airlines Engine Explosion

- Southwest flight 1380, April 17th, 2018
  - Engine failure after takeoff from New York LaGuardia
  - Metal fragments from explosion punctured fuselage
  - 1 fatality, several injuries
- How can we understand fragment flight to prevent future safety incidents?





https://www.nytimes.com/2018/04/17/us/southwest-airlines-explosion.html
https://en.wikipedia.org/wiki/Southwest_Airlines_Flight_1380

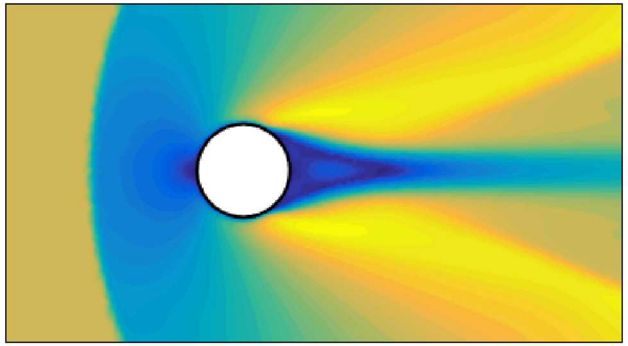
Understanding range of fragment flight

Explosive fragments fly at supersonic speeds

Current methods assume single drag coefficient

Geometry is complicated, high aspect ratio

Fragment-air interaction leads to tumbling and chaotic motion

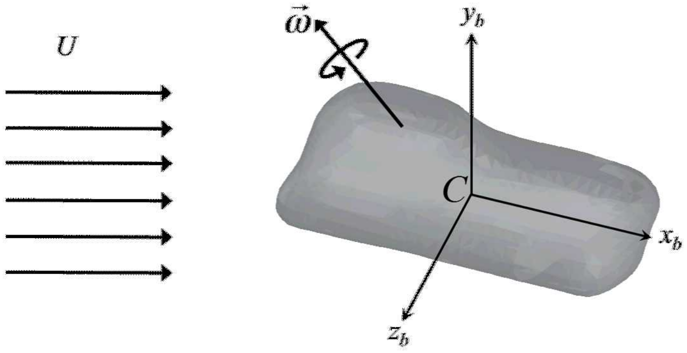


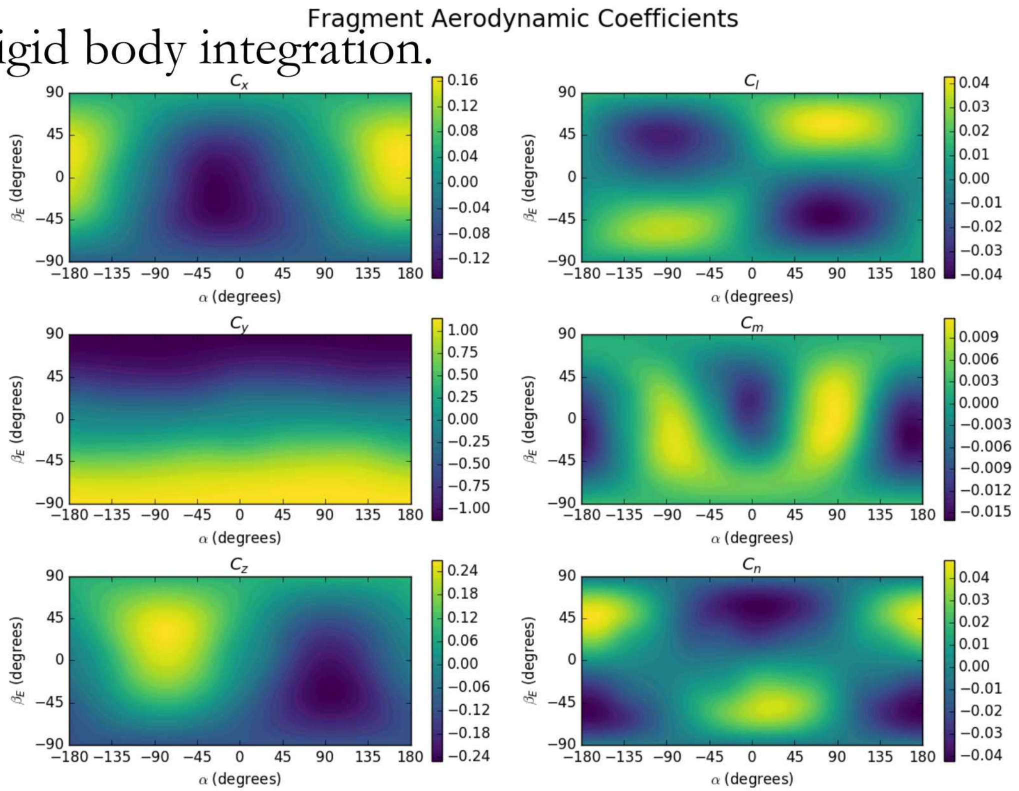
Goal is to characterize range of a set of flying fragments

Previously developed fragment flight simulation procedure

1) Compute aerodynamic coefficients at all orientations with high fidelity solver.

2) Compute trajectories with rigid body integration.





An explosive may generate over 10,000 fragments. Simulating all of them is prohibitively expensive!

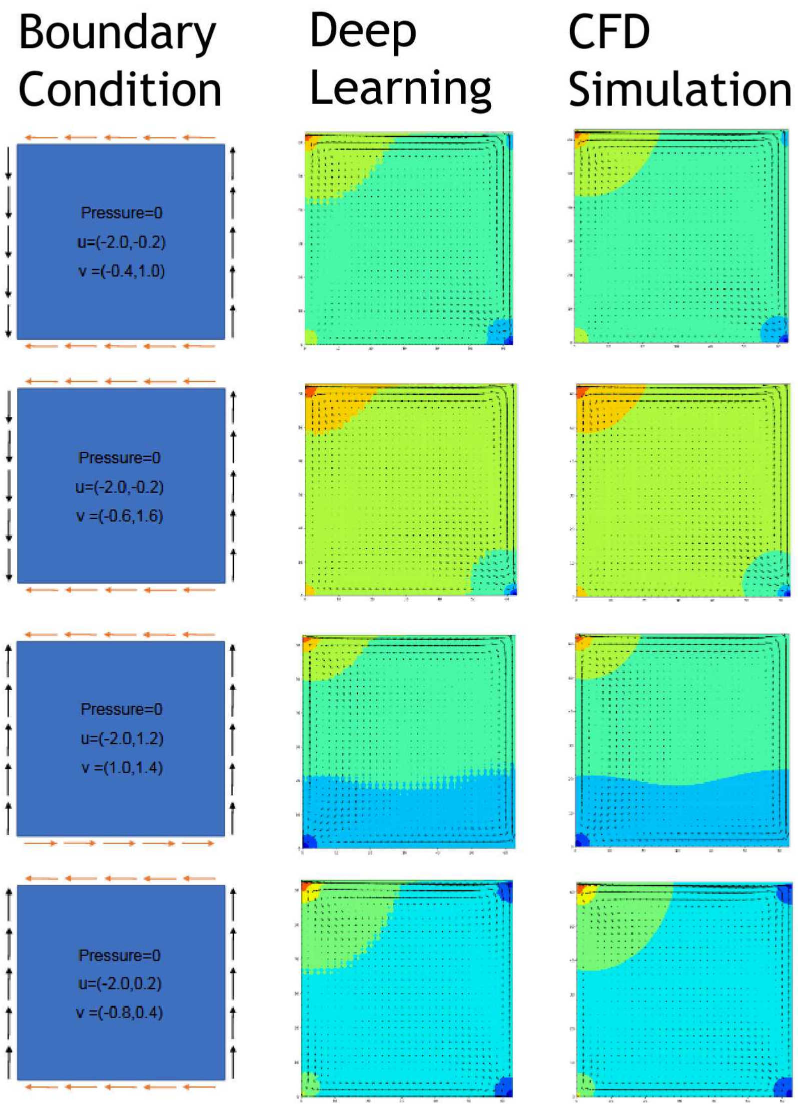
How can we speed up aerodynamic calculation? Deep Learning!

Lid driven cavity solution approximated using Deep Learning (Stanford, 2017)

Used Generative Adversarial Networks (GANs), adapted pix2pix algorithm

Achieved **orders of magnitude** speed-up in inference time

Approximating aerodynamics using deep learning shows potential

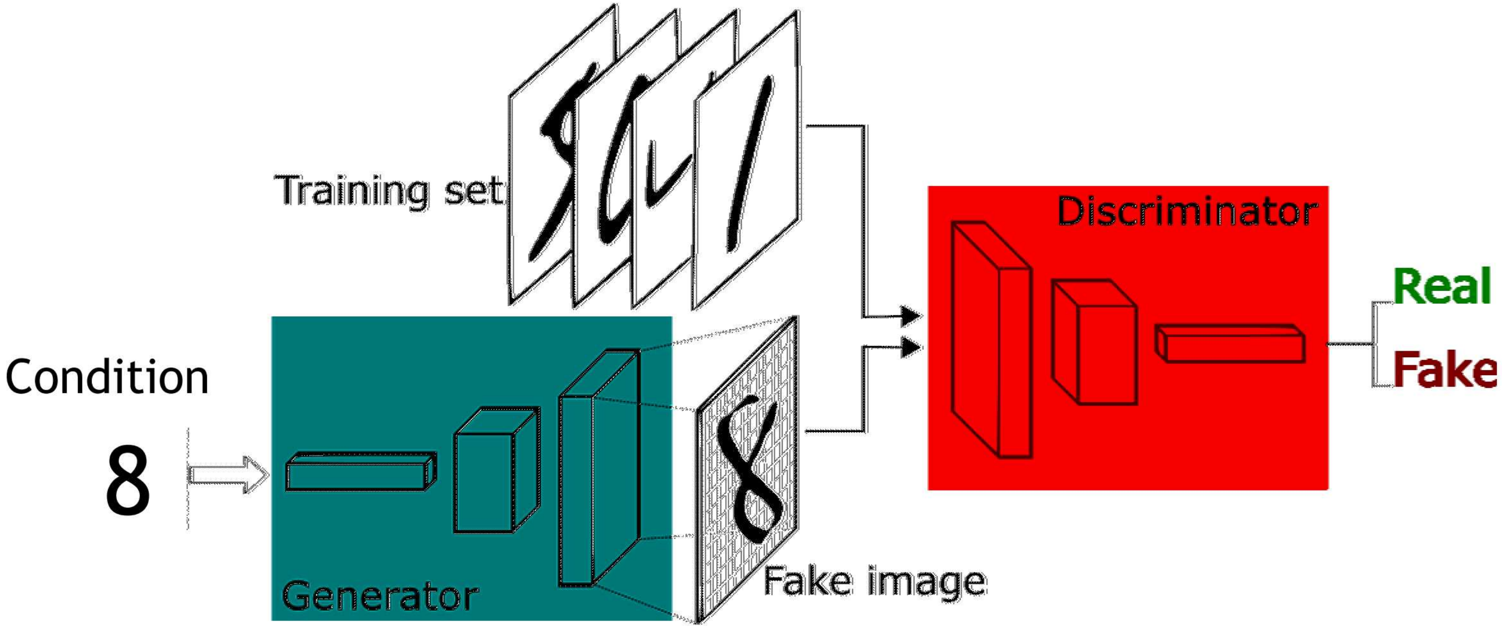


Farimani et al. 2017

Generative Adversarial Networks: a game theoretic approach to machine learning

Generative Adversarial Networks (GANs) pit two competing neural networks against each other

- **The generator**, tries to mimic real results
- **The discriminator**, tries to identify mimicked results from real results



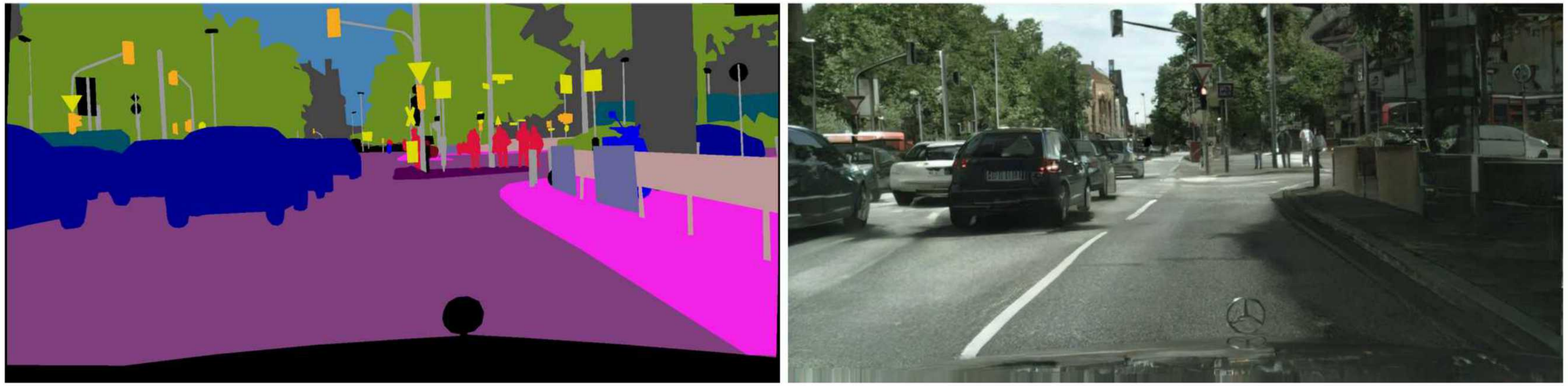
https://deeplearning4j.org/generative-adversarial-network

Generative Adversarial Networks (GANs) learn to mimic complex systems with wide applicability

Pix2pix model

- Condition is an image instead of a label
  - E.g. color segmentation of a scene
- GAN has to learn how to fill in segmentations convincingly
- Training goal is to fool the discriminator

Input Output

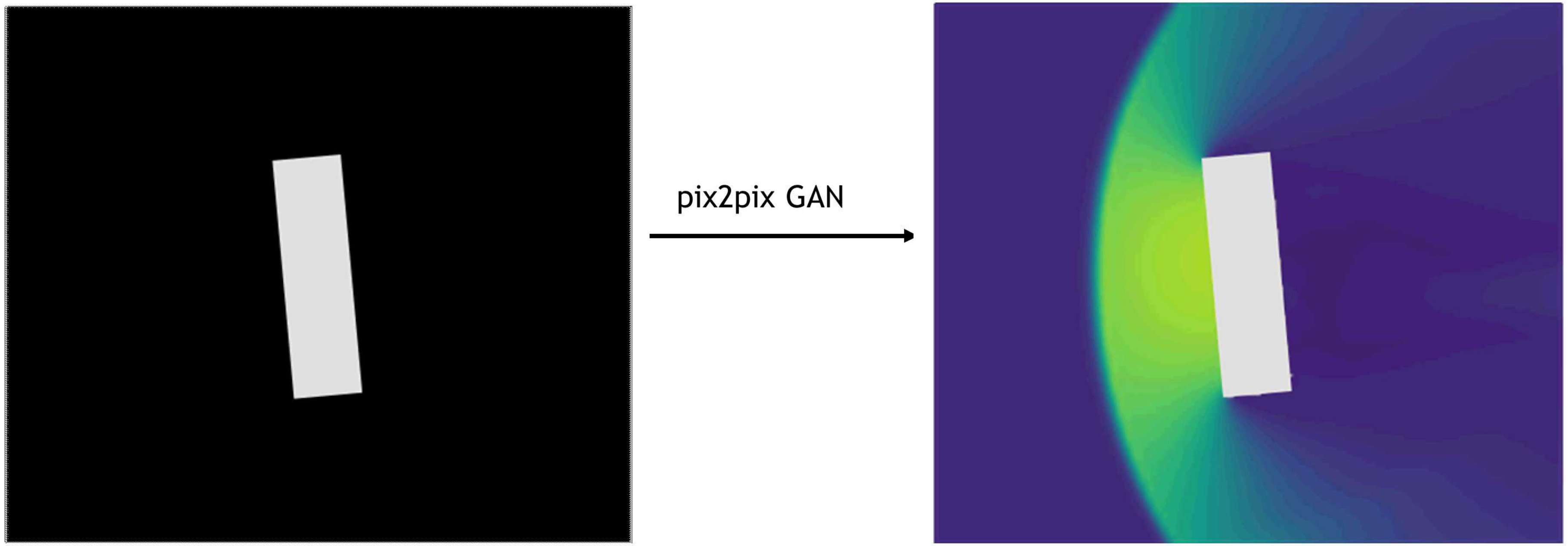


Wang, et al. “High-Resolution Image Synthesis and Semantic Manipulation with Conditional GANS”

Our first attempt at using a GAN for flow prediction

Train pix2pix GAN using computed flow solutions as ground truth

- Simulated 1000 rectangles with random orientations & aspect ratios, Mach 5 external flow
- 900 training examples, 100 held out test examples
- Pressure fields calculated with compressible Euler equation solver (CE Solver)
- Ideal gas assumption for simplicity



Input Data

Intended Outcome

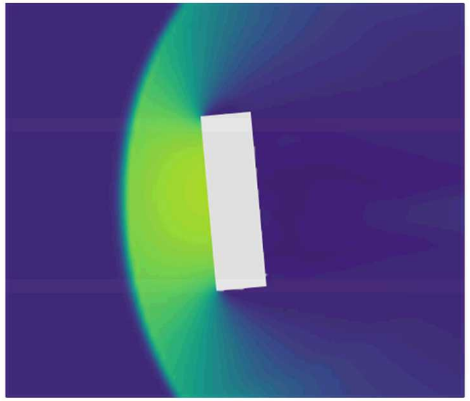
Predicted pressure is close on leading edge, but drag is inaccurate

GAN successfully approximates the pressure map along the leading edge (left) of the object

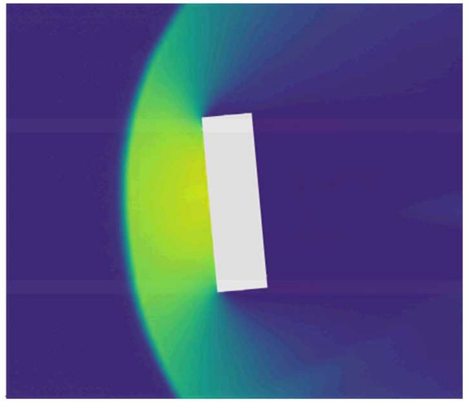
Larger error behind the object due to unsteady wake and fluctuating lower pressure

Inaccurate drag calculation despite close values in pressure

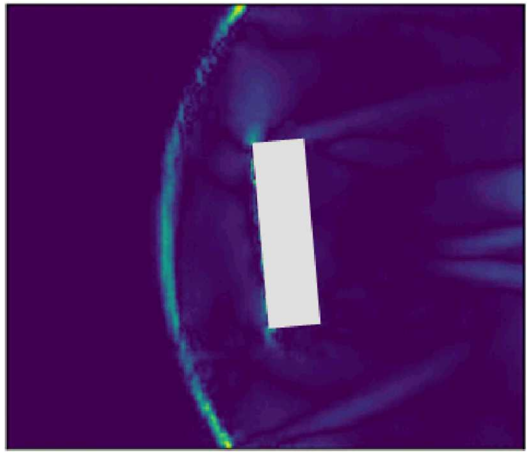
CE Solver



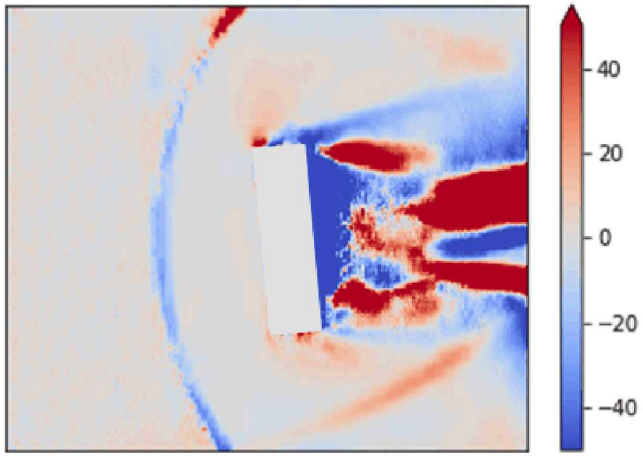
ML



Pressure Abs. Difference



Pressure Percent Difference



Solving conservation equations is expensive and complex… …but solutions are easy and cheap to check

Punishes model for violating physics constraints

More accurately represents shockwave

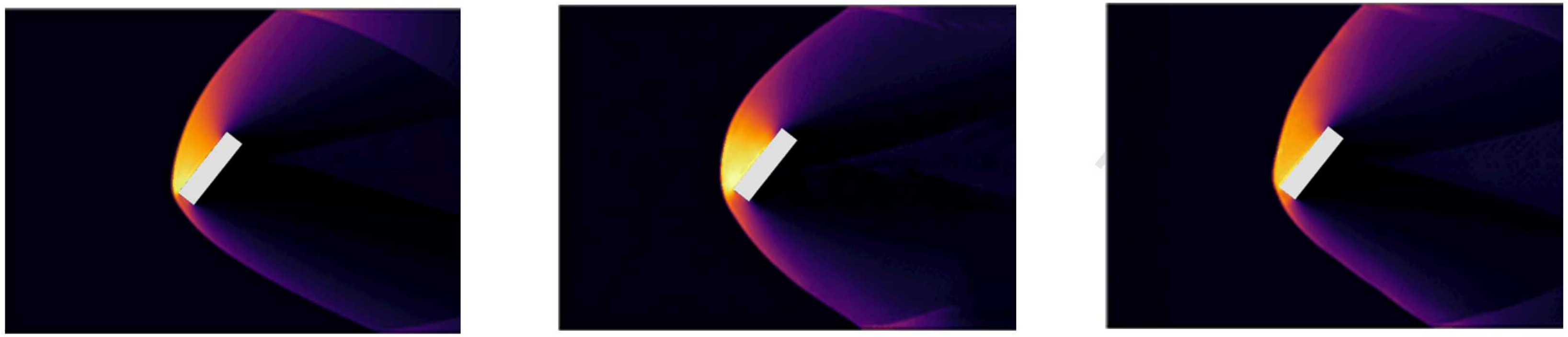
Improved accuracy around the edges of the object

$$\begin{cases} \dfrac{\partial \rho}{\partial t} + \boldsymbol{u} \cdot \nabla \rho + \rho \nabla \cdot \boldsymbol{u} = \boldsymbol{0} \\ \dfrac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} + \dfrac{\nabla p}{\rho} = \boldsymbol{g} \\ \dfrac{\partial e}{\partial t} + \boldsymbol{u} \cdot \nabla e + \dfrac{p}{\rho} \nabla \cdot \boldsymbol{u} = 0 \end{cases}$$

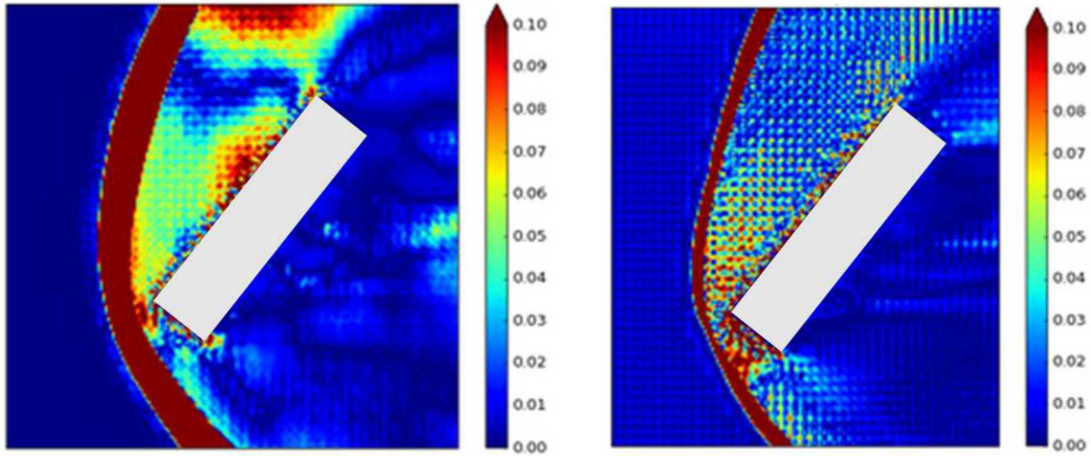
Exact solutions to differential equations are not required – ML can learn to approximate solutions

Adding physics loss term to loss function is a clear improvement

CE Solver | GAN Loss | GAN Loss + Physics Loss



Error



Enforce momentum and mass conservation in generator

Punishes model for violating physics constraints

Improved accuracy in pressure field prediction

Successful Deep Learning prediction of fragment aerodynamic forces in 2D

Total Loss = **GAN Loss** + weights * **Physics Loss** + weights * **Force Loss**

Appropriate tuning of weights leads to successful predictive model

	Mean Relative Error vs CE Solver
Drag	1.87%
Lift	5.63%
Torque	2.29%

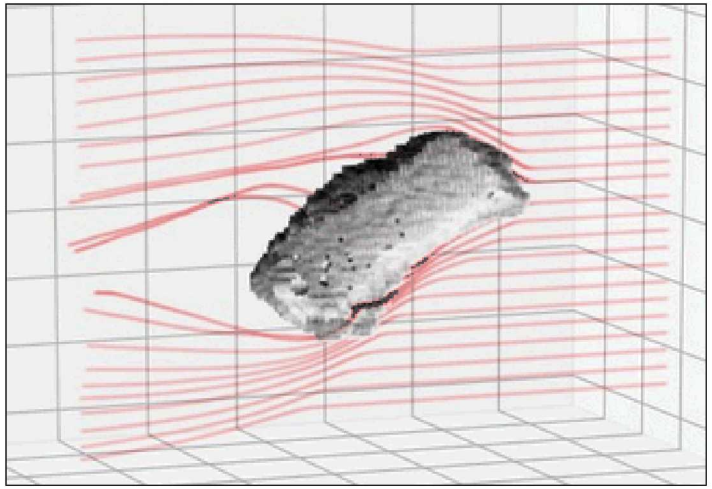
ML predictions approximate CE solver results within 6%

3D calculations are showing the same early promise as 2D

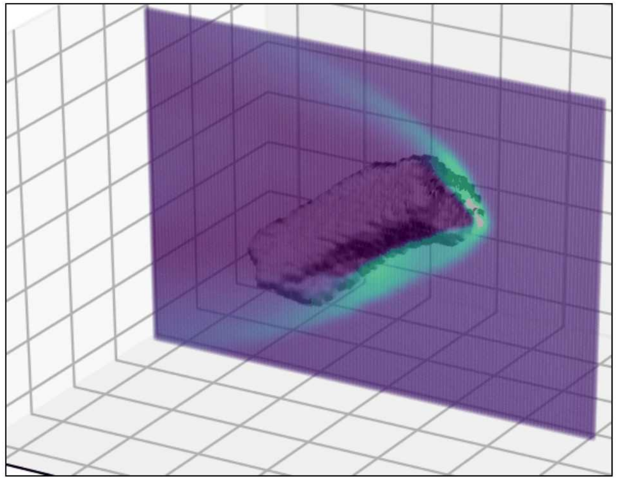
Qualitative results are visually similar to simulation output

Captures complex fragment geometries

~10,000x faster



Virtual wind tunnel



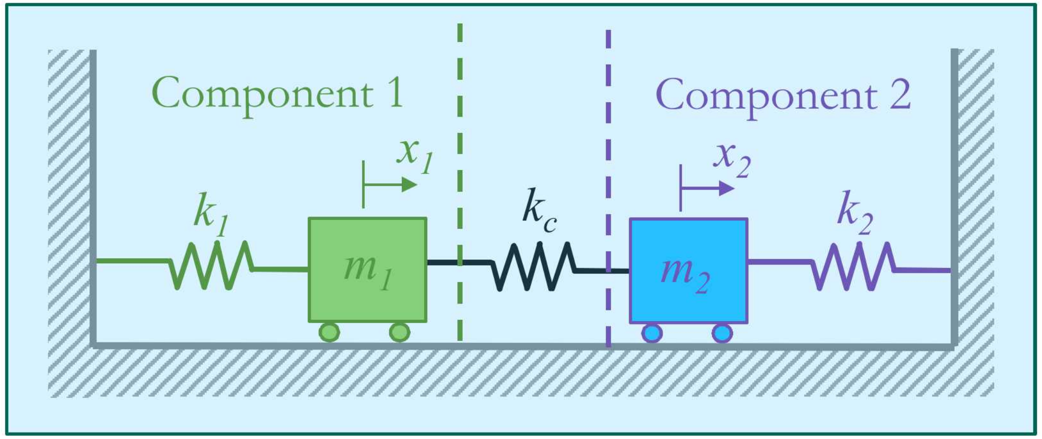
ML pressure prediction and fragment surface showing pressure field in a single plane

ML shows the same early promise with 3D as 2D showed before physics loss was added

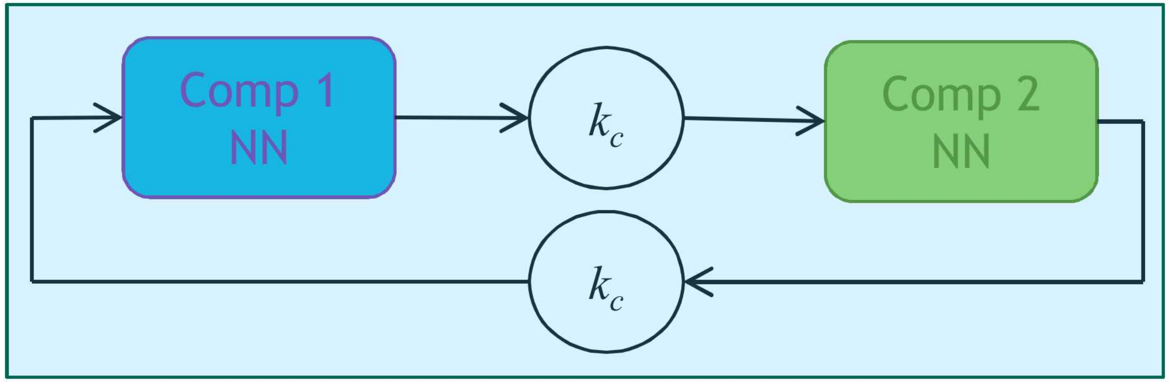
Data-Driven Coupled Neural Networks

Allowing neural networks to communicate will allow for the combination of disparate datasets as well as the emulation of coupled PDEs.

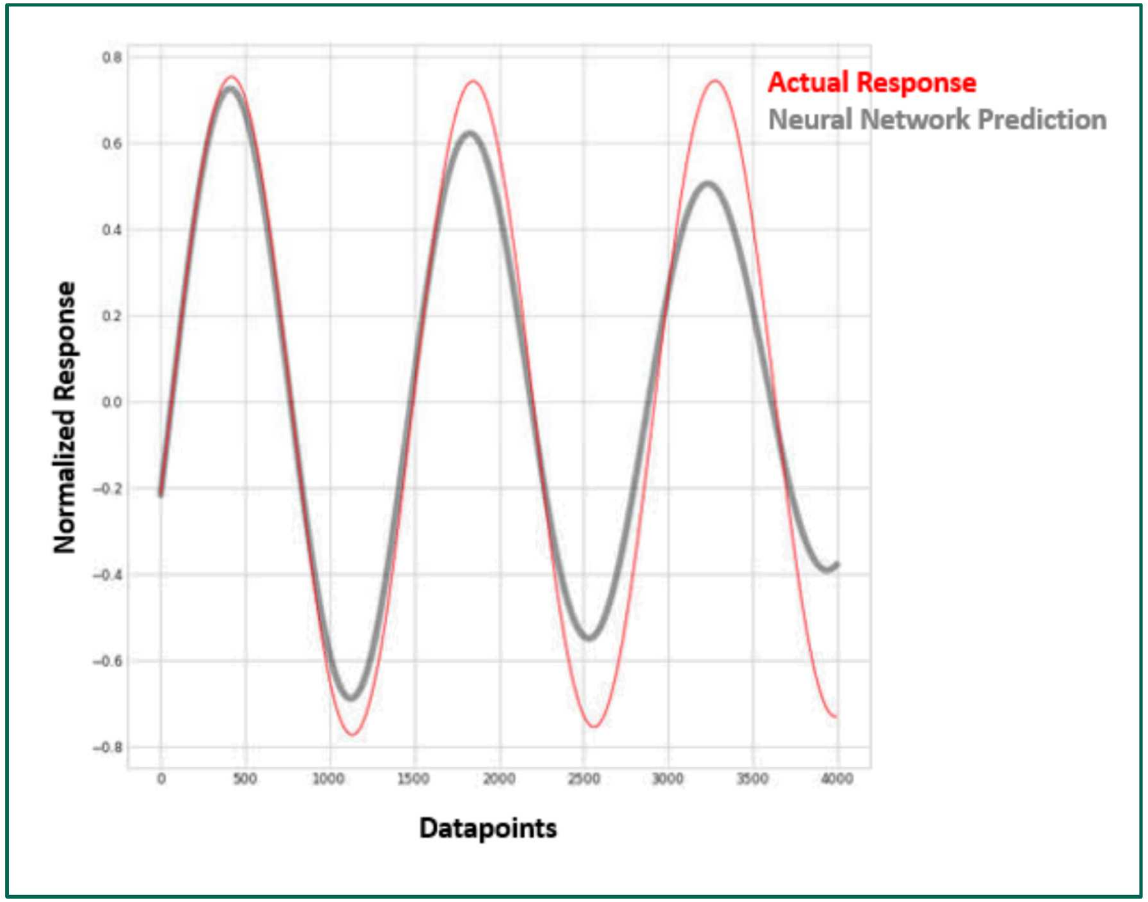
With a typical data-driven model, error accumulates too quickly when solving step-by-step



Multi-component system under consideration



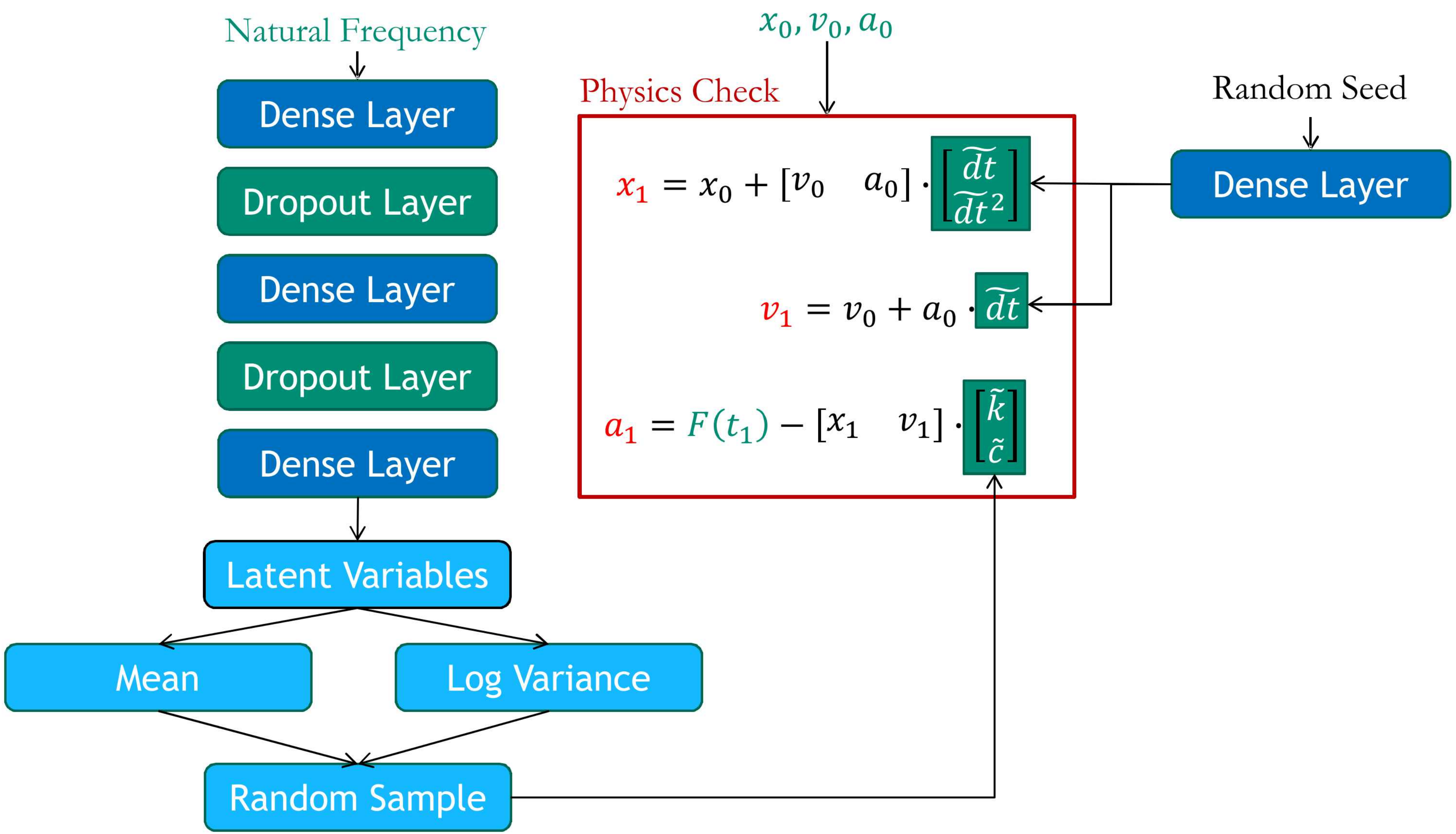
Step-by-step prediction with output from one NN feeds as input to another



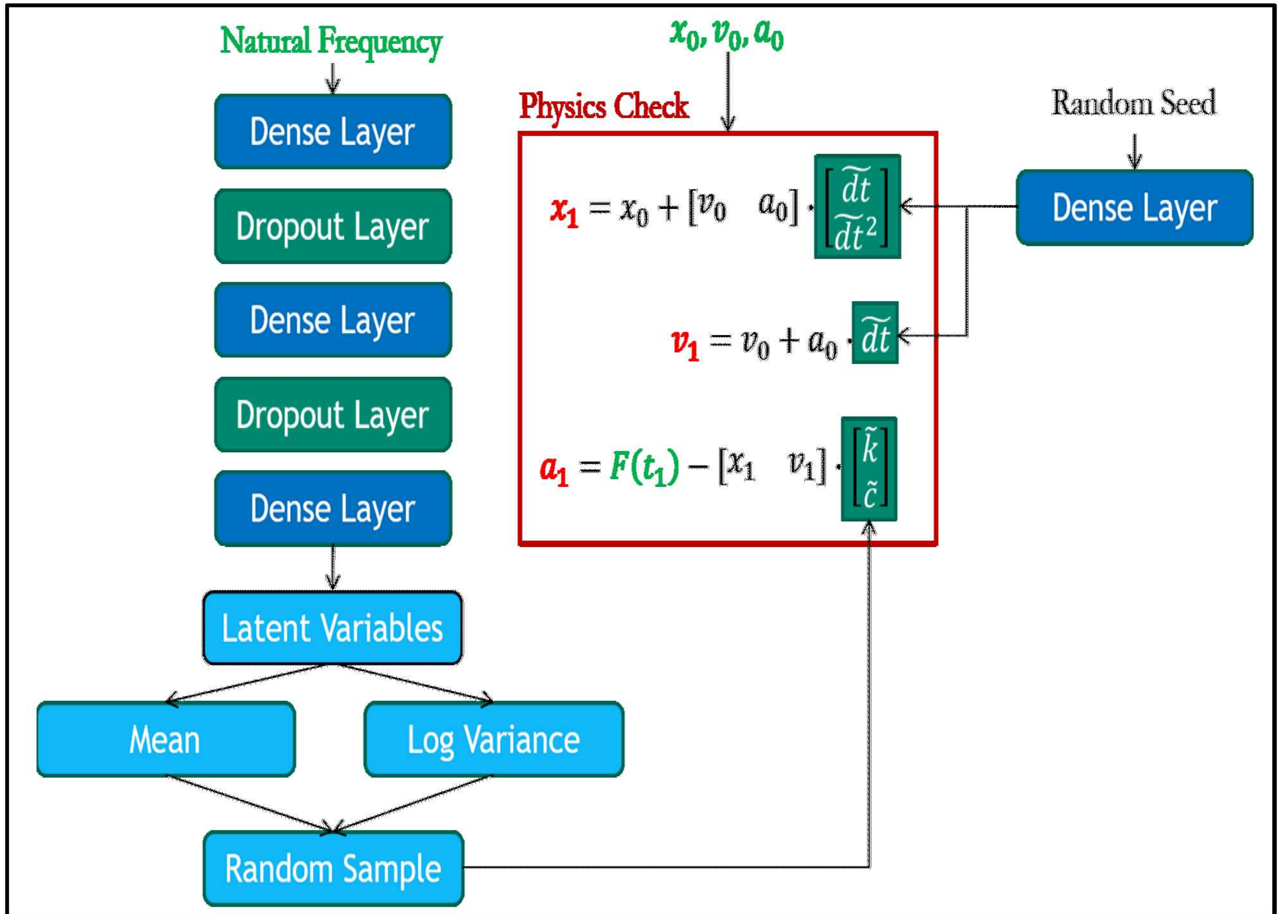
Error on amplitude growing with ML prediction

Lesson learned: communication between neural networks is challenging

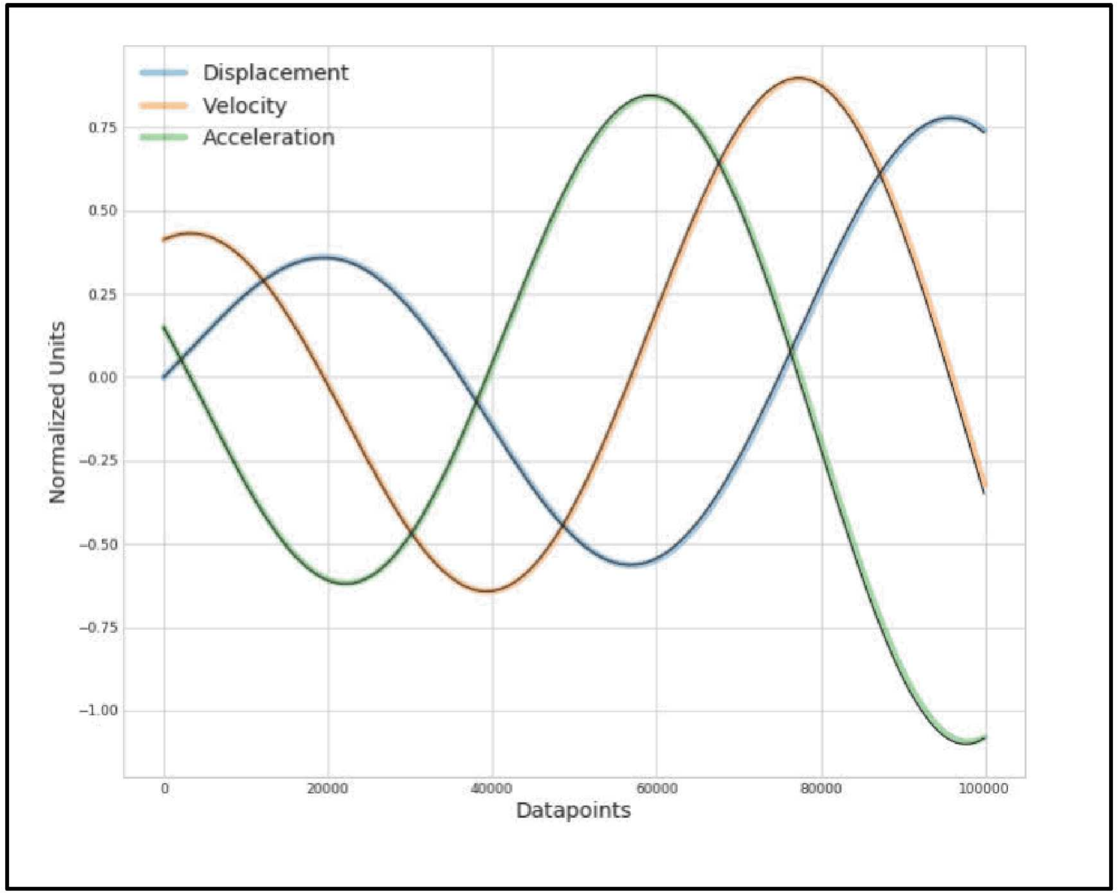
Error is significantly reduced by adding physics constraints to neural networks



Physics informed neural networks significantly reduce error



Training – Numerical (Python)
Inputs – Natural Frequency, Initial Conditions, Force
Outputs – Displacement, Velocity, Acceleration



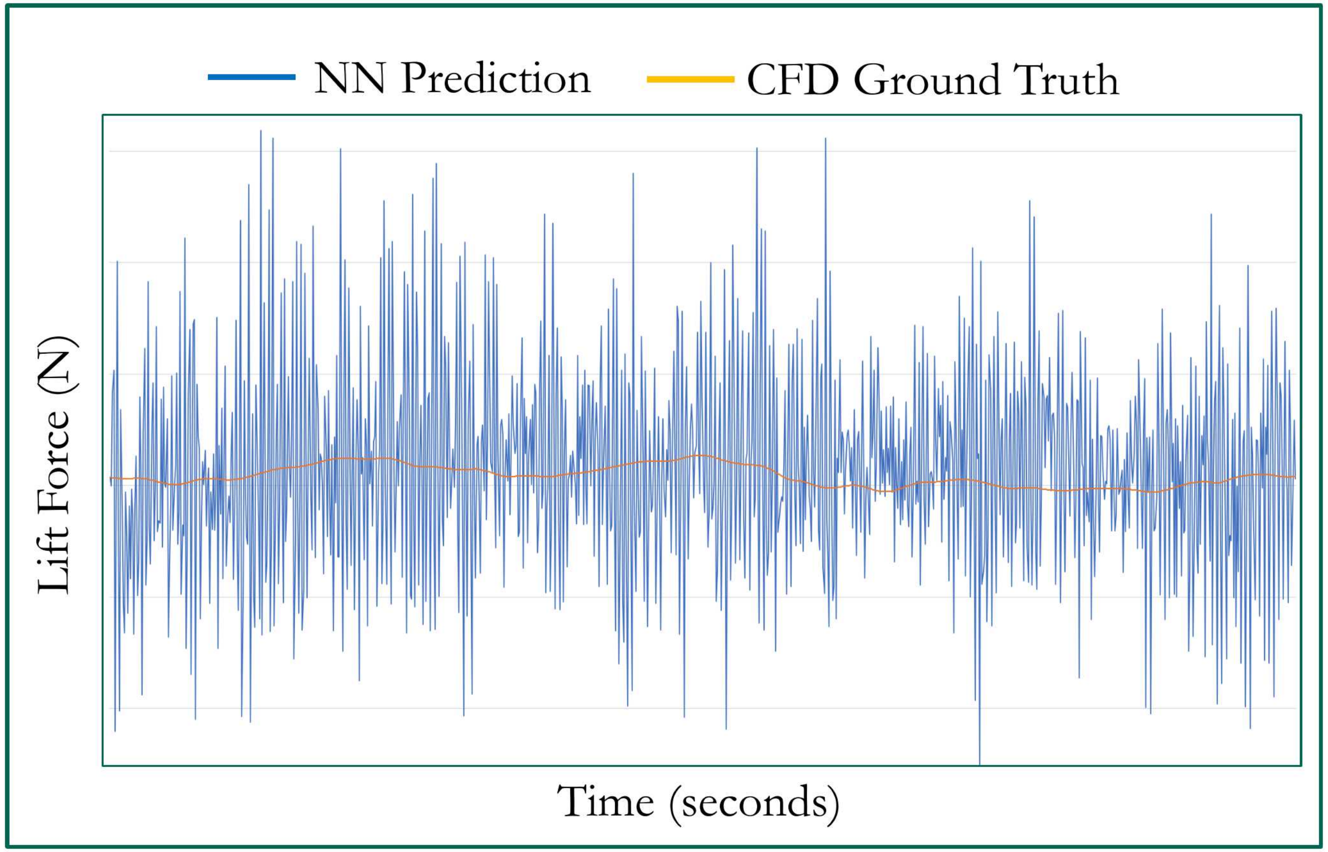
Prediction of single DOF oscillator
Black – Truth Data | Colored – NN Prediction

KEY TO MULTI-PHYSICS, MULTI-SCALE, MULTI-COMPONENT ANALYSIS.

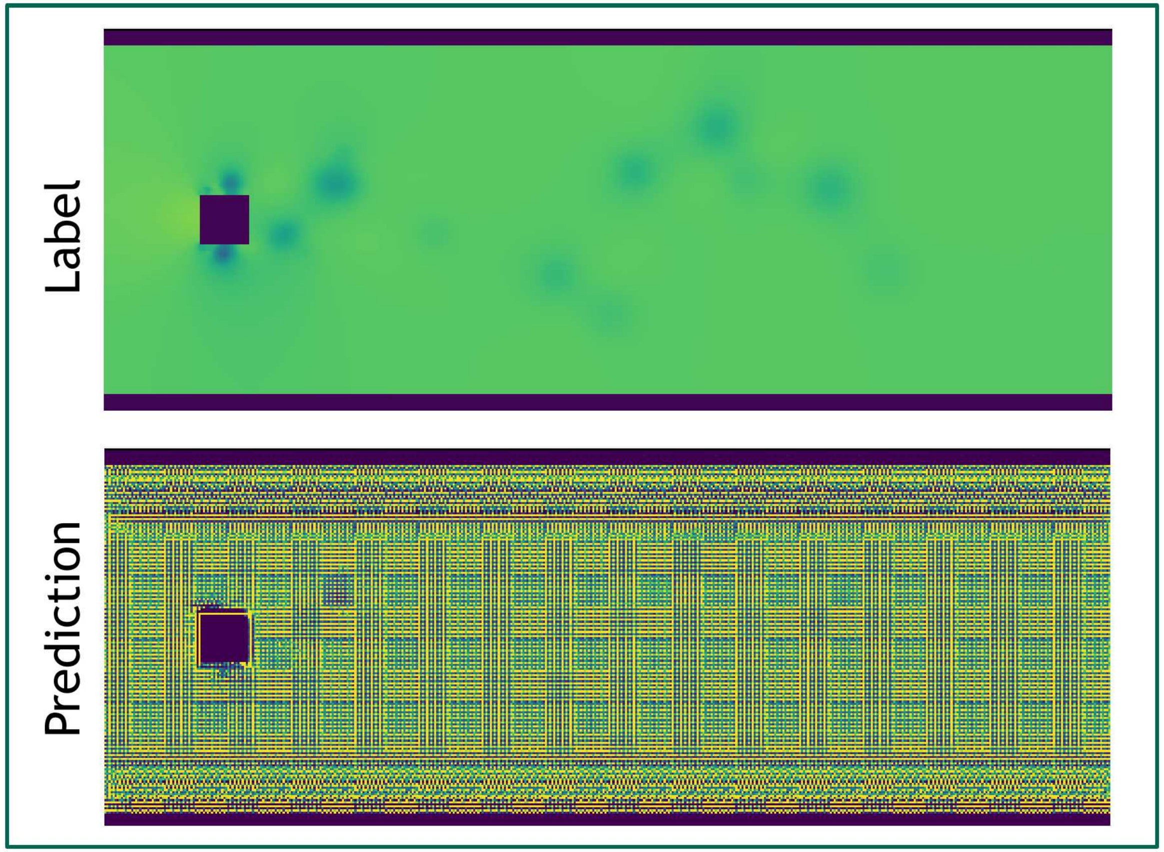
Deep Learning for Transient Fluid Flow

Novel deep learning training method

Naïve extension of deep learning model to transient systems resulted in poor predictive performance



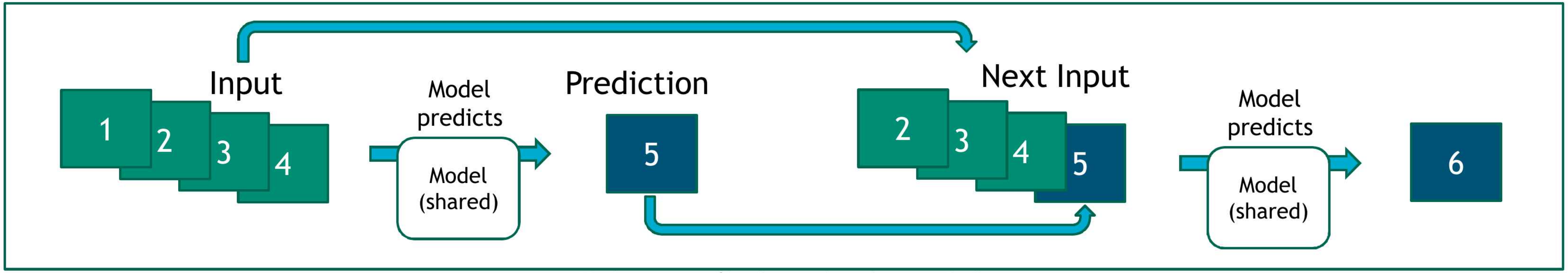
Lift force error with step-by-step fluid NN prediction



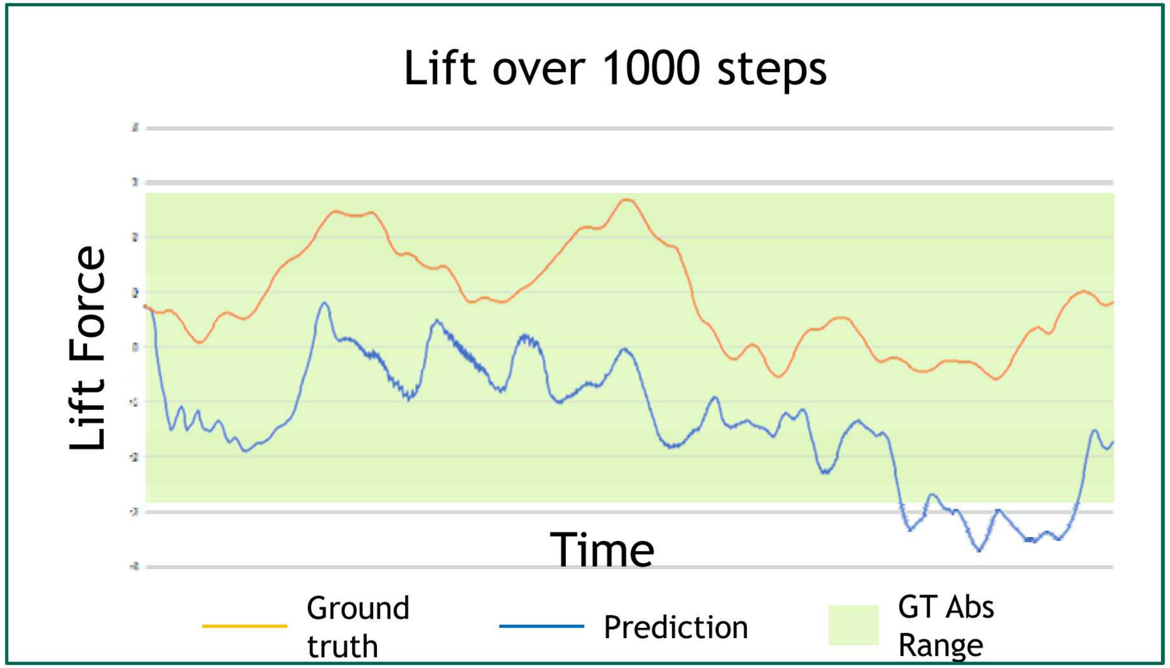
Pressure Field error with step-by-step fluid NN prediction

Transient full field solution suffered similar error accumulation.

Novel iterative delayed back propagation reduces error accumulation



Using predictions in the training set

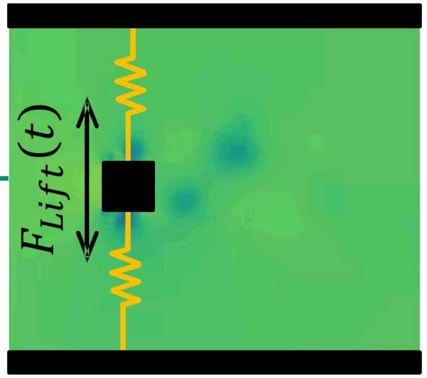


Novel algorithm improves lift force prediction

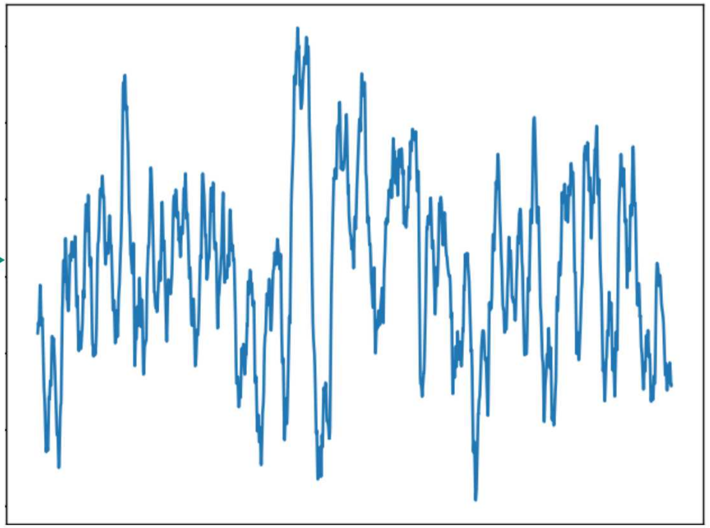
- Predictions are used in conjunction with ground truth during training
- Back propagation is delayed for *n* iterations
- Iteration number, *n*, is limited by error accumulation

Training – CFD (Fuego)
Inputs – Prior step pressure and velocity field
Outputs – Current step pressure and velocity field

Physics informed neural networks can solve one-way coupled problems

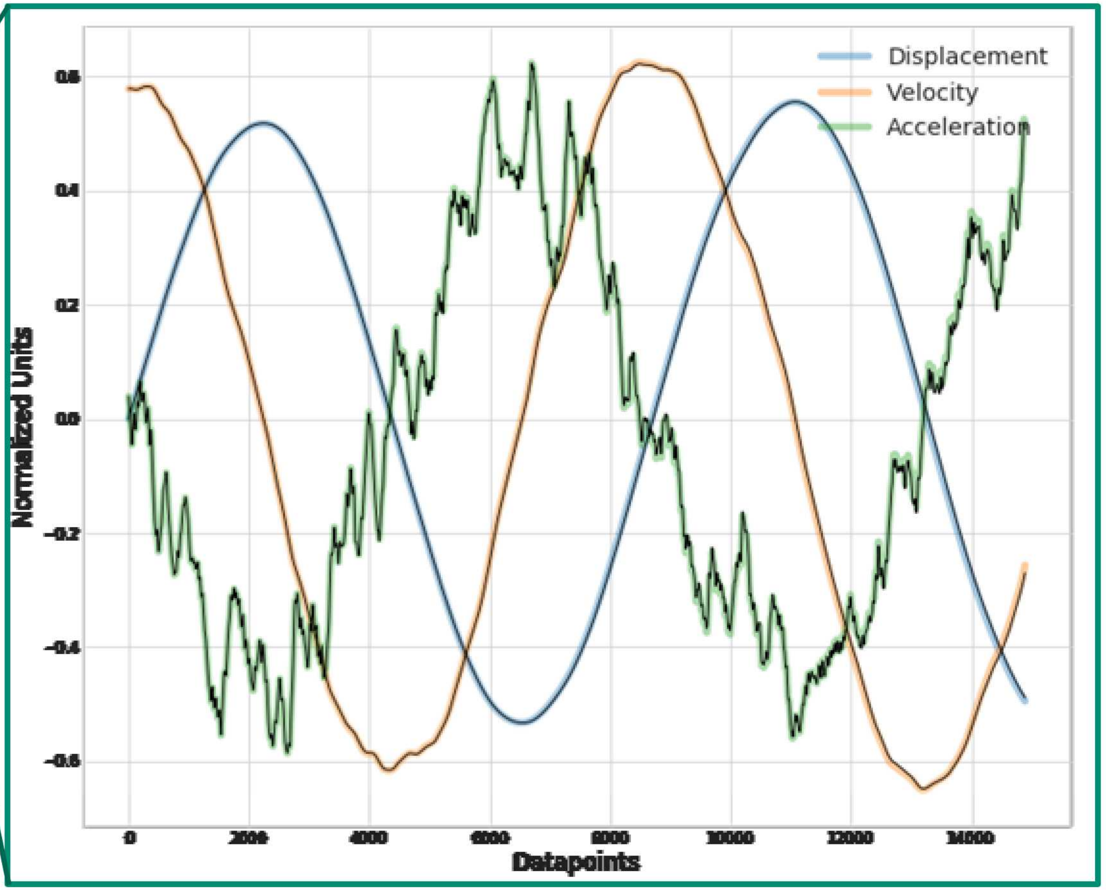


Truth CFD Analysis



Lift Force on Body

Physics Informed Neural Network



Response of single DOF oscillator

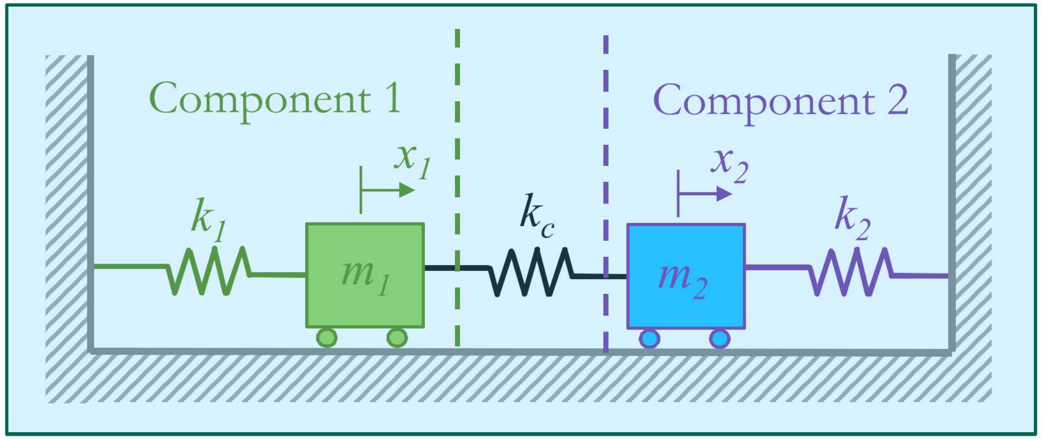
Black – Truth Data | Colored – NN Prediction

Training – Numerical (Python)

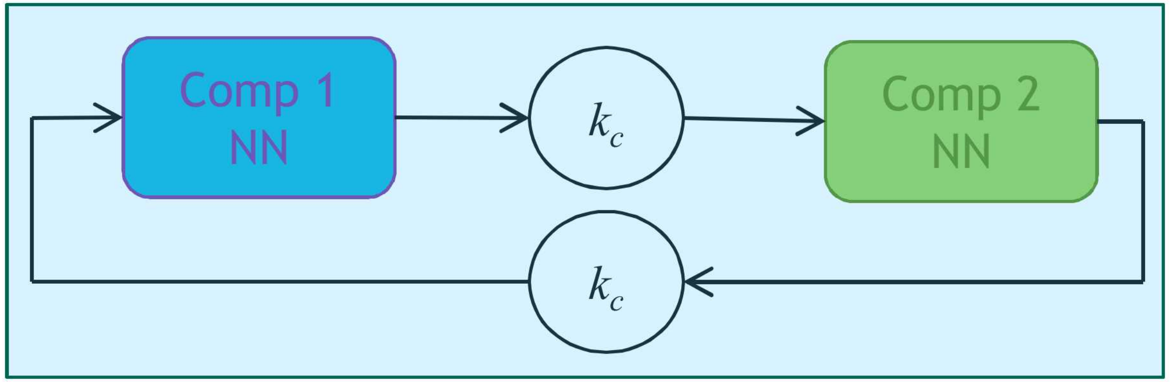
Inputs – Natural Frequency, Initial Conditions, Force

Outputs – Displacement, Velocity, Acceleration

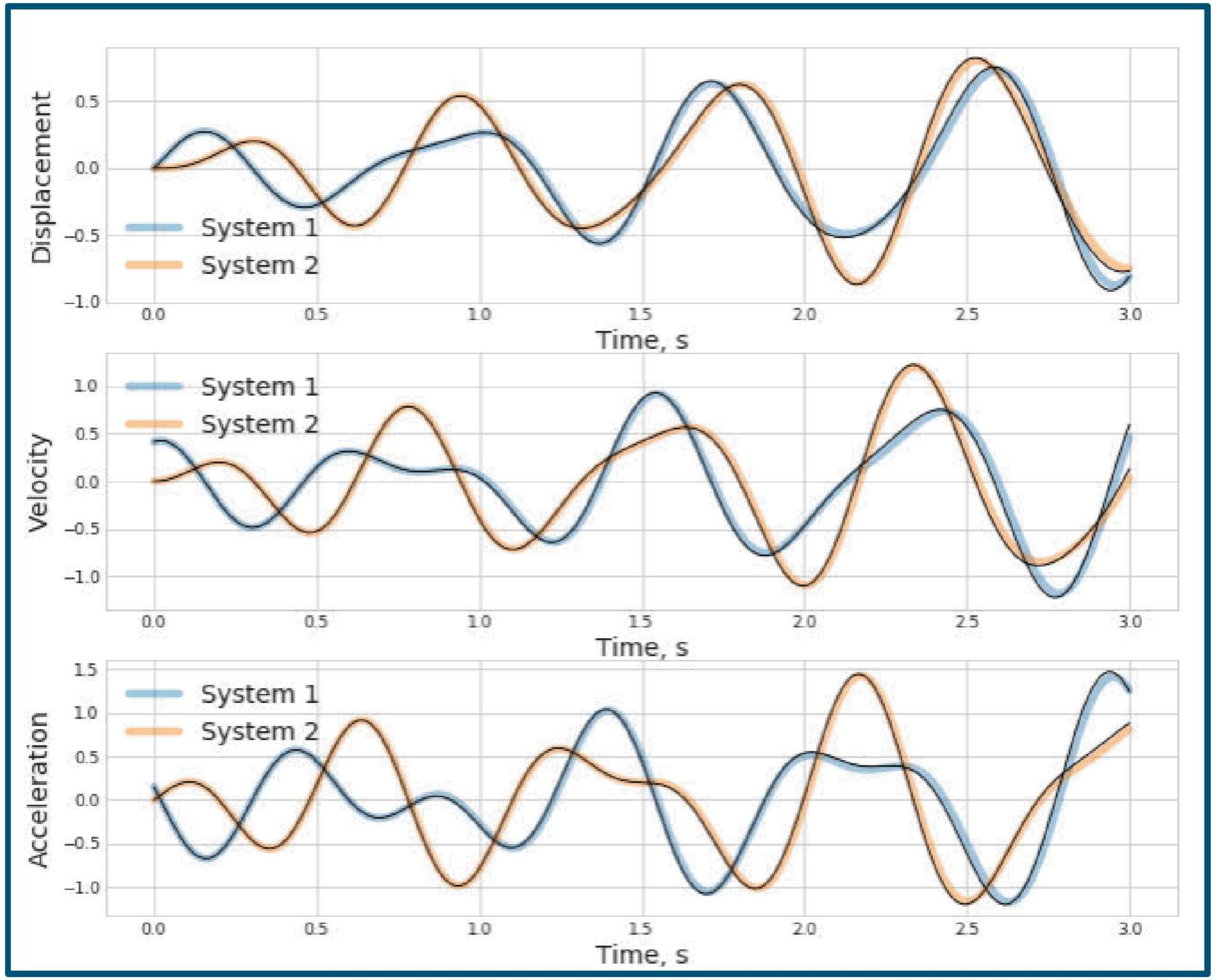
Future work: Physics informed neural networks are a feasible approach to solving two-way coupled problems



Multi-component system under consideration



Step-by-step prediction with output from one NN feeds as input to another



Black – Truth Data | Colored – NN Prediction
Training – Numerical (Python)
Inputs – Natural Frequency, Initial Conditions, Force
Outputs – Displacement, Velocity, Acceleration

Questions?