Using conservation laws to infer deep learning model accuracy of Richtmyer-Meshkov instabilities

ECCOMAS 2022
Deep Learning Approaches for Applied Sciences and Engineering I

Charles F. Jekel, Dane M. Sterbentz, Sylvie Aubry, Youngsoo Choi, Daniel A. White, Jon L. Belof
jekel1@llnl.gov
https://people.llnl.gov/jekel1
https://jekel.me

June 5 – 9, 2022

This work was supported by the LLNL-LDRD Program under Project No. LDRD 21-SI-006.

LLNL-PRES-835760
This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract DE-AC52-07NA27344. Lawrence Livermore National Security, LLC

Lawrence Livermore National Laboratory

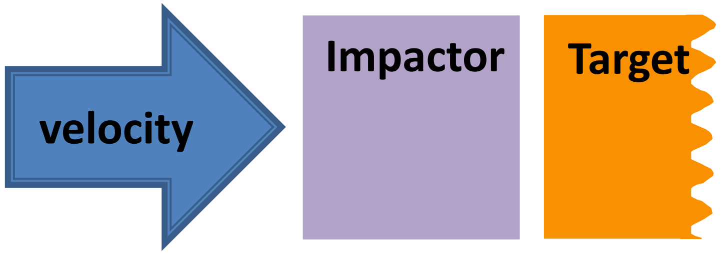
What are Richtmyer-Meshkov or Rayleigh-Taylor instabilities?

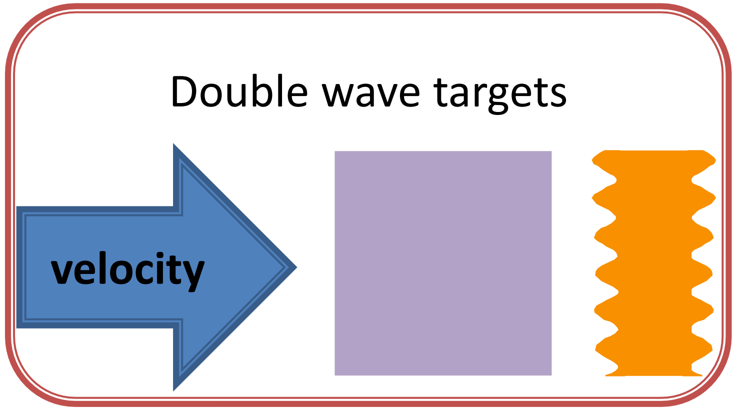
- Rayleigh–Taylor instability occurs at an interface of two different densities [2]
  - Water suspended above oil
- Richtmyer-Meshkov Instability (RMI) is impulsively accelerated
  - Two substances with different density
  - Some initial small perturbation between materials
  - Shock wave through interface causes large "jet-like" growths
  - Various importance and interest (e.g. ICF at NIF [1] [3])
- Our project seeks to 'control' RMI (PI Jon Belof)
  - State of the art experiments and computations
  - **Machine Learning to predict RMI**

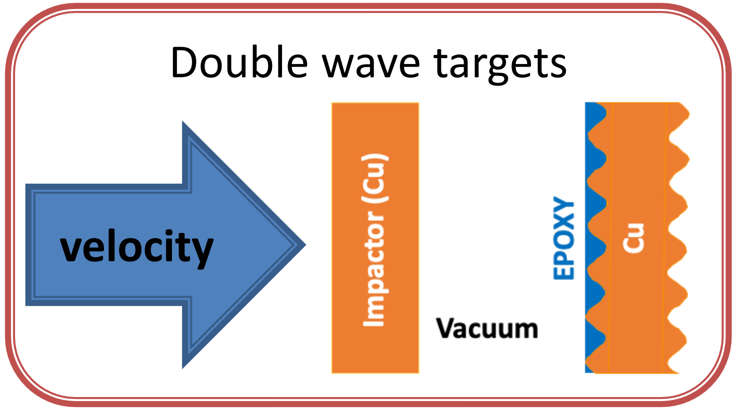
Snapshots of density in time increments of 0.1μs from left to right as an RMI forms.

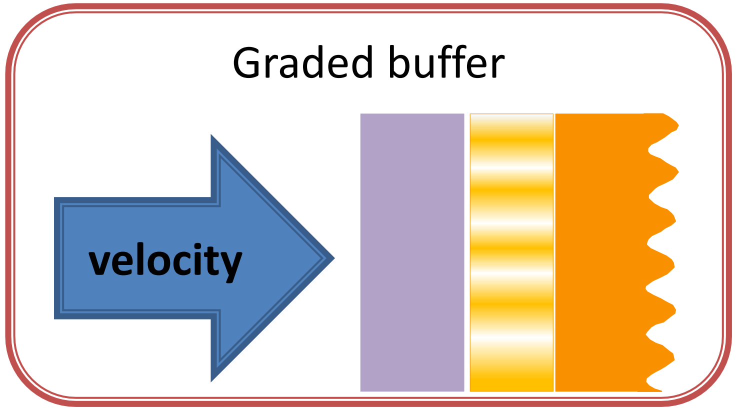
Various Impact experiments to design for RMI

- Seeking designs that maximize RMI
- Also attempting to mitigate known RMI

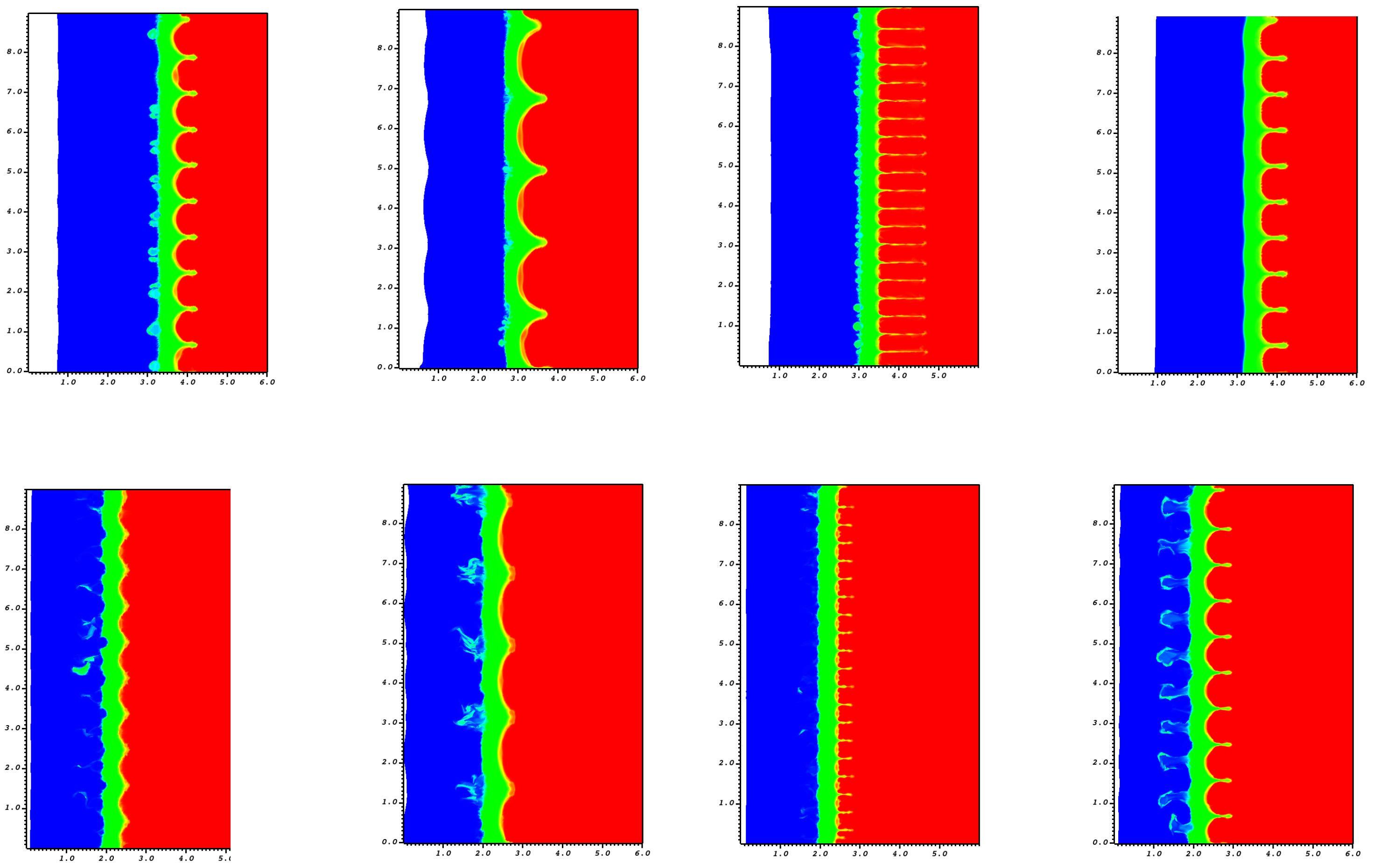


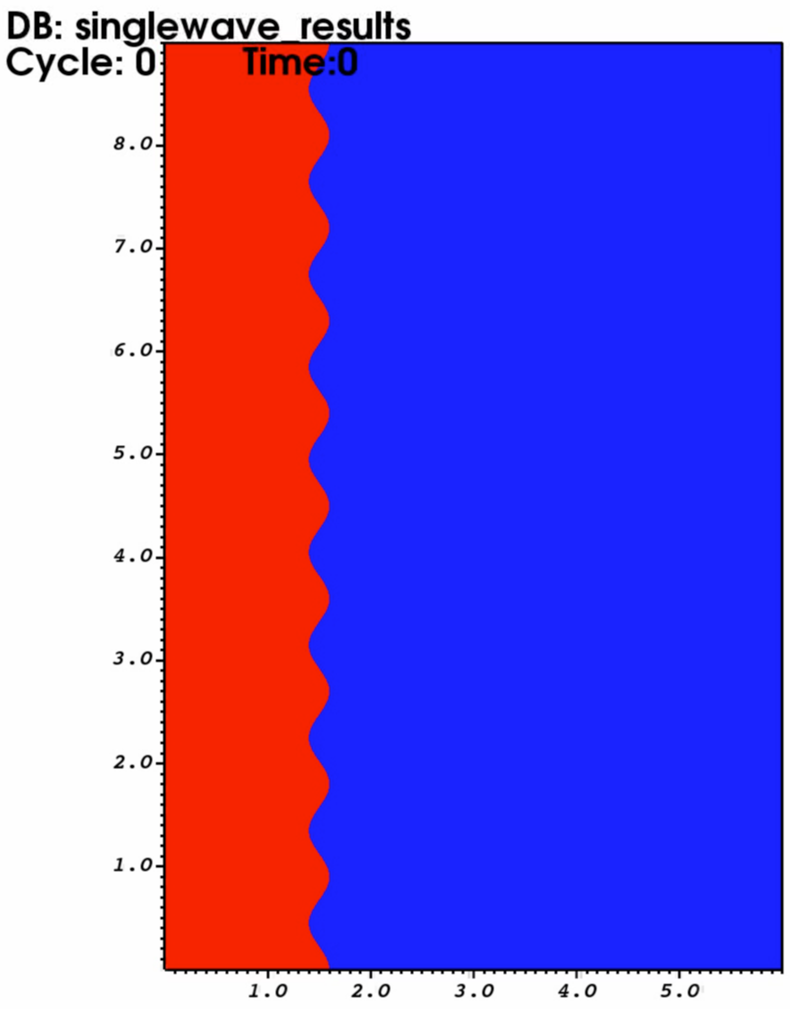




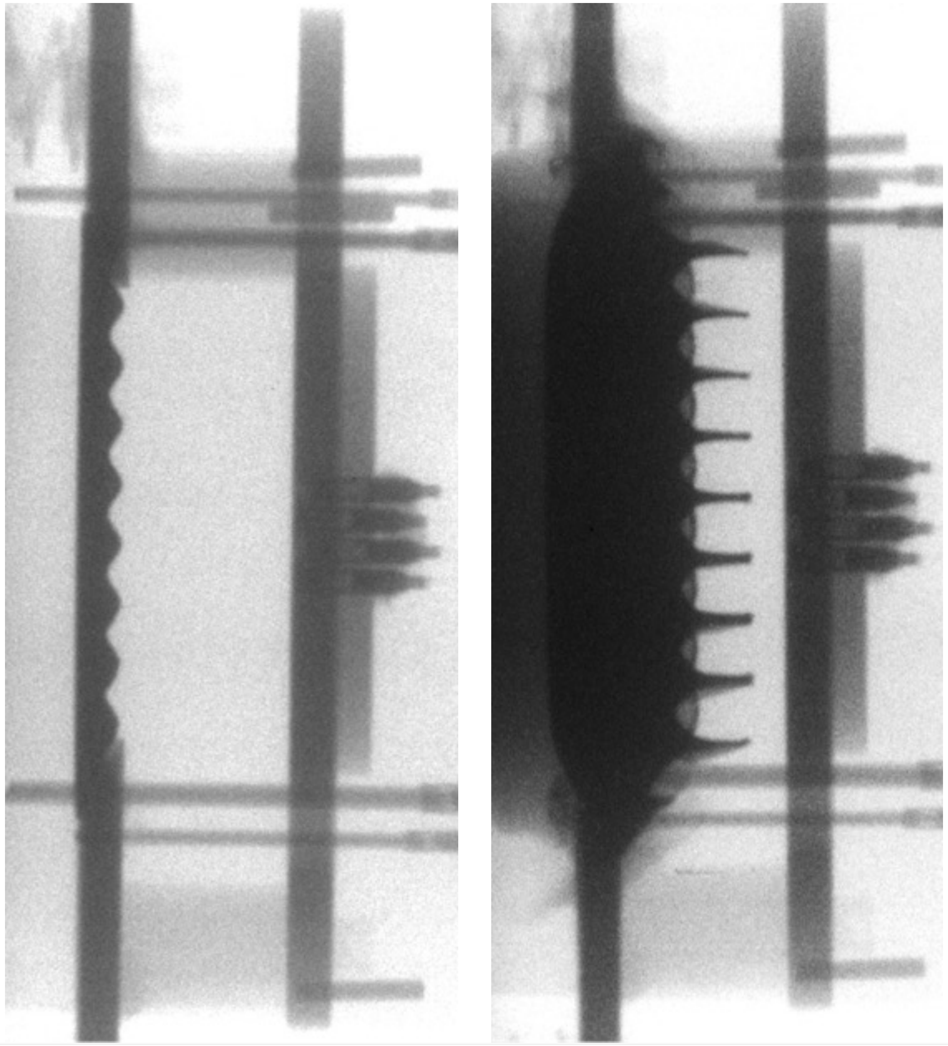


Simulated RMI at the same impact velocity
Changing impact materials and initial amplitude

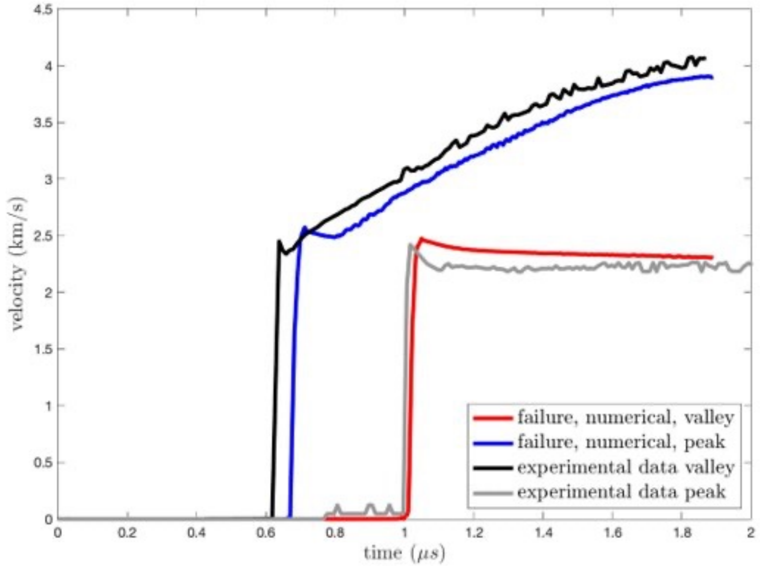




How well do simulations agree with experiments?



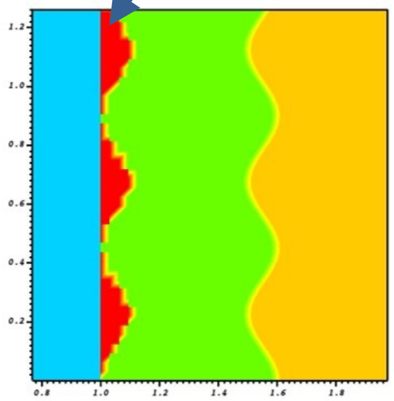
- HEAF gas gun experiments at LLNL
  - 9cm diameter
  - Hector Lorenzana, Jeff Nguyen, Mike Armstrong

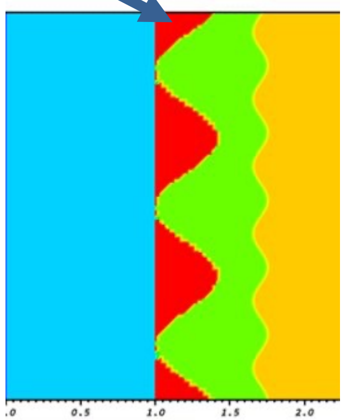


Comparison with sinusoidal wave.

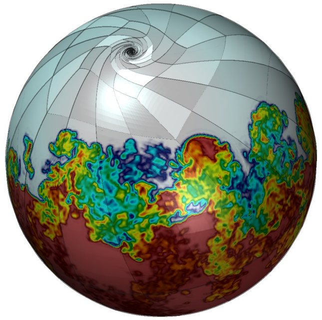
A parameterized impactor simulation to study RMI

- 3 parameters to change
  - Changes perturbation in ”Target”
  - B, Q, S





$$x = B \cos\left(\frac{2\pi Q y}{9} - s\pi\right)$$



- Machine learning ready LLNL tools!
  - MARBL / BLAST: ALE Hydrodynamics [4] [5]
    - https://computing.llnl.gov/projects/blast
  - Ascent: fast ray tracing ‘images’
  - Merlin: HPC workflow management

Ascent

Merlin

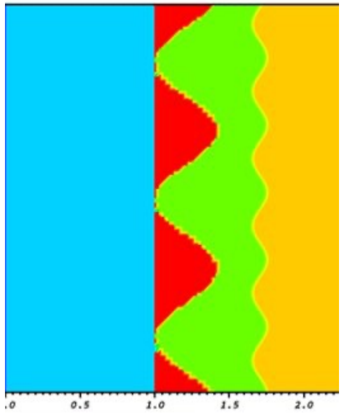
Materials of simulation

- — Copper impactor, high initial velocity
- — Lucite, used to fill in target’s perturbation
- — Copper target, zero initial velocity
- — Air

Machine learning model overview

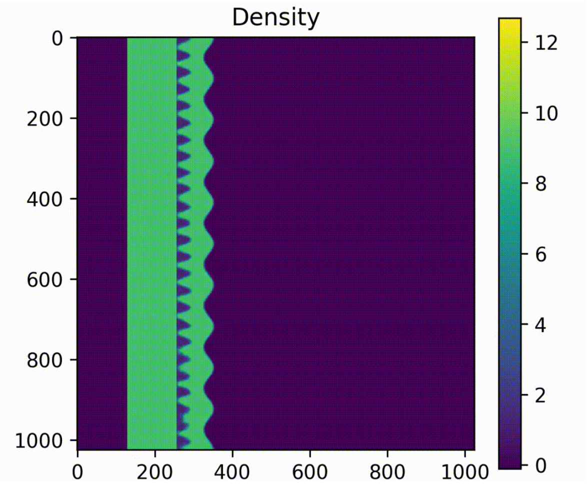
- Model predicts full RMI formation
  - **Input**: Initial conditions
  - **Output**: Full field response
- Why do this?
  - Use ML model to quickly explore designs
  - Optimization on the ML model is fast

3 input parameters defining initial conditions (perturbation in green target)



Machine Learning Model

Entire time dependent density field prediction



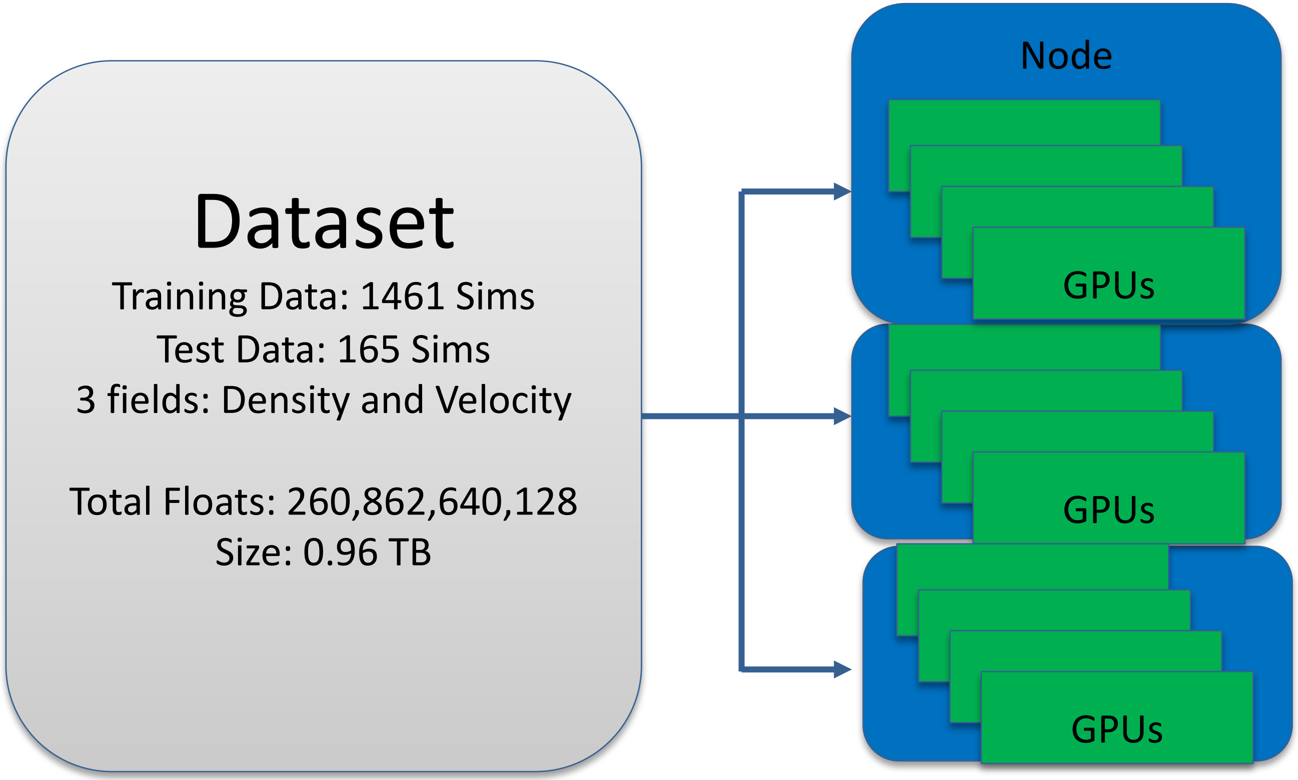
Machine learning dataset at a glance

- For the three parameter study
  - 1,600 simulations
    - 600 Lassen/Sierra node hours
  - 51 times steps per simulation
  - 5 output fields
    - Density
    - Velocity X & Y
    - Energy
    - Materials
  - 1024 x 1024 “pixels”
  - 427,819,008,000 single precision floats
  - 1.6 TB
- Plan to release open datasets!
  - Please reach out to be notified
  - jekel1@llnl.gov

```
12G     dataset_000.h5
12G     dataset_001.h5
12G     dataset_002.h5
12G     dataset_003.h5
12G     dataset_004.h5
12G     dataset_005.h5
12G     dataset_006.h5
12G     dataset_007.h5
12G     dataset_008.h5
12G     dataset_009.h5
12G     dataset_010.h5
12G     dataset_011.h5
12G     dataset_012.h5
12G     dataset_013.h5
12G     dataset_014.h5
11G     dataset_015.h5
11G     dataset_016.h5
11G     dataset_017.h5
11G     dataset_018.h5
11G     dataset_019.h5
11G     dataset_020.h5
```

143 - 12 GB h5 files

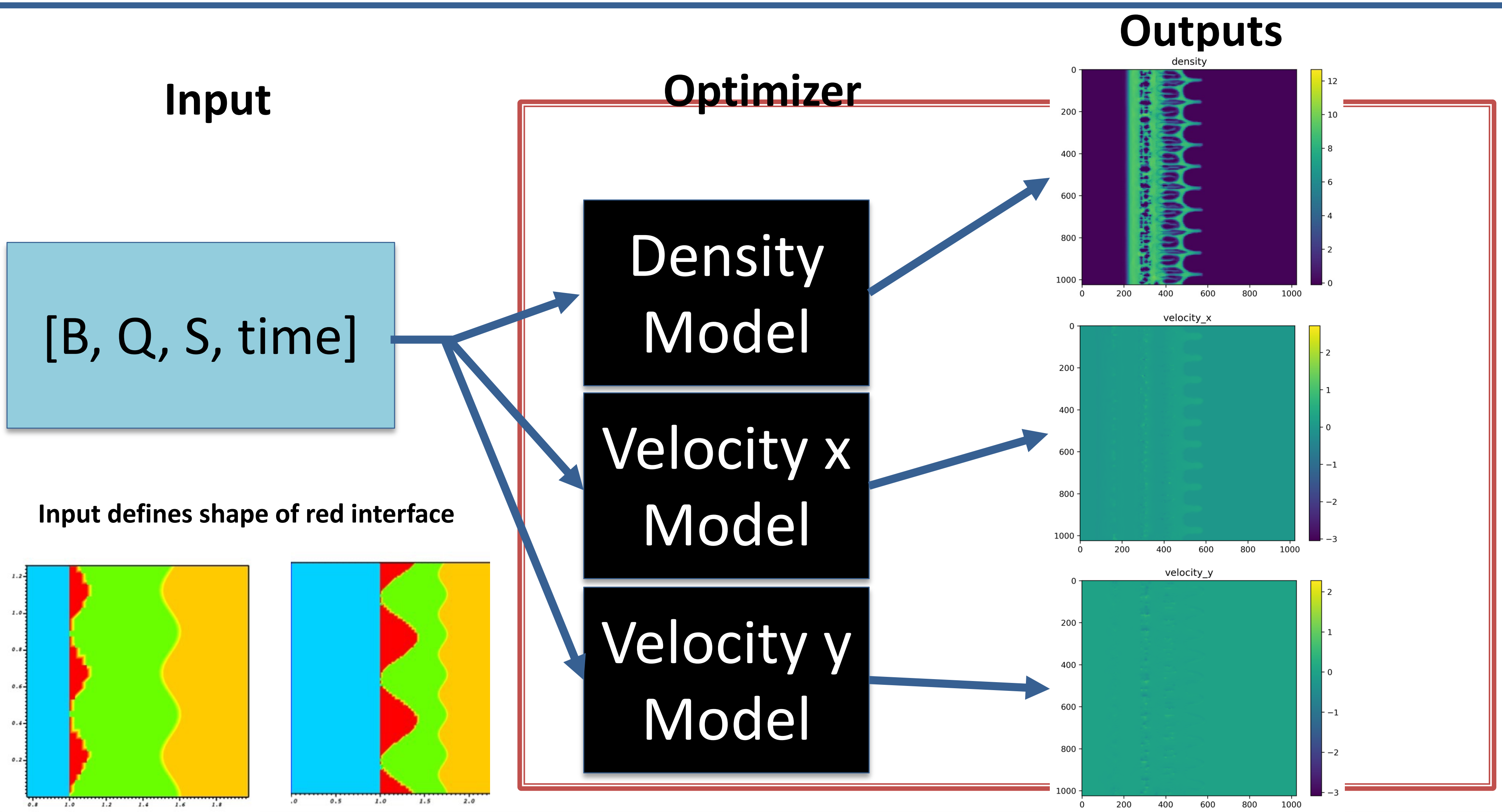
Distributed data model training paradigm



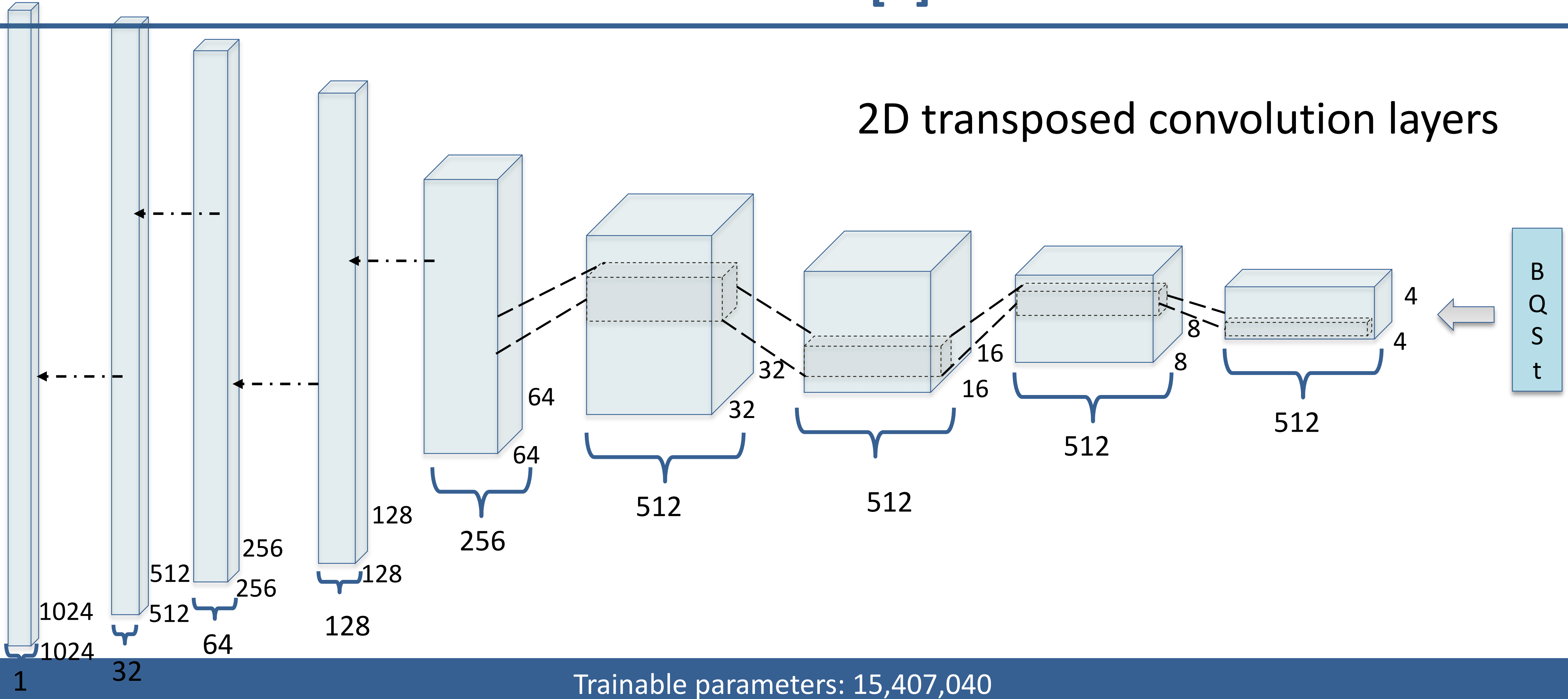
- We can train the ML model in one hour using **160 GPUs**
- Dataset split among multiple nodes
- Each GPU
  - receives unique fraction of dataset
  - Duplicate copy of model and optimizer
  - MPI syncs model and optimizer states
- GPU memory limited
  - Can only generate N number of 1024x1024 'images' at a time
  - More GPUs -> faster training and inference throughput

Simultaneous training of separate models for each field

- 3 models
- 1 optimizer
- 1 loss function
- 4 inputs
- 1024x1024 output for each field



The ML model for each physical field
See 'Generator' model from DCGAN [6]

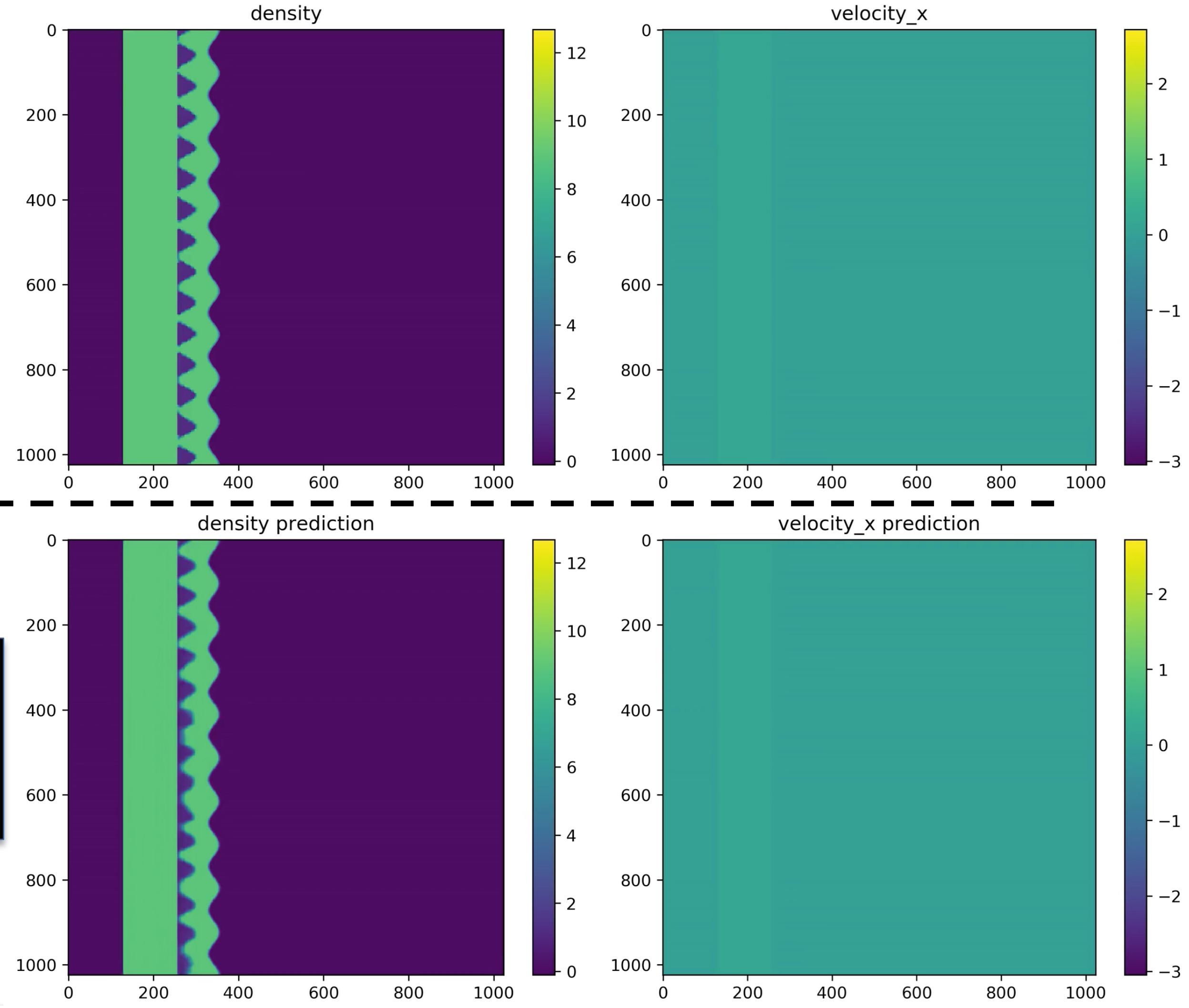


Trainable parameters: 15,407,040

Best left-out ‘test’ simulation comparison

MARBL simulation

ML Model

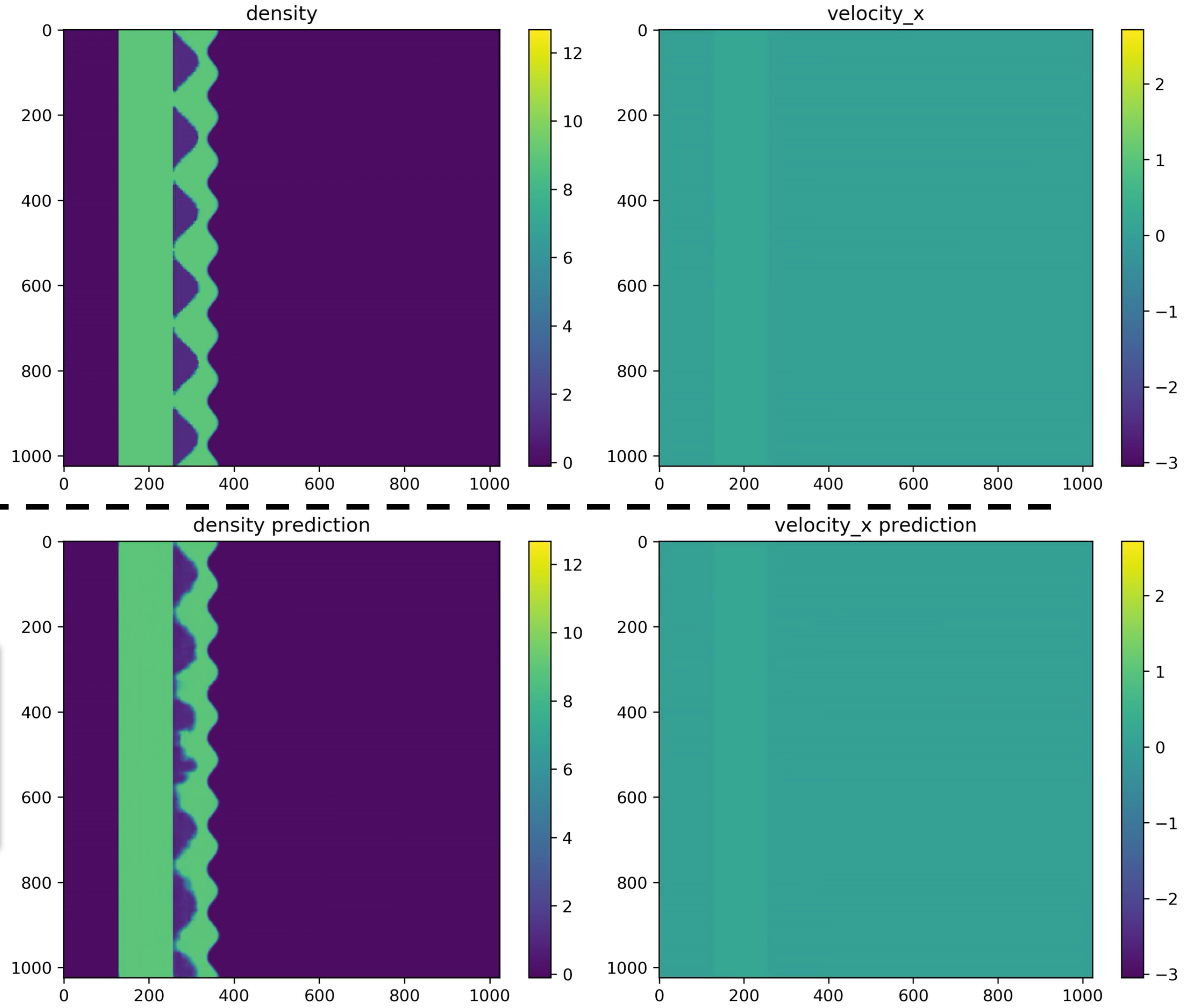


- Lowest L1 error in test set
- Epoch 500
- MARBL simulation **top**
- ML prediction **bottom**

Worst left-out ‘test’ simulation comparison

MARBL simulation

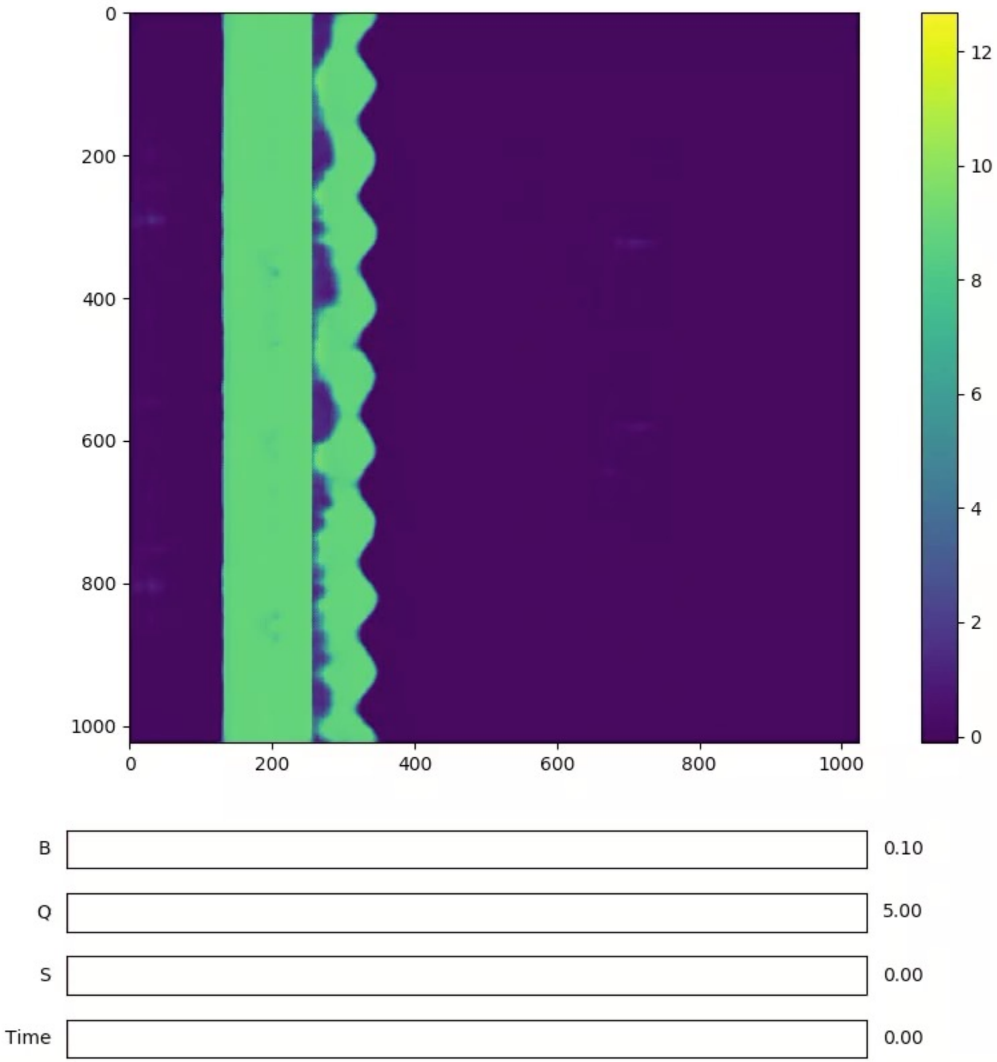
ML Model



- Highest L1 error in test set
- Epoch 500
- MARBL simulation **top**
- ML prediction **bottom**

Interactively exploring the ML model in the entire design space

- Live visualization from ML model
- Corners of design space yield worst visual
- From HPC dataset to laptop visualization
- Quickly step forward and backward in time



How well can you trust the ML model's predictions?

- Trying to use first principles to infer the accuracy of our predictions
  - Continuity equation
  - Conservation of mass
  - Conservation of momentum

- These metrics can be calculated without running a simulation

- Simulations are all closed domain, so these equations should be preserved

How well can you trust the ML model's predictions?

- Trying to use first principles to infer the accuracy of our predictions
  - Continuity equation
  - Conservation of mass
  - Conservation of momentum
- These metrics can be calculated without running a simulation
- Simulations are all closed domain, so these equations should be preserved

- Continuity Equation

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \boldsymbol{u}) = 0$$

- Mass and Momentum as functions of time

$$m(t) = \frac{1}{n_y}\frac{1}{n_x}\sum_i^{n_y}\sum_j^{n_x} \rho_{i,j}(t)$$

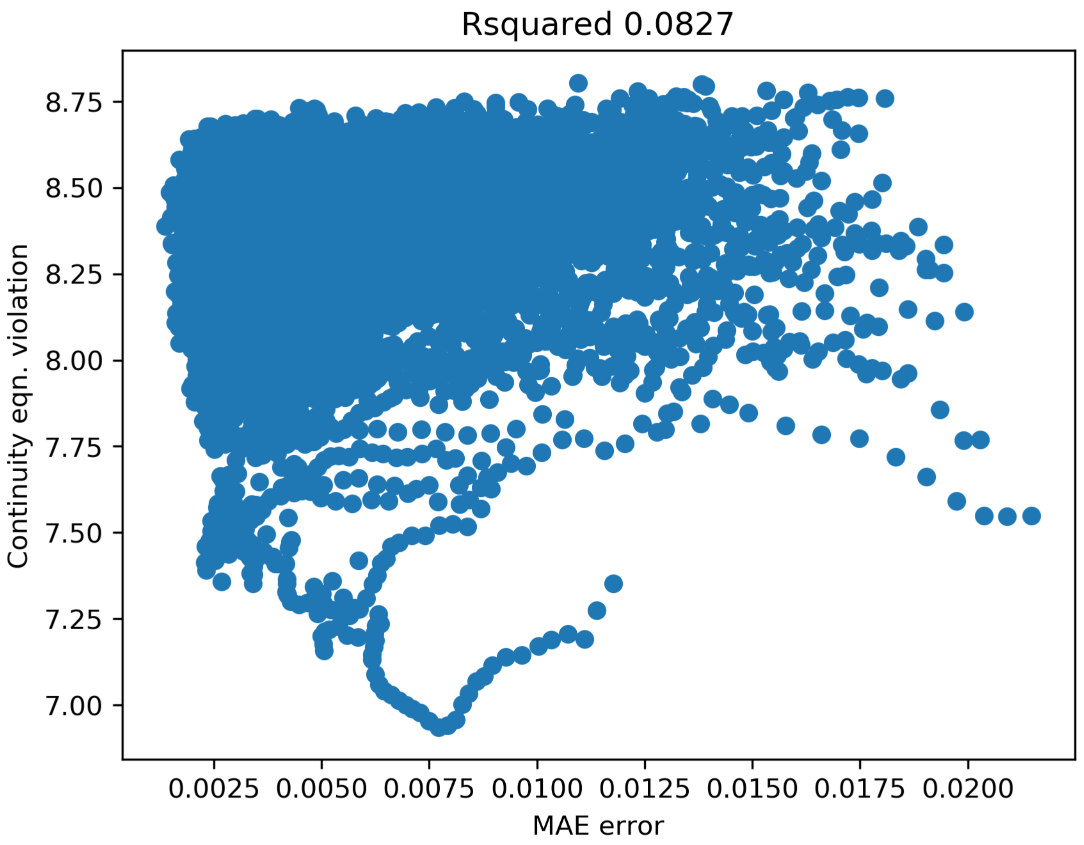
$$p(t) = \frac{1}{n_y}\frac{1}{n_x}\sum_i^{n_y}\sum_j^{n_x} \nabla \cdot \left(\rho_{i,j}(t)\boldsymbol{u}_{i,j}(t)\right)$$

- Variance of mass and momentum

$$\mathrm{Var}(\psi(t)) = \frac{1}{n_t}\sum_i^{n_t}\Big(\psi(i) - \mathrm{Mean}(\psi(t))\Big)^2$$

Correlation plot of MAE vs Continuity Equation Violation (L1) on left-out simulations

- Strong correlation would give us some predictive capability
- This is not good enough!



Physics informed training via soft constraint

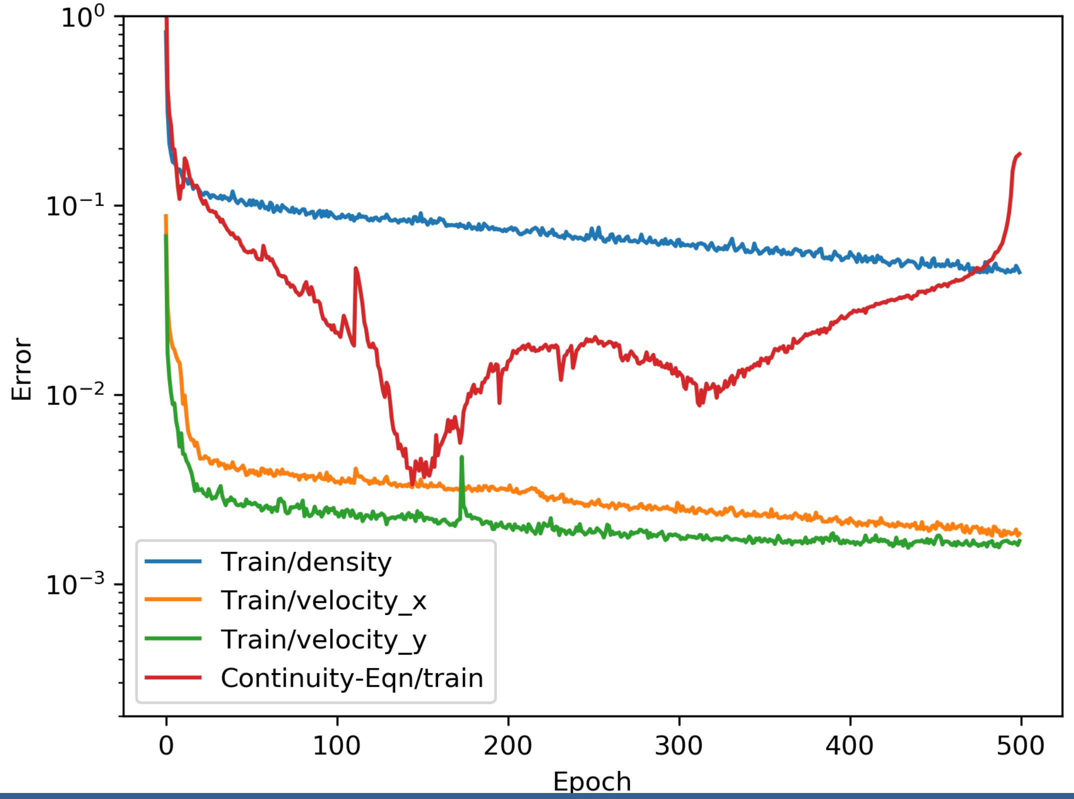
- What happens if you put the continuity equation violation into the loss function? [7]

- Training is very difficult
  - The results are sensitive to your penalization parameter
  - Mean absolute error (L1) plus penalized continuity equation violation

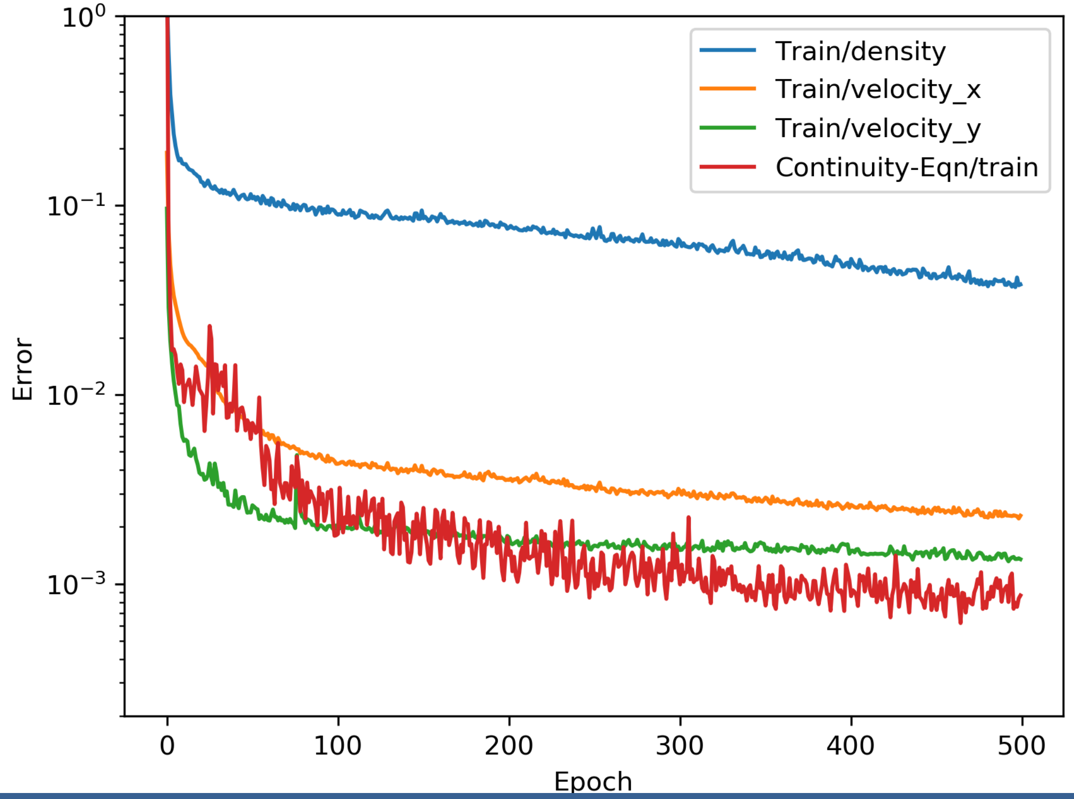
$$\frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| + \lambda_c\left|\frac{\partial\rho}{\partial t} + \nabla\cdot(\rho\boldsymbol{u})\right|$$

Training loss curves with and without physics-guided loss

Mean absolute error loss



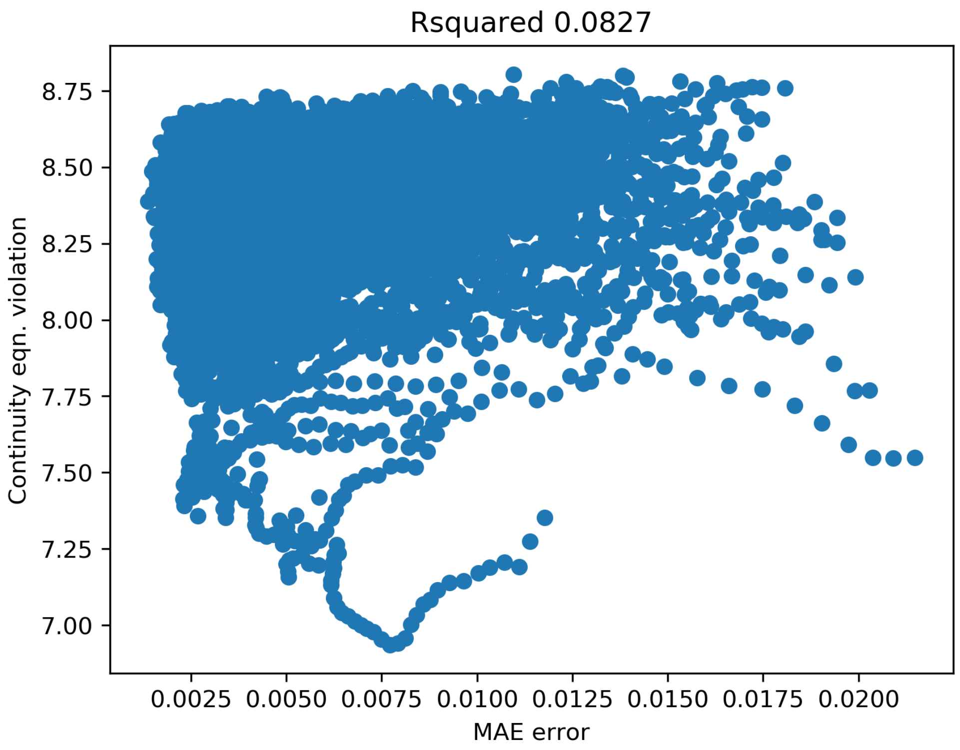
With continuity equation violation



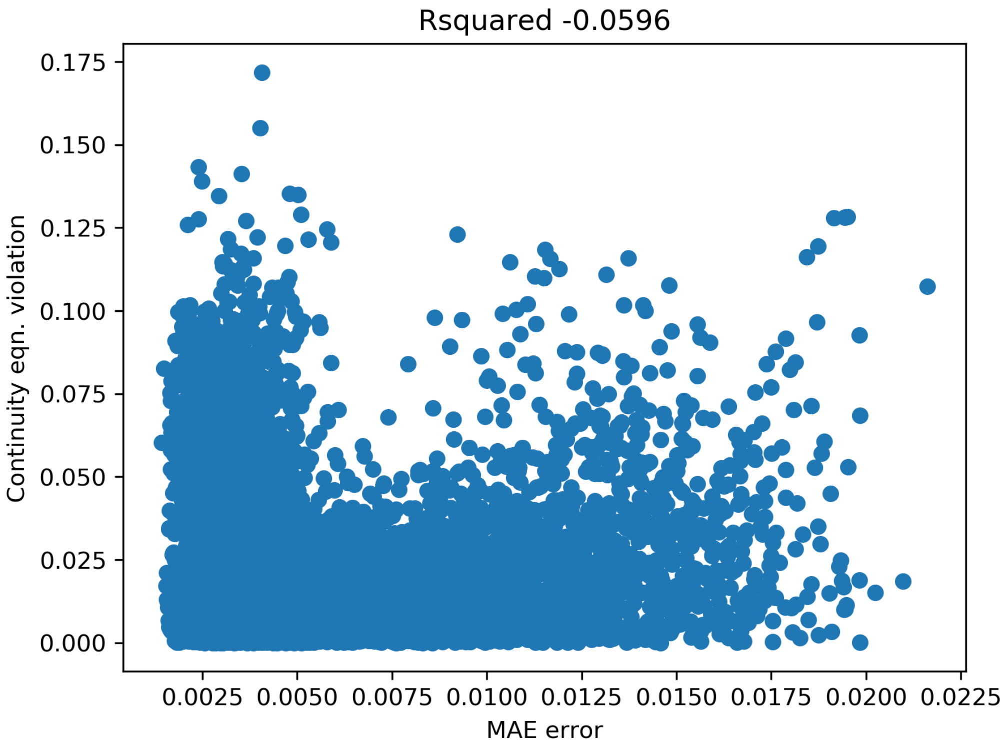
Continuity equation violation (RED) is much better in training when added as a loss function.
Errors in density and velocity were relatively the same.

Left-out correlation with and without physics-guided loss

Mean absolute error loss



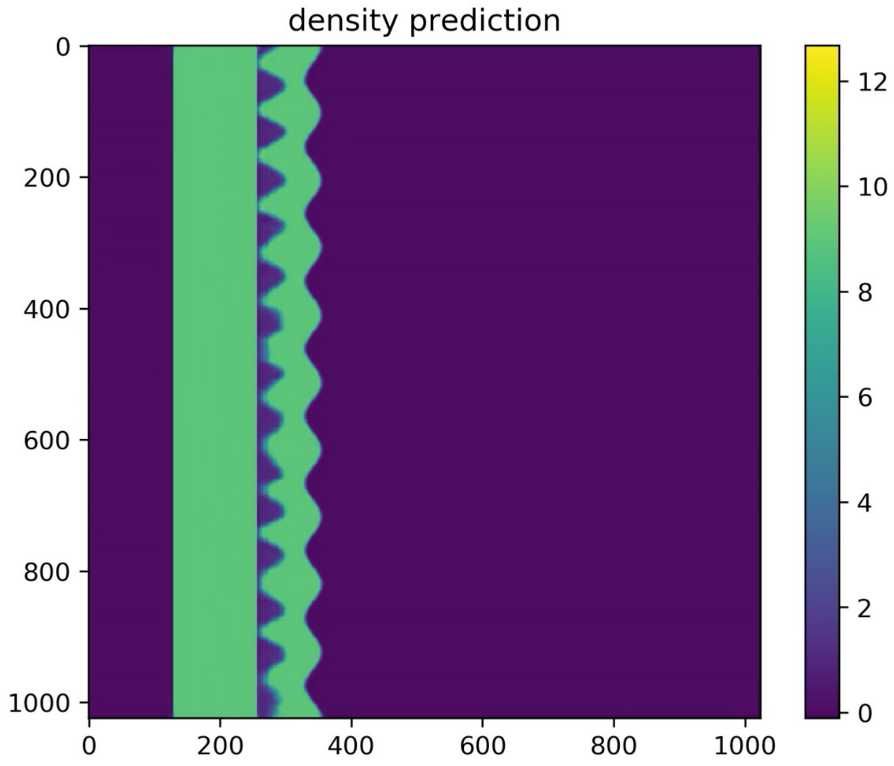
With continuity equation violation



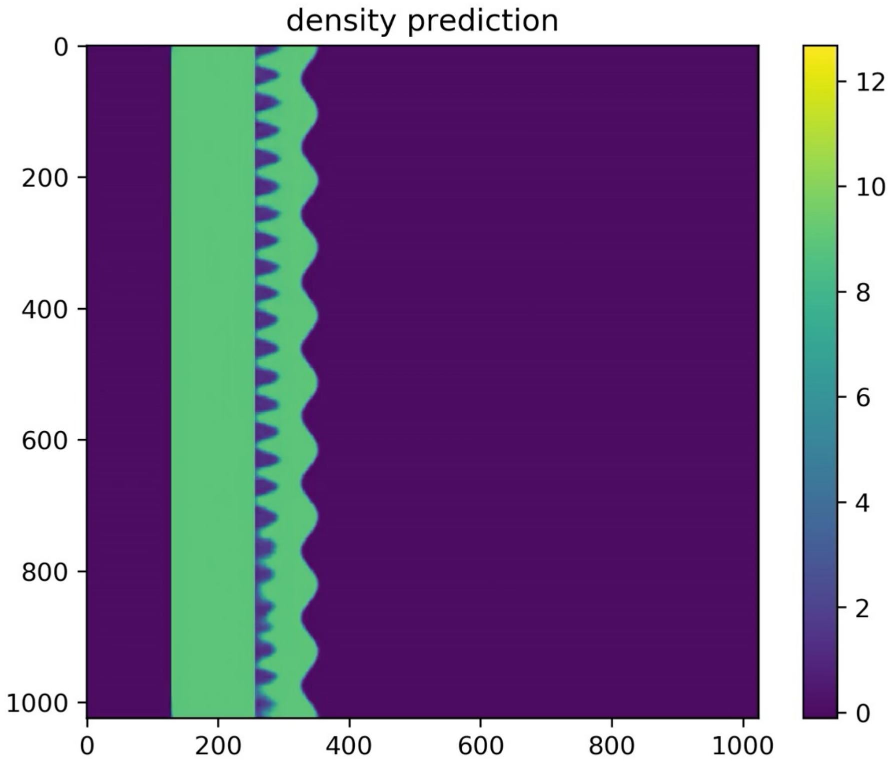
Continuity equation violation is much better with continuity equation penalty (right), however MAE error is relatively unchanged.

Best left-out simulations with and without physics-guided loss

Mean absolute error loss



With continuity equation violation



Similar level of detail on these different predictions.

Conclusions

- ML modeling of RMI hydrodynamic simulations
  - Predictions are 10,000 times faster than simulation
  - allows for quick visualization of a design space
  - models can be ‘run backwards’ and inverted
- Using conservation laws to infer deep learning ML model accuracy
  - Strong correlation early in training
  - Weak correlation with finalized models
- Continuity equation penalty into loss function
  - Reduced continuity equation violation
  - Did not improve on prediction accuracy
- Open datasets and code coming!
- Slides will go live on https://jekel.me/cv under “Presentations”

References

1. Zylstra, A.B., Hurricane, O.A., Callahan, D.A. *et al.* Burning plasma achieved in inertial fusion. *Nature* **601,** 542–548 (2022). https://doi.org/10.1038/s41586-021-04281-w

2. Park HS, Lorenz KT, Cavallo RM, Pollaine SM, Prisbrey ST, Rudd RE, Becker RC, Bernier JV, Remington BA. Viscous Rayleigh-Taylor instability experiments at high pressure and strain rate. Physical review letters. 2010 Apr 2;104(13):135504.

3. T.R. Desjardins, C.A. Di Stefano, T. Day, *et al.* A platform for thin-layer Richtmyer-Meshkov at OMEGA and the NIF, *High Energy Density Physics*, Volume 33, 2019, 100705, ISSN 1574-1818, https://doi.org/10.1016/j.hedp.2019.100705

4. R. Rieben, Poster: The MARBL multi-physics code, in: Exascale Computing Pro ject Annual Meeting, 2020. doi:10.13140/RG.2.2.12326.14403.

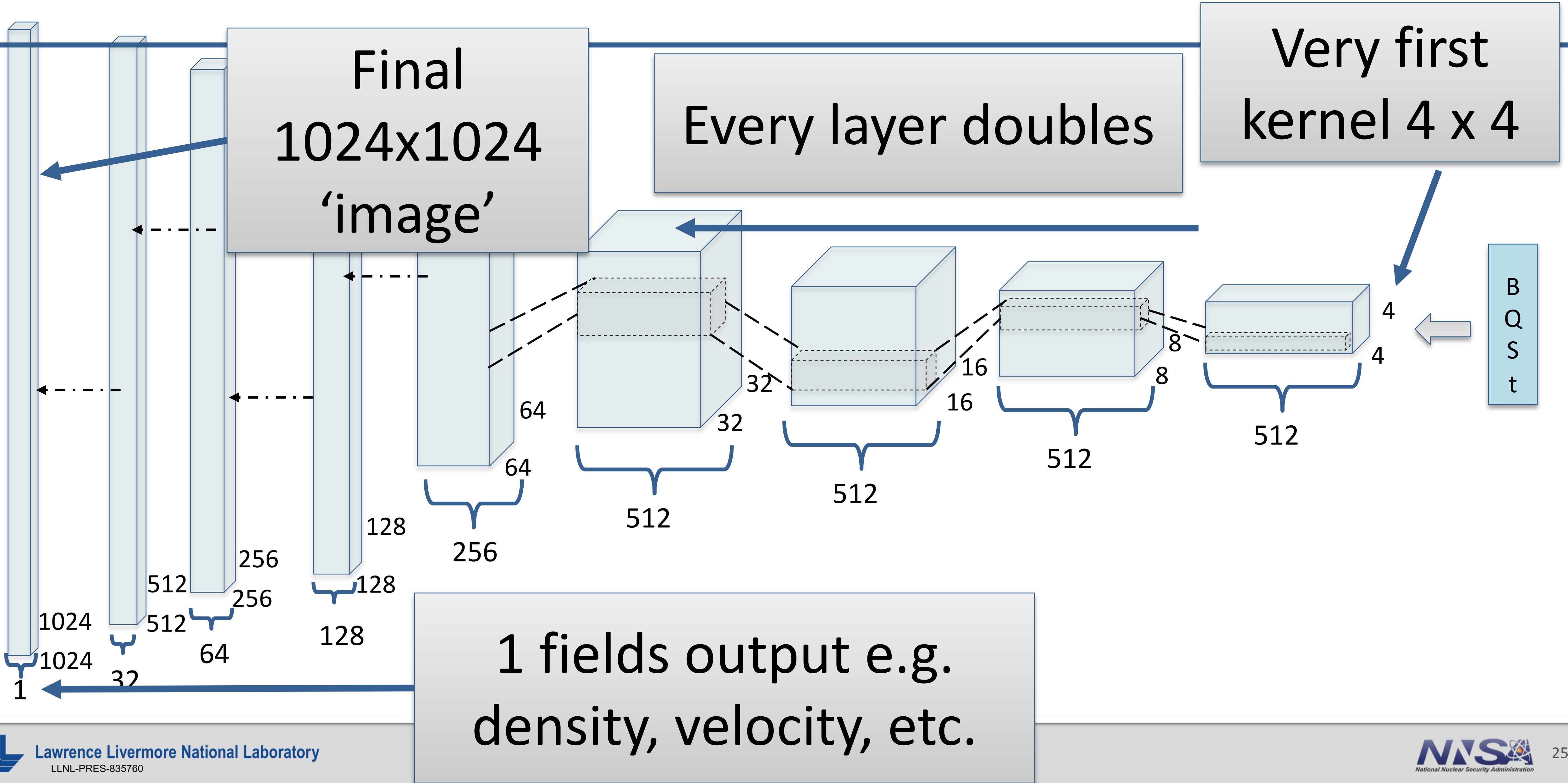
5. R. W. Anderson, V. A. Dobrev, T. V. Kolev, R. N. Rieben, V. Z. Tomov, High-order multi-material ALE hydrodynamics, SIAM J. Sci. Comput. 40 (1) (2018) B32–B58.

6. Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv:1511.06434, 2015.

7. Raissi, Maziar, Paris Perdikaris, and George E. Karniadakis. "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations." Journal of Computational Physics 378 (2019): 686-707.

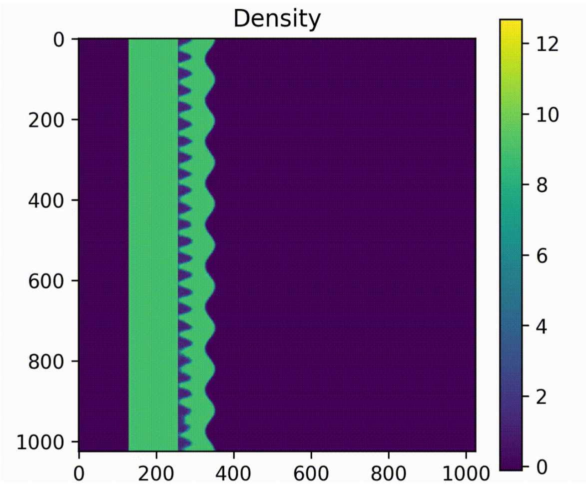
Lawrence Livermore National Laboratory

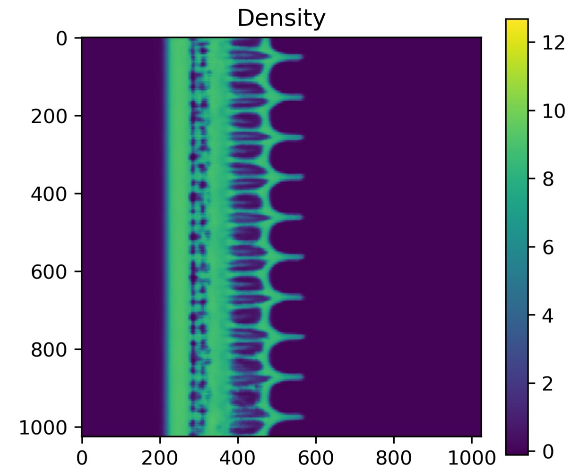
Layer by layer progression

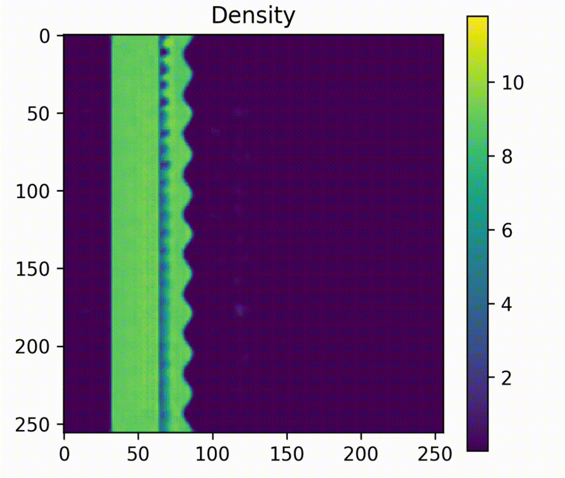


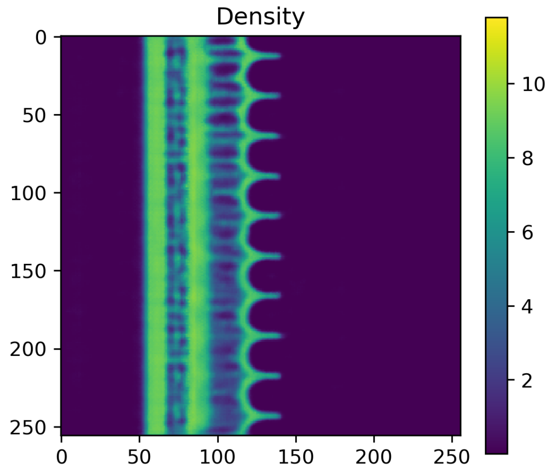
More pixels gave us much more detail
but significantly increase computation demand

1024x1024









256x256

Lowest MAE from each left-out 'test' set shown

Data compression of the ML model (for two fields)

- 1626 simulations
- 171 billion floats
- Exported model is 178 MB
- **4,000 to 1 compression**
- Brings data visualization from HPC world to laptop world
- With losses to accuracy/detail