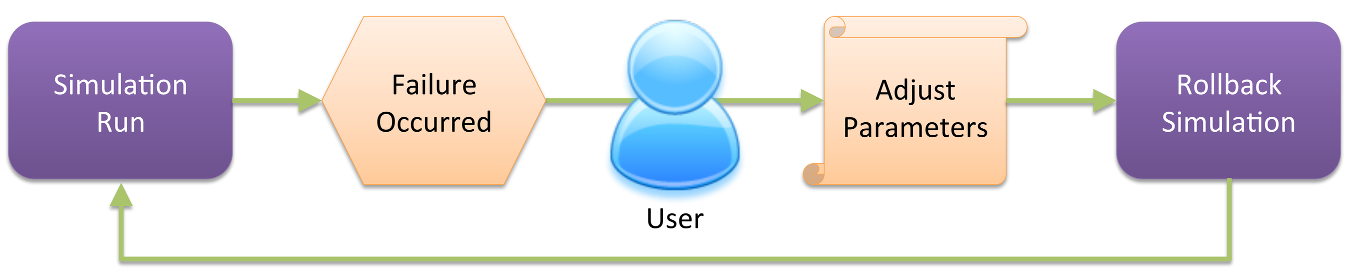
A Supervised Learning Framework for Arbitrary Lagrangian-Eulerian Simulations

Ming Jiang, Brian Gallagher, Joshua Kallman, Daniel Laney
Lawrence Livermore National Laboratory

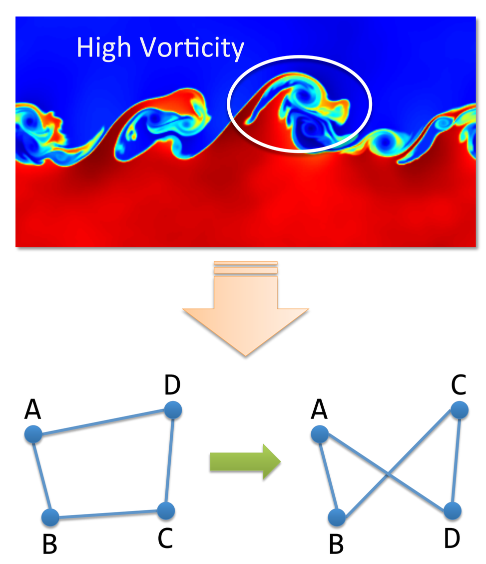
CASC

Introduction

- The **Arbitrary Lagrangian-Eulerian (ALE)** method is used in a variety of engineering and scientific applications for enabling multi-physics simulations.
- Unfortunately, the ALE method can suffer from **simulation failures** that require users to adjust parameters manually in order to complete a simulation.
- Finding the right **ALE strategy** (which parameters to adjust, when and by how much) is a trial-and-error process that can be disruptive and time consuming.

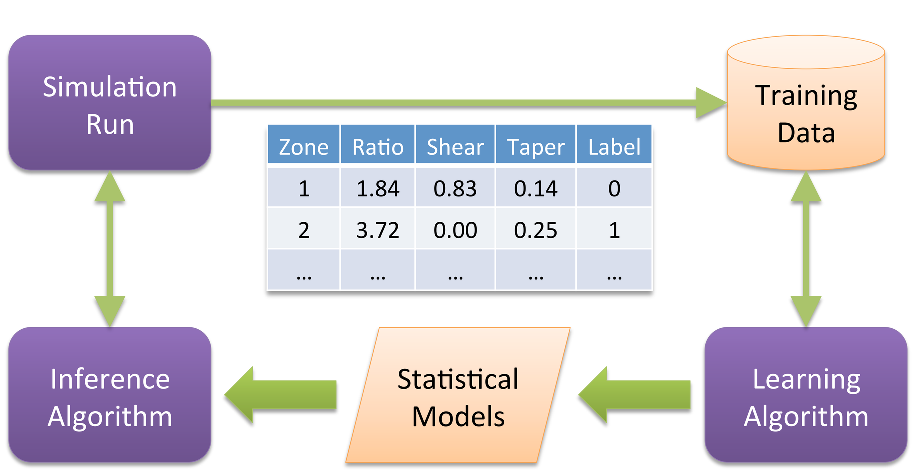


- In general, the ALE method consists of two phases:
  1. An arbitrary number of Lagrangian steps
  2. An advective remap phase to relax the mesh
- There are numerous parameters that control how the mesh moves during the remap phase.
- There are two general ways that failures can occur:
  - Mesh geometry distortions: **mesh zone tangling**
  - Physical quantity distortions: **anomalous hot spots**



Goal: Semi-automate the manual process for developing ALE strategies, by applying machine learning to predict ALE simulation failures.

Supervised Learning Framework



Zone	Ratio	Shear	Taper	Label
1	1.84	0.83	0.14	0
2	3.72	0.00	0.25	1
…	…	…	…	…

- Novel workflow for **ALE + ML**:
  - Collect labeled training data from ALE simulation run
  - Apply learning algorithm to derive statistical models for predicting failures
  - Use models in inference algorithm for ALE strategy in future simulation runs

Learning Representation

- Map ALE simulation domain onto supervised learning formulation
- Given zone z at time t:
  - A learning instance i is a pair (z,t)
  - Features $X(i)$ are metrics of (z,t)
  - Class of i is the failure state of z at t
- **Features**: 16 mesh quality metrics for 2D quadrilateral zones

Generating Labeled Data

+ Positive class examples:
  - Failed zones: negative side or corner

− Negative class examples:
  - Non-failed zones: everything else

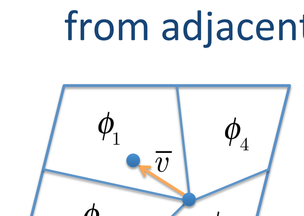
Predicting Failures

- Need to predict failures before they occur to apply mesh relaxation
- **Our approach**: train a regression model using time history of failures
- Given failure instance (z_f, t_f), we define labels for **soon-to-fail** zones:

$$\phi(z_f,t) = 1 - \frac{(t_f - t)}{T},\ s.t.\ \phi : [t_f - T, t_f] \rightarrow [0,1]$$

ALE Code Integration

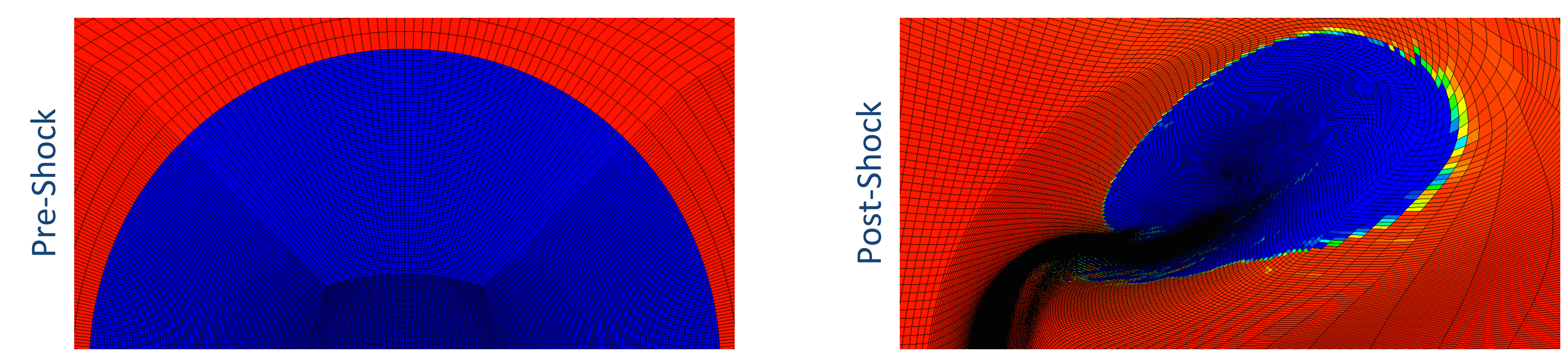
- Learning algorithm: **random forest**
- We developed LAGER library:
  - Integrates KULL with scikit-learn
  - Nodal relaxation: average predicted ϕ from adjacent zones z_a of node n

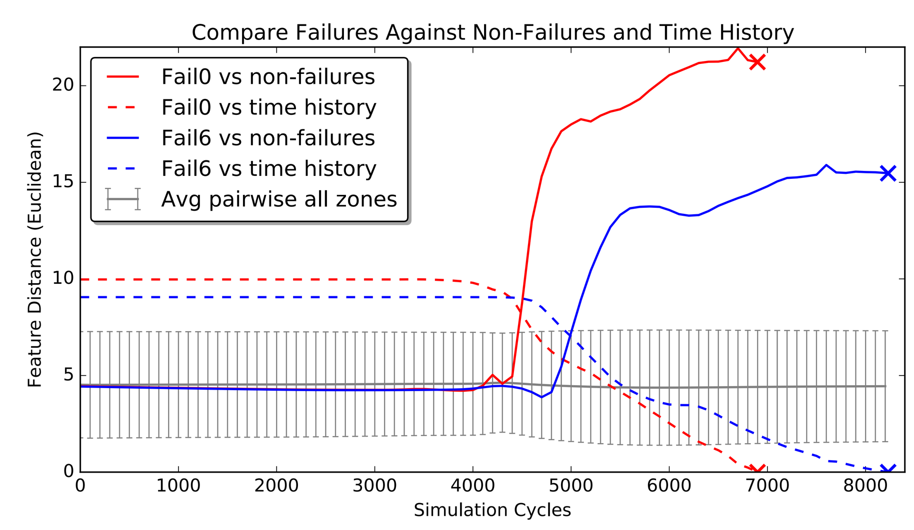


$$\Phi(n,t) = \frac{1}{N}\sum_{a=1}^{N}\phi(z_a,t)$$

Experimental Results

- Bubble Shock: a well-known ALE test problem modeling a planar shock traveling through a spherical helium bubble in air from the right side of the geometry.





$$\delta(i,j) = \left\|\hat{X}(i) - \hat{X}(j)\right\|^2,\ s.t.\ \hat{X}(i) = \frac{(X(i) - \mu_i)}{\sigma_i}$$

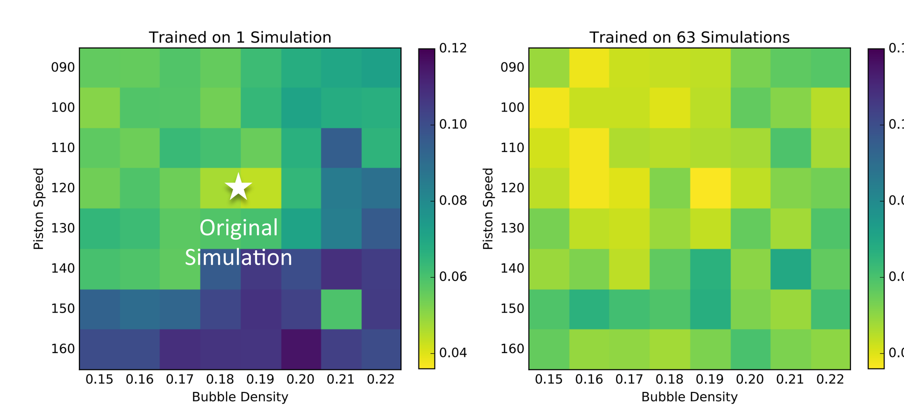
$$\Delta(z,t) = \frac{1}{M}\sum_{m=1}^{M}\delta((z,t),(z_m,t))$$

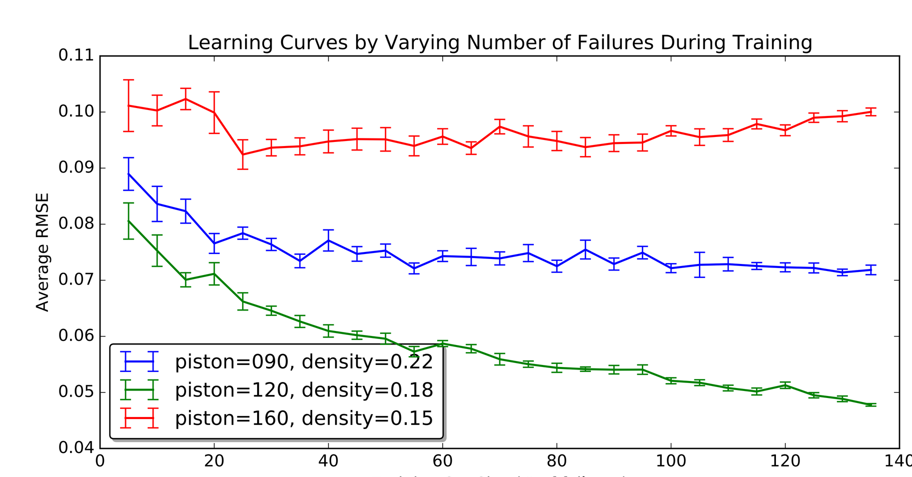
Framework Evaluation

- Evaluate framework design choices:
  - Are features useful for distinguishing failures from non-failures? (*see graph*)
  - How effective is our procedure for generating labeled data? (*see paper*)
  - Is time history of failures suitable for predicting failures? (*see graph*)
- We compare failures against: 1) non-failures, 2) other failures, and 3) time histories, using **feature distance** δ based on mesh quality metrics

Learning Evaluation

- For performance evaluation, we use data from (64) other simulations:
  - Vary piston speed and bubble density
  - Measure RMSE for predicting ϕ values
- Observations from results:
  - Heatmap(left) indicates that **variance** is a significant source of error
  - Graph shows how training data affects performance via **learning curves**
  - Heatmap(right) suggests generalization performance may improve by training on more simulations ($\mu: 0.08 \rightarrow 0.05$ and $\sigma: 0.02 \rightarrow 0.01$)



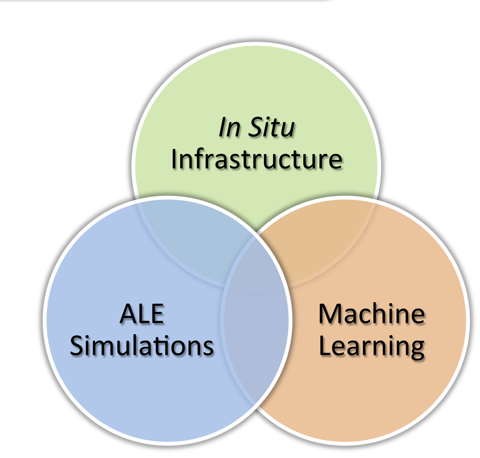


Conclusion and Future Work

- **Key takeaways**:
  - Mesh quality metrics provide early indicators of failure, which are captured by our ML approach (*see time history graph*).
  - Individual simulations differ. Training across multiple simulations improves generalization performance (*see heatmaps*).
  - Our ML approach automatically performs successful ALE on KULL Bubble Shock simulation without human intervention (*see paper*).

We developed a **first-of-a-kind** supervised learning framework for predicting conditions leading to ALE simulation failures.

- **Future research directions**:
  - How much failure time history to consider? Which transition function to use? Can we reduce noise in the negative labels?
  - In terms of physical accuracy, how do we compare simulations produced from different ALE strategies?
  - What is an effective infrastructure for integrating machine learning with ALE simulations?



This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344 (LLNL-POST-714918), with funding provided by LDRD 16-ERD-036.