Bridging Scales for Materials Modeling using Machine Learning

Using ML to rapidly provide upscaled crack statistics for more accurate damage modeling

Gowri Srinivasan
April 28, 2021
Motivation

- Simulating individual cracks has proven to be important to predict failure in brittle materials
- Simulating individual cracks, as Hybrid Optimization Software Suite (HOSS) (high-fidelity, HF) does, is expensive
- Recurrent Neural Networks (RNN) can predict time sequences reducing computation cost
- Continuum models like FLAG (low-fidelity, LF) do not simulate individual cracks
Our goal is to bridge between meso and continuum scales in an efficient ML-driven workflow.
Validation Data


**HOSS Data – 2D HF model**

*Evolution of the shock wave velocity*

- Number of HF simulations available = 100
- **Cost** = hundreds of processors hours
- **Variable** = initial crack location, length and orientation
Experimental Results

- Results extracted from experiments using a VISAR probe centered on the rear of the target.
- The figure shows the temporal evolution of the shock wave profile.
- Velocity roughly constant between 0.5μs to 2.5μs.
- Oscillations are due to the reflection of the shock wave.

Temporal evolution of shock wave profiles
• In black results using FLAG LF model

• PTW (Preston-Tonks-Wallace) model specifies stress as a function of strain, strain rate, and temperature

• PTW does not model cracks

• FLAG model needs to be improved to validated

Temporal evolution of shock wave profiles
Quantities of Interest for training ML

Compliance Tensor Correction

- Number of cracks
- Length of each crack

Maximum tensile stress as a function of time

Temporal evolution of number of cracks as a function of crack length

At each time step!
A Closer Look at the Crack Density Evolution

Initial distribution does not change substantially as a function of time.
Updated Damage for Training ML

Our approach: Use the initial distribution for all times evolving only the longest crack

- Crack counts
- Crack length (mm)
- Initial crack length distribution
- Length of the longest crack as a function of time (100 simulations)
Results using HF HOSS

- HOSS performs similar to the experiments

- FLAG with corrected compliance evolving all cracks compare with HOSS performance

- FLAG with corrected compliance with only the longest crack shows also a good performance
One-to-Many Machine Learning Model

Training data shape = # simulations (N), # timesteps, # features = N x 480 x 2

Input Layer
Input = Nx1x2
Output = Nx1x2

LSTM Layer
Input = Nx1x2
Output = Nx10

Repeat Vector Layer
Input = Nx10
Output = Nx100x10

LSTM Layer
Input = Nx100x10
Output = Nx100x10

Time Distributed (Dense(100)) Layer
Input = Nx100x10
Output = Nx100x100

Time Distributed (Dense(2)) Layer
Input = Nx100x100
Output = 100x2

# Training data = 70 → 70,000 (augmented)
# Test data = 30

Model input = Nx1x2
Model output = Nx100x2
Length of the Longest Crack Predictions

**Training Data**

![Graph showing the length of the longest crack as a function of time.]

**Error**

![Graph showing the mean range normalized RMSE error.]

*Length of the longest crack as a function of time.*

*Mean range normalized RMSE using 30 test data points.*
Maximum Stress Predictions

Training Data

Maximum tensile stress as a function of time

Error

Mean range normalized RMSE using 30 test data points.
Results using HOSS vs. FLAG + ML

- HOSS and FLAG +ML have a similar performance
- A single HOSS simulation costs 160 processors hour
- Once trained ML takes seconds to predict
- Computational cost reduction by four orders of magnitude

Temporal evolution of shock wave profiles
Summary, Conclusions and Future Work

- Reduced dimensionality of the problem
- Developed a fast ML simulator of HOSS (HF)
- Incorporated ML into the FLAG (LF) workflow
- FLAG + ML give us the accuracy of HOSS but 10,000 times faster
- The current model was tested only for this single experimental condition. Our next step is to consider varying loading conditions


