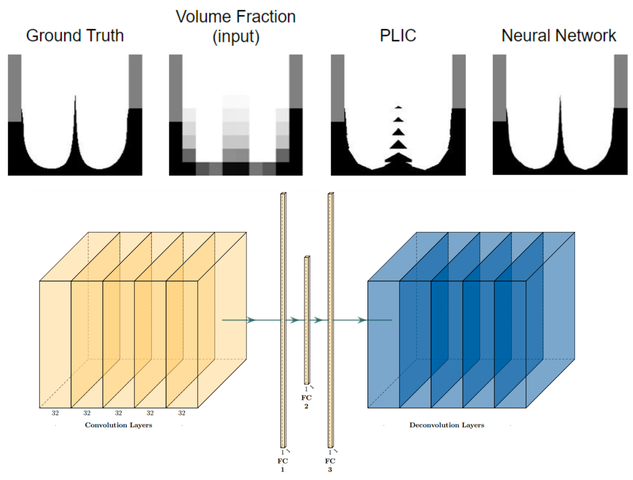
Optimizing Deep Learning Material Interface Reconstruction

Valen Yamamoto

Advisors: Ian Karlin, Dan Fenn

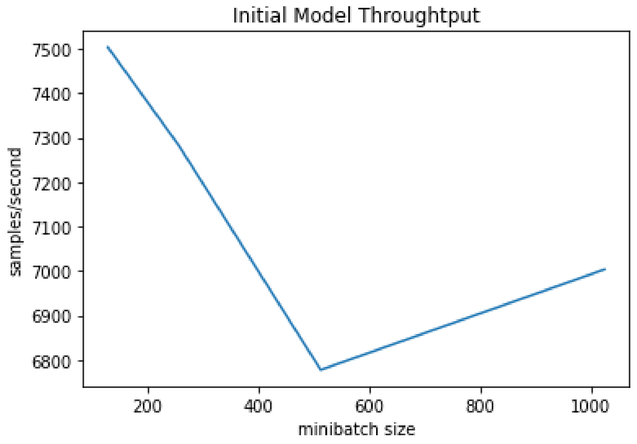
Problem Overview

- Material Interface Reconstruction is the process of constructing boundaries between two or more immiscible materials from a discrete mesh of volume fractions.
- Current methods require trade-offs between conservation of material and the continuity of the border; PLIC conserves all the material in a zone at the cost the continuity of the border.
- A neural network provides more accurate reconstructions
- A throughput of 100,000 samples/s (per MPI rank) required for in-the-loop inference for physics simulations
- Initial model is a convolutional autoencoder with 5 convolution/deconvolution layers, each with 32 convolution channels



Initial Results

- Throughput is 7500 samples/second.
- Bottleneck is the 132k size fully connected layers; 50% of computation time spent these layers
- Fully connected layers need to be reduced in size in order to get better performance.



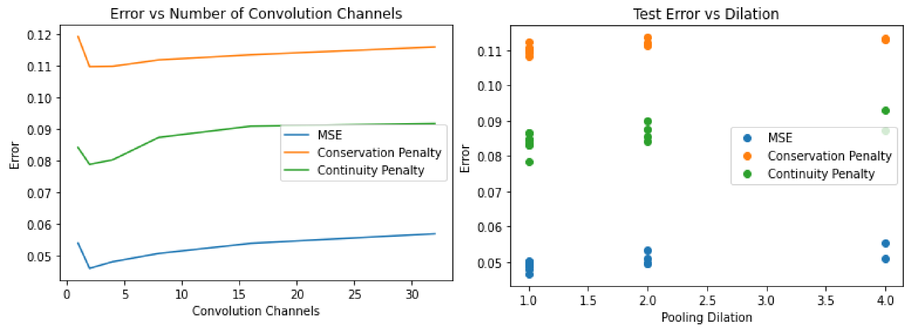
This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under contract DE-AC52-07NA27344. Lawrence Livermore National Security, LLC. LLNL-POST-825968

Abstract

Current material interface reconstruction methods provide inaccurate reconstructions; a neural network provides more accurate reconstructions. The initial model architecture was too large to provide the required throughput. Reducing the size of the model shows a smaller model could be used and provide similar accuracy. The throughput of the reduced model reaches the target throughput and increases throughput by 15x on NVIDIA A100 GPUs. Further work is being done porting the full model to SambaNova systems as well as additional optimizations on NVIDIA GPUs.

Finding a Reduced Model

- Number of convolution channels was decreased; pooling was introduced.
- Trained with mean squared error (MSE) and extra penalties for continuity of the border and conservation of the material.



Model Parameters	FC Layer Size	MSE	Conservation Error	Continuity Error
32 Channels (Original Model)	131072	0.0570	0.1160	0.0918
2 Channels	8192	0.0513	0.1098	0.0789
4 Channels	16384	0.0559	0.1099	0.0804
Pooling (Dilation 1 2 Channels)	4604	0.0466	0.1082	0.0783
Pooling (Dilation 2, 2 Channels)	2048	0.0535	0.1120	0.0877

	Mean	Standard Deviation
Non Pooling Model	4.9548%	0.4145%
Pooling Model	4.77%	0.1129%

- Conducting a 2-sample t-test of two best models gives a t-score of -2.0399 and a p-value of 0.02166; the pooling model provides better accuracy.

Two most accurate models:
- 2 convolution Channels, pooling with dilation of 1
- 2 convolution Channels

The pooled model has a fully connected layer size of 4608 while also providing better accuracy.

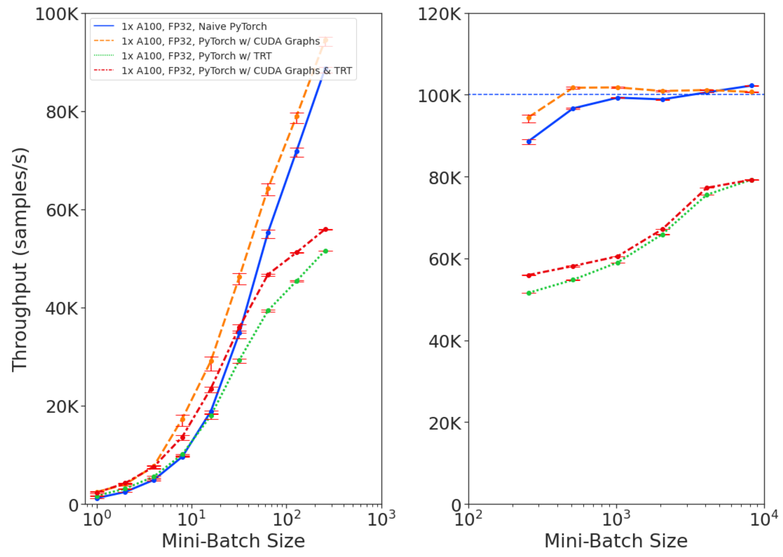
Next Steps

- Run full model remote inference on hardware accelerators: SambaNova and Cerebras
- Run TensorRT through ONNX, which has better support to get better throughput

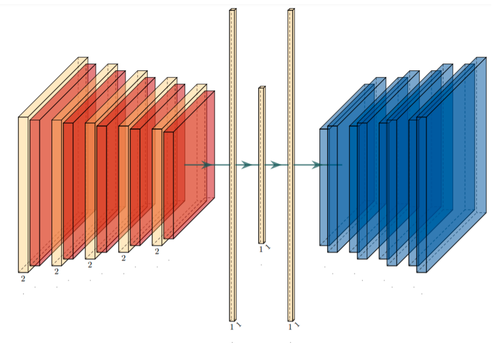
Final Results

Running on A100 GPUs, model achieves goal throughput using FP32

- Different optimizations have advantages at different mini-batch sizes



With 2 convolution, bottleneck is moved from the fully connected layers to the convolution layers where 70% of computation time is spent



Running a version of the model without layernorm on both Nvidia A100 GPUs and SambaNova Systems' SN10 at half precision, the model achieves goal throughput in both cases

- SambaNova hits throughput target earlier than Nvidia