In Situ ML for HPC Simulations

Riccardo Balin, Filippo Simini, and Ramesh Balakrishnan, Argonne National Laboratory

Kenneth E. Jansen, John A. Evans, and Alireza Doostan, University of Colorado Boulder

Data-Intensive Computing and AI/ML Applications at Scale

08/17/2023

In Situ ML for HPC Simulations

Simulation-based design of complex aerodynamic systems needs:

- Accurate physics models (closure/surrogate modeling)
- Understanding of key physical phenomena (visualization)
- Generation of large datasets (O(10^9-10^{10}) grid points)

Developed in situ ML framework (CFDML) that offers:

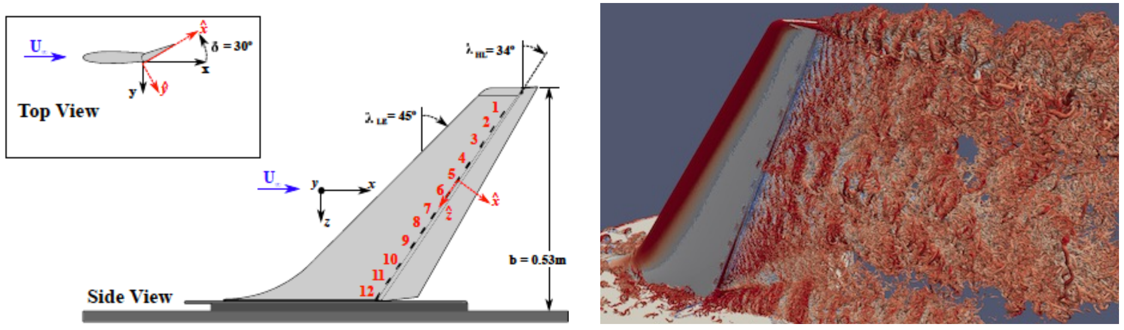
- Inference and training with SmartSim/SmartRedis libraries
- Data streaming to database and avoiding file system IO/storage
- Scalability and negligible overhead on simulation and training
- Efficient use of CPU and GPU resources
- Selection from multiple models at runtime – ANN for closure modeling and autoencoder for compression
- Distributed data parallel training with Horovod and DDP
- No dependency on CDF code (PHASTA, libCEED and NekRS)
- Portability (installed on Polaris and Sunspot)

Data transfer overhead on simulation during in situ training

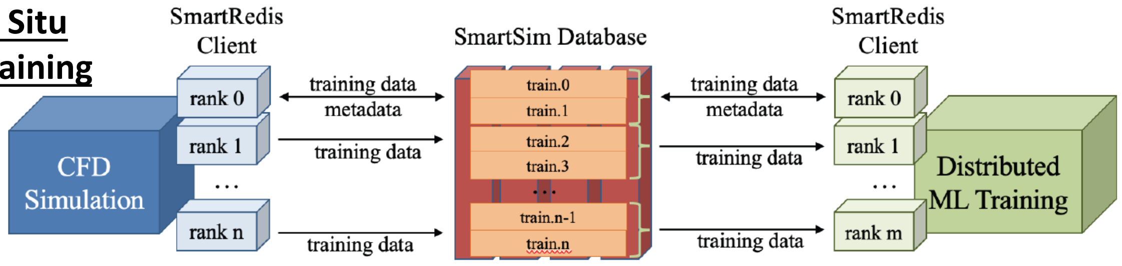
Solver Component	Average [sec]	Standard Deviation [sec]	
Equation formation	45.426	0.678	solver time
Equation solution	453.386	0.698	solver time
Client initialization	0.002	0.001	
Metadata transfer	0.065	0.005	
Training data send	0.120	0.021	<< 1% of solver time

Balin et al., "In Situ Framework for Coupling Simulation and Machine Learning with Application to CFD". arXiv:2306.12900, 2023. https://github.com/rickybalin/CFDML.

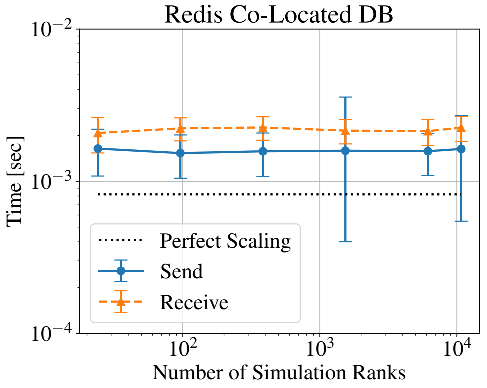
Simulation of vertical tail with synthetic jet actuation



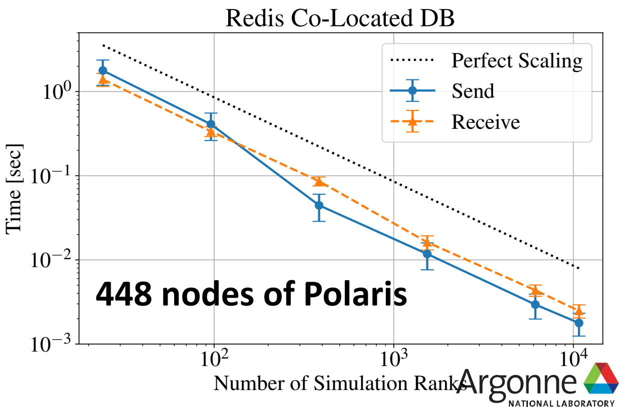
In Situ Training



Weak Scaling on Polaris



Strong Scaling on Polaris



Argonne NATIONAL LABORATORY

In Situ ML for HPC Simulations

Current challenges

- Model architecture and hyperparameters must be tuned offline
- Continual learning from sequentially generated data
- Database may result in performance bottleneck, depending on inferencing needs
  - May require more invasive tightly-coupled approach with ML inference libraries (OpenVINO, LibTorch, ONNX)

Future work and longer term goals

- Integration with tools for scalable model discovery and hyperparameter optimization (e.g., DeepHyper, HYPPO)
- Smarter on-the-fly selection of useful simulation snapshots for training
  - Use UQ and accuracy metrics
  - Conscious of system memory limitations
- Move towards ML-training informed data generation (e.g., AMR and launching supplemental simulations)

Collaboration opportunities

- Framework is not limited to CFD and easily extendable to any computational science
- Always looking for new applications to drive development of in situ ML capabilities