Coupling Simulations with AI

Riccardo Balin and Bethany Lusch
With help from Christine Simpson, Harikrishna Tummalapalli
Argonne National Laboratory, LCF

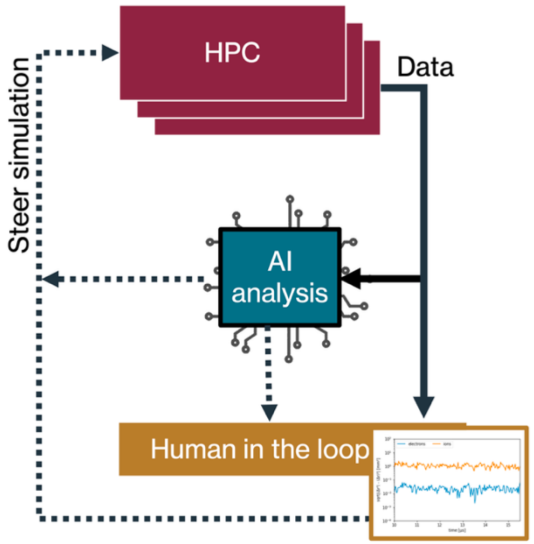
AI4Science Series:
Advanced Topics in AI for Science
October 28, 2025

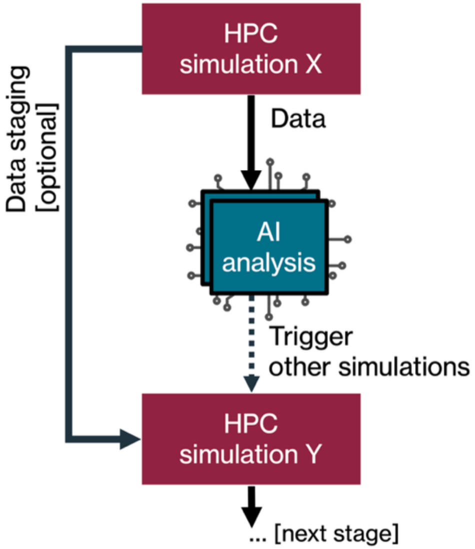
Why Couple HPC Simulations with AI/ML?

- Substitute inaccurate or expensive components of simulation with ML models
  - E.g.: Closure or surrogate modeling
- Optimize simulation parameters on-the-fly
  - E.g.: Select solver parameters at runtime based on AI inference
- Avoid IO bottleneck and disk storage issues during offline training
  - E.g.: In situ/online training through data streaming or in-memory staging
- Active learning and model fine-tuning
  - E.g.: Continuous fine-tuning and deployment of model
  - E.g.: Access training data not available during offline pre-training
- Steering of simulation ensembles
  - E.g.: Design space exploration or parameter optimization guided by AI

How to Couple HPC Simulations and AI/ML?

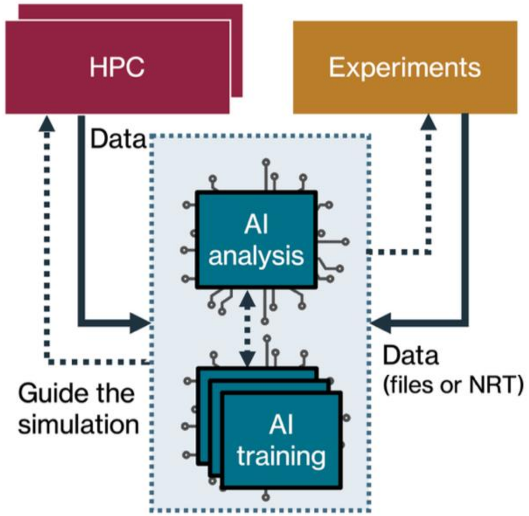
Steering of Ensembles

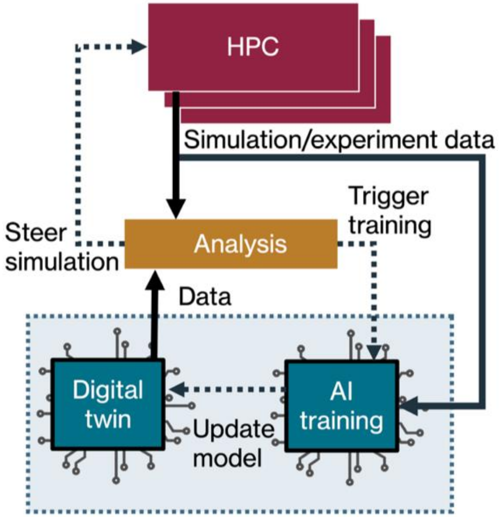




Optimize Simulation Parameters

Active Learning/ Online Fine-Tuning





Digital Twin

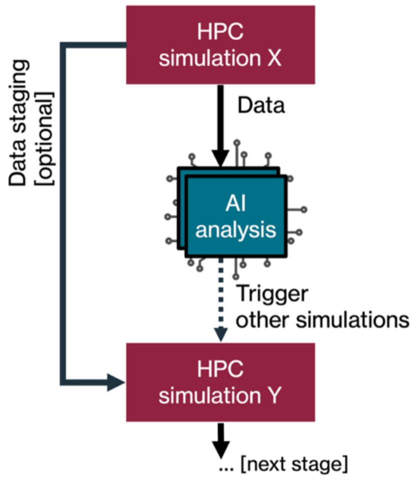
Brewer et al. "AI-coupled HPC workflow applications, middleware and performance." *arXiv:2406.14315* (2024)

How to Couple HPC Simulations and AI/ML?

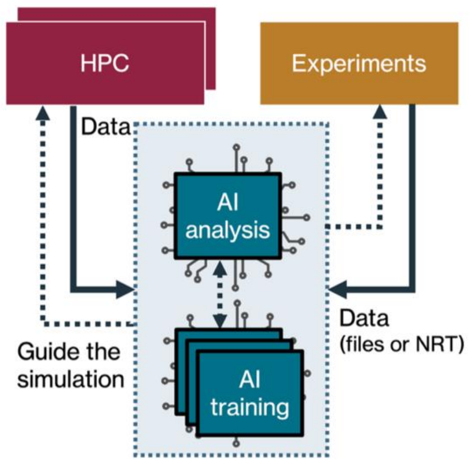
Type of Coupling

- ❑ ML inference only
- ❑ ML training only
- ❑ Both training and inference

Inference Only



Training and Inference



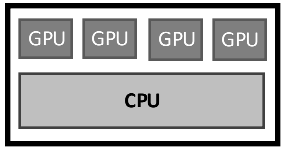
Programming Languages and Programming Models

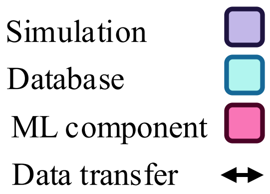
- ❑ Simulation and ML components often written in different languages
- ❑ AI/ML relies heavily on vendor libraries (stuck with vendor language or programming model)
- ❑ Separate or shared parallelism strategies
  - ➢ Shared vs. separate MPI communicators
  - ➢ Domain decomposition vs. batch or model parallelism

Brewer et al. "AI-coupled HPC workflow applications, middleware and performance." *arXiv:2406.14315* (2024)

How to Couple HPC Simulations and AI/ML?

Heterogeneous HPC node

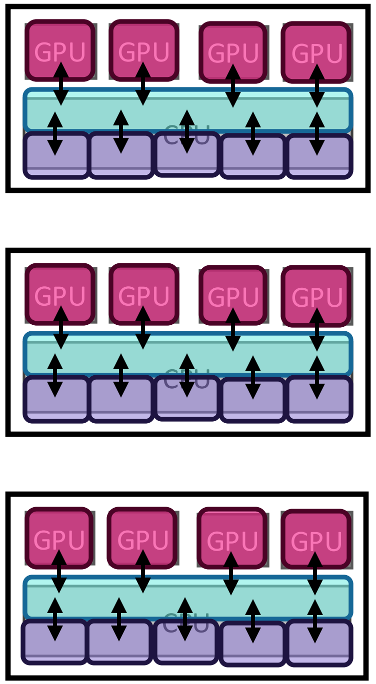




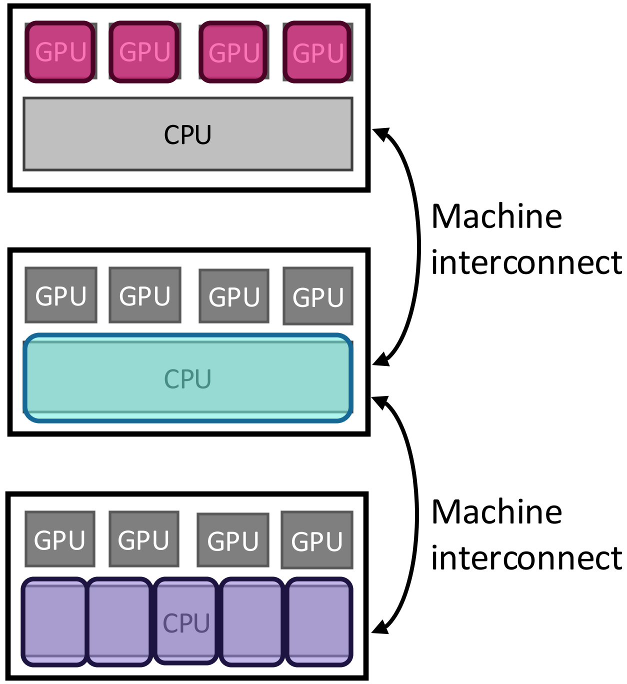
Physical Proximity

- ❑ Colocation: components share the same nodes
- ❑ Node-level clustering: components use different nodes on the same system
- ❑ Multi-system: components are run on separate specialized systems

Colocated Deployment



Node Clustered Deployment



System Clustered Deployment

How to Couple HPC Simulations and AI/ML?

Data Access

- ❑ Coupling simulation and ML requires frequent data sharing/transfer between components
- ❑ Direct: components share same memory space (may allow for zero-copy data transfer)
- ❑ Indirect: components use distinct logical memory (requires data copy and may require data transfer)

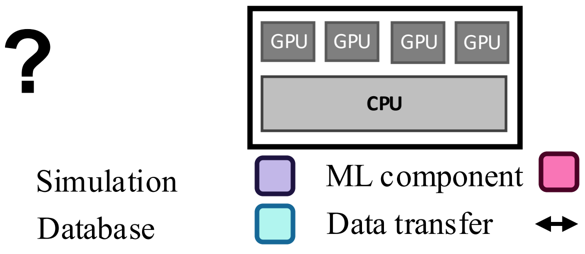
Data Staging or Streaming

- ❑ Staging: data is staged in memory or on disk (can reduce idle time but increases number of transfers)
- ❑ Streaming: data is streamed directly between components (can increase idle time due to synchronization)

Childs et al., "A terminology for in situ visualization and analysis systems", Intl. Journal of High Performance Computing Applications, 2020

How to Couple HPC Simulations and AI/ML?

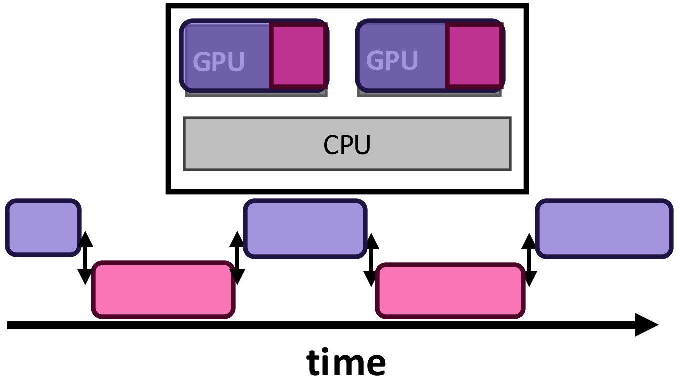
Heterogeneous HPC node



Execution Management

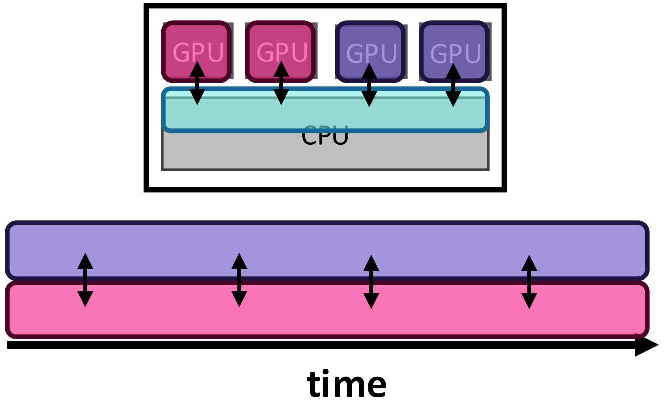
- ❑ Time division (tight coupling)
  - ➢ Components run on same compute resources (may even use same processes)
  - ➢ Staggered in time, execution of one component halts the other
  - ➢ May allow for direct memory access and no data copy/transfer
  - ➢ Idle time of individual components may be significant

Time Division: Same Compute Hardware



- ❑ Space division (loose coupling)
  - ➢ Components run on separate compute resources
  - ➢ Concurrent in time, components run simultaneously
  - ➢ Minimal idle time of components for fast data copy/transfer
  - ➢ Usually requires indirect memory access with data copy/transfer

Space Division: Separate Compute Hardware



Childs et al., "A terminology for in situ visualization and analysis systems", Intl. Journal of High Performance Computing Applications, 2020

How to Couple HPC Simulations and AI/ML?

- Coupling simulation and AI/ML can be a complex space to navigate
- Implementation choices can vary significantly depending on ML task and application needs
- This session will cover some common tools and approaches supported at ALCF

Hands On Part 1: Molecular Design with Parsl

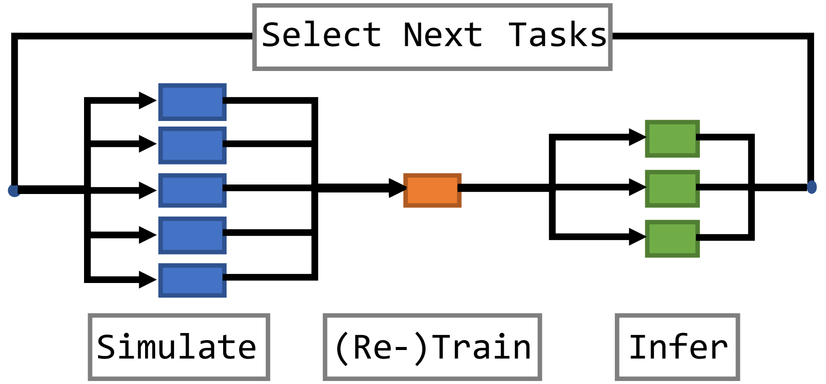
Science Problem: identify high value molecules (i.e. molecules with high ionization energy) among a search space of billions of candidates

Challenge: The simulation is too computationally expensive to run for every candidate molecule

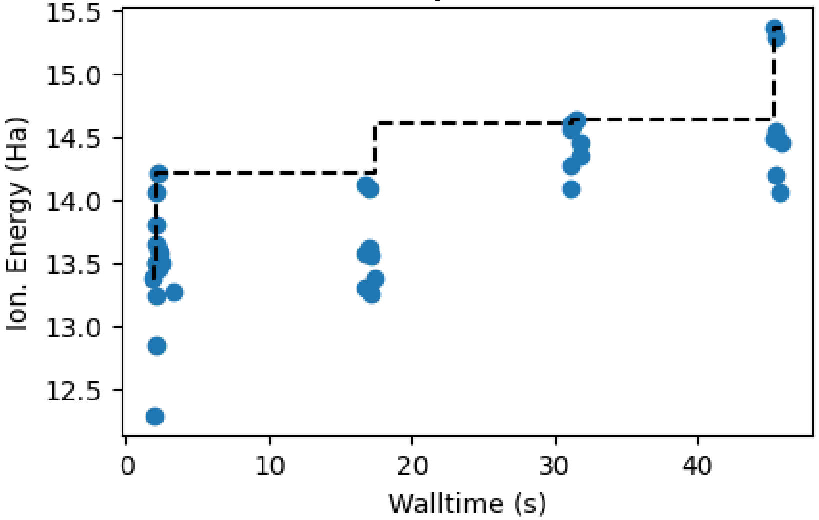
Approach: Create an active learning loop that couples simulation with machine learning to simulate only high value candidates

Tools:
- **Parsl** is used for task launching and integration
- Use RDkit and scikit-learn to train a k-nearest neighbor (knn) model
- Simulations done with MD package xTB

Workflow Pattern: AI/ML Components steer Simulations



Simulation Results produced over time



Observations from Hands On…

Hands On Part 2: Parsl vs. DragonHPC

- DragonHPC is a composable distributed run-time for managing processes, memory, and data at scale through high-performance communication objects
- Open-source project developed by HPE
- One key feature: Dragon Dictionaries
  - Dragon Dictionaries take in data from client processes in the form of key-value pairs
  - Data are sharded across nodes through channels by Memory Pool managers that sit on each node
  - Dragon Dictionary Managers dynamically load balance key-value pairs across managers
  - Transfers are done with RDMA (slingshot networks) or TCP (non-slingshot networks)

Software for Coupling Simulations and AI/ML

- Tight coupling
  - —Python and ML frameworks embedding into simulation code
    - PythonFOAM, TensorFlowFOAM, HONEE (by Romit Maulik, Saumil Patel, Bethany Lusch at ALCF)
  - —Linking to LibTorch or ONNX Runtime libraries for ML inferencing from C, C++ and Fortran
    - Aurora will support LibTorch and Intel's OpenVINO inference library
- External coupling
  - —Parsl
    - Workflow tool for distributed, parallel task execution
  - —SmartSim / SmartRedis
    - Workflow manager and client libraries for in-situ workflows by sharing data across a database
  - —ADIOS2
    - Same I/O API to transport data across different media (file, wide-area-network, in-memory staging, etc.), favoring asynchronous streaming
  - —Dragon
    - Run-time library for managing dynamic processes, memory, and data at scale through high-performance communication

Summary Performance Considerations

In-depth webinar available:

“Methods, Tools, and Best Practices for Coupling Simulation and AI on ALCF Systems”

This research used resources of the Argonne Leadership Computing Facility (ALCF), which is a DOE Office of Science User Facility supported under Contract DE-AC02-06CH11357.