Asynchronous and Adaptive Execution of AI-driven HPC Workflows

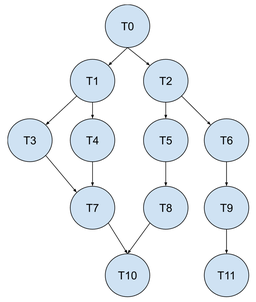
Ozgur Ozan KILIC (BNL)
Matteo TURILLI (BNL/Rutgers), Shantenu JHA (BNL/Rutgers)

Motivation

Scientific workflows are undergoing a transformation toward greater heterogeneity due to the increased integration of machine learning and the need for scaling on high-performance computing infrastructures. In this evolving landscape, asynchronicity and adaptivity have emerged as pivotal factors to increase the execution efficiency of workflows at unprecedented scale and, thus, accelerating scientific discoveries. Specifically, enabling asynchronous and adaptive execution of heterogeneous tasks within scientific workflows increases resource utilization, reduces makespan and minimizes costs.

We will devise execution models for AI/ML-coupled scientific workflows and use those models to study the correlation between degrees of asynchronicity and resource utilization of heterogeneous workflows. Further, we will design novel adaptive strategies to further execution makespan and cost reductions. Finally, we will implement those models and strategies into middleware to support the execution of a variety of real-world workflows on DOE HPC platforms.

Effect of Asynchronicity



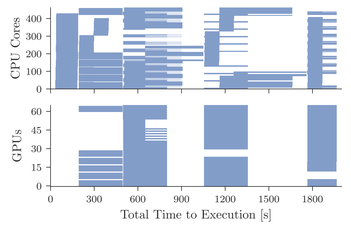
Arbitrary Workflow

Task	Number of Tasks	CPU	GPU	Run Time (sec)
T0	430	4	0	150
T1	30	30	1	300
T2	420	2	0	100
T3	300	3	0	400
T4	60	6	1	150
T5	30	10	0	150
T6	30	7	1	300
T7	60	5	1	300
T8	490	2	0	100
T9	230	2	0	500
T10	280	4	0	100
T11	60	10	1	200

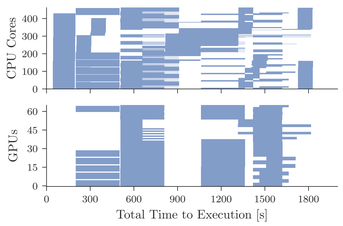
Table of tasks

Three levels of asynchronicity of the arbitrary workflow

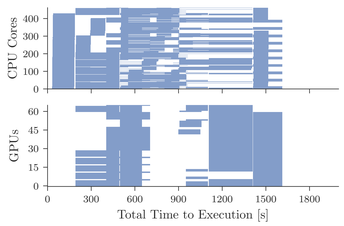
- Synchronous: Tasks on a higher level of DAG cannot start without every task on the previous level end. e.g., T5 waits for T2 and T1 (~2000 sec)
- Workload-Level Asynchronous: Tasks in different non-merging branches of DAG can be executed asynchronously. e.g., T8 waits for T3-5 but T9 only waits for T6 to execute (1800s)
- Task-Level Asynchronous: Every task can execute asynchronously as long as it satisfies its own dependencies. e.g., T8 can start as soon as T5 ends (1700s)



Synchronous Execution



Workload Asynchronous Execution

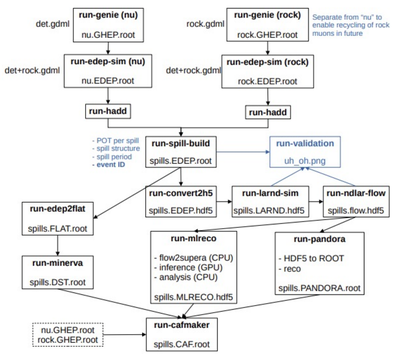


Task Asynchronous Execution

HEP Workflows

Challenges to run HEP workflows on HPC platforms at scale:

- HEP Workflows are non-uniform with data and compute requirements
- While most HEP computational tasks are CPU-based, there is an increased use of accelerators.
- HEP Workflows require large data movement and storage
- HEP workflow consists of Heterogenous and Hybrid tasks

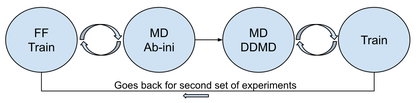


DUNE ND 2x2 Workflow

Requirements:

- Asynchronous execution on heterogenous hardware (GPU+CPU) to maximize the resource utilization and minimize cost
- Adaptive execution of heterogenous tasks to adjust resources across different types of tasks at runtime

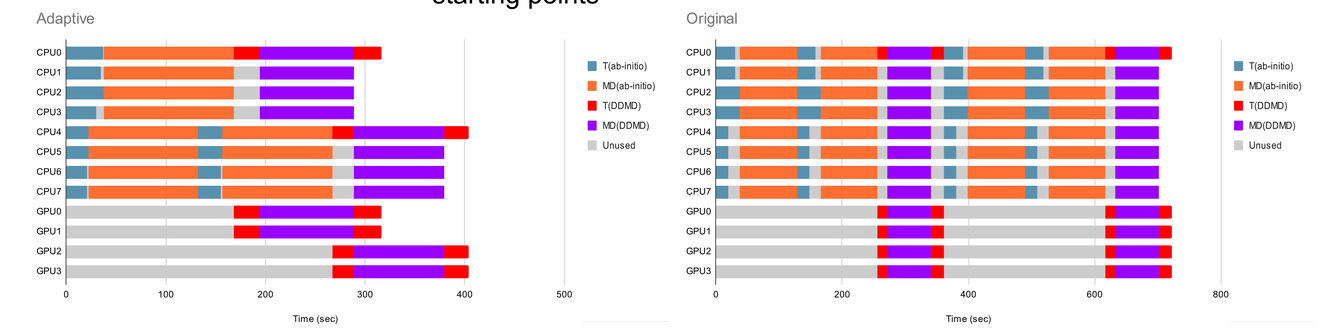
Effects of Adaptivity



Simplified NWChem Workflow mini-app

Two designs for the adaptive execution of the NWChem workflow :

1. Start a pretraining for MD-DDMD simulation after partially satisfying uncertainty quantification, reducing the total training time by building a surrogate model
2. Start a full set of MD-DDMD after partially satisfying uncertainty quantification, doubling the results of the workflow similar to running twice with different starting points



Expected results for the second design.

This project is supported by US Department of Energy, Office of Science, Office of High Energy Physics under HEP-CCE.

www.bnl.gov