

Advancing HPC Workload-Management Simulation

As High-Performance Computing (HPC) systems consume substantial energy, they require sophisticated workload-management strategies that balance performance with environmental impact. To develop and evaluate these strategies, researchers often rely on trace-driven simulation, since experimentation on production HPC systems is typically impractical due to limited access, operational constraints, and the difficulty of running controlled, reproducible studies.

Although several HPC workload-management simulators exist, many suffer from architectural limitations that hinder usability and extensibility. They are often monolithic, tightly coupling the simulation engine, workload-management policies, and performance-monitoring capabilities. This tight coupling makes the codebase difficult to understand and modify. For example, adding a new workload-management algorithm typically requires changes to core simulation code, creating a steep learning curve for users and discouraging collaborative development. As a result, researchers tend to create separate versions of the codebase to implement custom extensions, leading to different implementations and reproducibility challenges. The tight coupling also makes it difficult to integrate emerging HPC capabilities such as dynamic resource reconfiguration, multi-objective optimization, and carbon-aware policies. Insufficient documentation intensifies these issues, limiting both the use of existing simulators and the impact of simulation-based research.

This project aims to advance HPC workload-management simulation by developing an easy-to-use and extensible simulator that supports systematic study of energy-efficient strategies. We will investigate how design principles — separating stable simulation infrastructure from evolving workload-management policies through well-defined interfaces — can shift simulators from single-purpose prototypes to general-purpose research platforms. By establishing clear extension points and modular component boundaries, the project will enable researchers to combine independently developed workload management algorithms, predictive models, and performance-metric collectors without invasive changes to the core codebase.

Research Activities

- Analysis of existing HPC workload-management simulators to identify opportunities to improve usability and extensibility.
- Investigation of architectures that enable clean separation between simulation infrastructure (e.g., event scheduling, time management, state tracking) and domain logic (e.g., such as workload-management strategies, performance metrics collection).
- Development of a reusable and extensible framework that supports dynamic composition of workload parsers, workload-management algorithms, predictive models, and performance metrics collectors through declarative configuration files.
- Validation of the framework using production HPC workload traces through correctness testing, performance benchmarking, and comparative evaluation against baseline implementations.
- Production of a comprehensive documentation, including architectural rationale, user and developer guides, tutorials, experimental methodology, and contribution guidelines to support community adoption and collaborative development.

Evoluzione della simulazione della gestione dei carichi di lavoro per sistemi di HPC

I sistemi di High-Performance Computing (HPC) consumano notevoli quantità di energia e richiedono strategie sofisticate di gestione dei carichi di lavoro che bilancino le prestazioni con l'impatto ambientale. Per sviluppare e valutare tali strategie, i ricercatori si affidano spesso alla simulazione *trace-driven*, poiché la sperimentazione su sistemi HPC di produzione è generalmente impraticabile a causa dell'accesso limitato, dei vincoli operativi e della difficoltà di condurre studi controllati e riproducibili.

Sebbene esistano diversi simulatori per la gestione dei carichi di lavoro HPC, molti soffrono di limitazioni architetturali che ne ostacolano l'usabilità e l'estensibilità. Spesso adottano architetture monolitiche che accoppiano strettamente il motore di simulazione, le politiche di gestione del carico di lavoro e le funzionalità di monitoraggio delle prestazioni. Questo forte accoppiamento rende il codice difficile da comprendere e modificare. Ad esempio, l'aggiunta di un nuovo algoritmo di gestione del carico di lavoro richiede tipicamente modifiche al codice principale del simulatore, creando una curva di apprendimento ripida per gli utenti e scoraggiando lo sviluppo collaborativo. Di conseguenza, i ricercatori tendono a creare versioni separate del codice per implementare estensioni personalizzate, portando a implementazioni divergenti e a problemi di riproducibilità. Questo forte accoppiamento del codice rende inoltre difficile integrare funzionalità emergenti in ambito HPC, come la riconfigurazione dinamica delle risorse, l'ottimizzazione multi-obiettivo e politiche *carbon-aware*. Una documentazione insufficiente intensifica ulteriormente questi problemi, limitando sia l'utilizzo dei simulatori esistenti sia l'impatto della ricerca basata sulla simulazione.

Questo progetto mira ad avanzare la simulazione nell'ambito della gestione dei carichi di lavoro di supercomputer sviluppando un simulatore facile da usare ed estensibile, che supporti lo studio sistematico di strategie energeticamente efficienti. Verrà analizzato come i principi di progettazione — quali la separazione tra infrastruttura di simulazione stabile e politiche di gestione del carico di lavoro in evoluzione tramite interfacce ben definite — possano trasformare i simulatori da prototipi monouso in piattaforme di ricerca di uso generale. Stabilendo chiari punti di estensione e confini modulari tra i componenti, il progetto consentirà ai ricercatori di combinare algoritmi di gestione del carico di lavoro, modelli predittivi e collettori di metriche di prestazione sviluppati in modo indipendente, senza apportare modifiche invasive al codice principale.

Attività di ricerca

- Analisi dei simulatori per la gestione dei carichi di lavoro HPC già esistenti al fine di identificare opportunità per migliorarne l'usabilità e l'estensibilità.
- Studio di architetture che consentano una separazione chiara tra infrastruttura di simulazione (ad esempio, pianificazione degli eventi, gestione del tempo, tracciamento dello stato) e logica di dominio (come le strategie di gestione del carico di lavoro e la raccolta delle metriche di prestazione).
- Sviluppo di un framework riutilizzabile ed estensibile che supporti la composizione dinamica di parser dei carichi di lavoro, algoritmi di gestione del carico di lavoro, modelli predittivi e collettori di metriche di prestazione tramite file di configurazione dichiarativi.

- Validazione del framework utilizzando carichi di lavoro reali mediante test di correttezza, benchmarking delle prestazioni e valutazione comparativa rispetto a implementazioni di riferimento.
- Produzione di una documentazione completa, comprensiva di motivazioni architettoniche, guide per utenti e sviluppatori, tutorial, metodologia sperimentale e linee guida per contribuire allo sviluppo del simulatore, al fine di favorire l'adozione da parte della comunità e lo sviluppo collaborativo.