HPC Project Proposal

Sowmya Yellapragada

March 2025

1 Description

I'm planning to work on a high-performance numerical solver for Partial Differential Equations in C++, focusing on parallelism and AI-driven optimization. The goal is to develop an efficient, CPU-parallelized solver using parallel programming frameworks while integrating machine learning to enhance solver performance, specifically for adaptive mesh refinement or parameter tuning.

2 Steps Involved

1. **Research & Setup -** Choose a PDE (likely the Heat or Wave Equation), study numerical methods like the Finite Difference Method (FDM) or Finite Element Method (FEM), and set up a structured C++ project with CMake.
2. **Serial Solver Implementation -** Develop a working single-threaded solver and validate its correctness.
3. **Parallelization -** Introduce PThreads for multithreading, OpenMP for shared-memory parallelism and MPI for distributed-memory (and other methods, that I might find interesting, say, CUDA if I can get hold of the resources), and optimize memory access patterns.
4. **Performance Benchmarking -** Use profiling tools to measure efficiency, analyze bottlenecks, and refine performance.
5. **AI Integration -** Train an ML model to predict optimal solver parameters or guide adaptive mesh refinement.
6. **Visualization & Documentation** – Create visual outputs, or maybe, a dynamic dashboard, document findings, and publish everything on GitHub.

3 Expected Results

A fast, scalable PDE solver that runs efficiently on CPUs/GPUs, with AI-enhanced adaptivity to improve accuracy and performance. This project could lead to further research opportunities and even a publication!