Computação em Larga Escala

Summary

Eurico Pedrosa

António Rui Borges

Universidade de Aveiro - DETI

2025-05-25

Large-Scale Computing

What is High-Performance Computing (HPC)

- Use of powerful resources to solve complex problems
  - Multicore CPUs, GPUs, clusters
  - Goals: high throughput, efficiency
- Importance in fields like AI, weather, and physics
- Emerging paradigms: heterogeneous computing

Parallelism Models and Architectures

Classification by Granularity

- Coarse-grain: tasks run independently (e.g., MPI)
- Medium-grain: shared memory threads (e.g., std::thread)
- Fine-grain: per-instruction (e.g., CUDA)

Architectures

- Shared vs. distributed memory
- Interconnection topologies: mesh, torus, tree, fat-tree

Concurrency and Synchronization

Types of Process Interactions

- Independent vs. Cooperating processes
- Mutual exclusion and critical regions
- Deadlock, livelock, starvation

Resource Management and Deadlock Conditions

- Preemptable vs. Non-preemptable resources
- Four deadlock conditions: mutual exclusion, hold and wait, no preemption, circular wait

Thread-Level Parallelism

Introduction to Thread Pools

- Pool of worker threads processing queued tasks
- Advantages: efficiency, reduced overhead, scalable
- Used in web servers, big data, and simulations

Message-Passing Programming

Concepts of Message Passing

- Communication without shared memory
- Blocking vs. Non-blocking synchronization
- Direct and indirect addressing (mailboxes, ports)

Communication Patterns

- One-to-one, broadcast, multicast
- Scatter and gather
- Producer-consumer models with mailboxes

MPI: Message Passing Interface

Basics and Setup

- Standard API for inter-process communication
- `MPI_Init`, `MPI_Finalize`, `MPI_Comm_rank`, `MPI_Comm_size`
- Compilation and execution using `mpic++`, `mpiexec`

Error Handling and Communicators

- Error handlers: `MPI_ERRORS_ARE_FATAL`, `MPI_ERRORS_RETURN`
- Communicators define communication contexts (e.g., `MPI_COMM_WORLD`)

Collective Communication

- Broadcast, Scatter, Gather, and their signatures
- Blocking nature and use cases

Non-Blocking Communication

- `MPI_Isend`, `MPI_Irecv`, `MPI_Wait`, `MPI_Test`
- Overlap computation and communication
- Use cases and performance considerations

SLURM Workload Manager

What is SLURM?

- Open-source job scheduler for Linux clusters
- Used in top supercomputers (El Capitan, Frontier)

Architecture and Usage

- Components: `slurmctld`, `slurmd`, `slurmdbd`, `slurmrestd`
- Job submission: `sbatch`, `srun`
- Configuration: `slurm.conf`, partitions, node definitions

CUDA Programming

- CUDA model: kernels, thread blocks, memory hierarchy
- `__global__` functions launched with `<<<grid, block>>>`
- Memory management: host vs. device memory

Execution and Scalability

- Thread/block/grid hierarchy
- Scalability across GPU architectures
- Compilation with `nvcc`