Lecture Summary

Concepts and Models of Parallel and Data-centric Programming

Lecture, Summer 2021

Dr. Christian Terboven <terboven@itc.rwth-aachen.de>

Outline

1. Foundations

Cluster Architecture

Convergence of HPC and Big Data

Overview of Parallel Programming Concepts and Models

Amdahl’s Law

2. Shared Memory

Processes and Threads

Threading in C++

RAII Idiom, Move Semantics

Example: Simple Queue

Mutual Exclusion

Race Condition / Data Race

Condition Variables

Futures

Implementation of a Lock

Peterson Algorithm

Test-and-set Locks

Remarks on implementations

Memory Consistency / Atomicity

Five Patterns of Synchronization

- Coarse-grained
- Fine-grained
- Optimistic
- Lazy
- Lock-free

Parallel STL

SIMD Programming

3. GPU Programming

Comparison CPU / GPU

Data-Parallel Computing

GPU Application Design Cycle

- **Assess:** Performance Models (Roofline)
- **Parallelize:** Concepts of GPGPU Programming / CUDA
- **Optimize:** Synchronization, Branching, Data Access Patterns
- **Deploy:** Compare outcome with expectations

Application Area: Machine Learning on GPUs

4. Bulk-Synchronous Parallelism

From Nodes to Clusters

Communication Primitives

BSP Computer

BSP Programming Model

- Computation / Communication
- Supersteps

BSP Cost Model

Example: Inner Product

Bulk Library: Data Distribution, Distributed Variables, Coarrays

5. Message Passing

Interface Overview

MPI Messages: Content + Envelope

Point-to-Point Communication: Send / Receive

Communicators

MPI Data Types

Collective Communication: Barriers, Reductions, …

6. Distributed Shared Memory

Distributed Shared Memory

PGAS Motivation

Comparing Shared Memory, Distributed Memory, and PGAS

DASH Overview

- Distributed Data Structures
- Data Distribution
- Algorithms
- Tasking

7. Parallel Algorithms

Berkeley DWARFS

Dense Linear Algebra

Sparse Linear Algebra

Monte Carlo Methods

Graph Traversal

8. Parallel I/O

I/O Hierarchy

File Systems (GPFS, Lustre, BeeGFS)

Parallel I/O Schemes

- Centralized, Task-Local, Shared File, Multi-Shared Files
- Data Distribution in Parallel Applications

False Sharing of Blocks

Parallel I/O Libraries

- POSIX-I/O, MPI-I/O, HDF5

9. MapReduce

MapReduce Programming Model

Hadoop Ecosystem

Hadoop Distributed File System (HDFS)

Yet Another Resource Negotiator (YARN)

Comparison to MPI / PGAS and SQL

MapReduce Design Patterns

- Summarization Patterns
- Filtering Patterns
- Data Organization Patterns

10. Apache Spark

Spark Programming Model

Resilient Distributed Datasets (RDDs)

Job Scheduling and Fault Tolerance

Streaming and Applications

Comparison to MapReduce

Comparison to Distributed Shared Memory

Exemplary Oral Exam Questions

Examples / 1

Foundations

What does Amdahl's Law formulate?

Shared Memory

Explain the semantics of a Future as realized by std::async.

Which Patterns of Synchronization do you know? Explain pattern X.

Accelerators

Explain the differences in the architecture of a GPU and a CPU.

Distributed Memory

Explain the concept of a Superstep in BSP.

Explain how you would implement …

Examples / 2

Distributed Shared Memory

What does PGAS stand for? Explain the concept.

Parallel I/O

Which schema of Parallel I/O do you know? Explain schema X.

MapReduce and Spark

How does MapReduce differ from MPI or OpenMP? Explain aspect X.

Explain NameNode and DataNode in in Apache Hadoop.

What does the given code do?

Questions ?