Parallel algorithms at ENS Lyon

Yves Robert
Ecole Normale Supérieure de Lyon
& Institut Universitaire de France

TCPP Workshop
February 2010

Outline

Scope

- Follow-on of classic CLRS-based algorithm
- Objective: Apprehend the complexity of parallel algorithms
- Focus is on models and algorithms
- Provides a sound basis for parallel programming
- Not a HPC course

Organization

- 16 weeks
- Each week = 2h class
  + 2h supervised exercises (or programming sessions)
- **MPI project**
- Midterm and final exam

Outline

Models (4 weeks)

- Sorting networks
  - Odd-even merge sort, 0-1 principle
  - Odd-even transposition sort
  - Odd-even sorting on a 1D network (work optimal)
- PRAM
  - Models (EREW, CREW, CRCW)
  - Pointer jumping (list ranking, prefix, Euler tour)
  - Performance evaluation
    $\Rightarrow$ Cost, work, speedup and efficiency, Brent's theorem
  - Comparison of PRAM models
    Model separation, simulation theorem
  - Sorting machine
    $\Rightarrow$ Merge, sorting trees, complexity and correctness
  - Relevance of the PRAM model

Networking (3 weeks)

- Interconnection networks
  ⇒ Static and dynamic topologies
- Communication models
  ⇒ Point-to-point communication protocols
- Case study: the unidirectional ring
  ⇒ Broadcast, scatter, all-to-all, pipelined broadcast
- Case study: the hypercube
  ⇒ Labeling vertices, paths and routing
  ⇒ Embedding rings and grids
  ⇒ Collective communications
- Peer-to-peer computing
  ⇒ Distributed hash tables and structured overlay networks
  ⇒ Chord, Plaxton's routing algorithm
  ⇒ Multi-casting in a distributed hash table

Algorithms on a processor ring (2 weeks)

- Matrix-vector multiplication
- Matrix-matrix multiplication
- First look at stencil applications
- LU factorization
  $\Rightarrow$ Basic version, pipelining on the ring, look-ahead algorithm
- Second look at stencil applications
  $\Rightarrow$ Granularity, overlap, mapping, dependencies
- Implementing logical topologies
- Distributed vs. centralized implementations
- Summary of algorithmic principles

Processor grids and load balancing (3 weeks)

- Logical 2-D grid topologies
- Matrix multiplication on processor grids
  - Outer-product algorithm
  - Grid vs. ring?
  - Three matrix multiplication algorithms
- 2-D block cyclic data distribution
- Load balancing for heterogeneous platforms
  - Load balancing for 1-D data distributions
    $\Rightarrow$ Static vs. incremental allocation algorithm
    $\Rightarrow$ Application to stencils and LU factorization
  - Load balancing for 2-D data distributions
  - Matrix multiplication on a heterogeneous grid
  - Hardness of the 2-D data partitioning problem

Scheduling and loop parallelization (4 weeks)

- Where do task graphs come from?
- Solving Pb(∞)
- Solving Pb(p)
  - NP-completeness of Pb(p), list schedules, Graham's bound and critical paths
  - Approximation algorithms for independent tasks
- Taking Communication Costs Into Account
  - NP-completeness of Pb(∞), guaranteed heuristics
  - List heuristics for Pb(p)
  - HEFT (extension to heterogeneous platforms)
- Scheduling at Compile-Time
  - Dependence levels and Kennedy-Allen algorithm
  - Dependence vectors and Lamport's hyperplane method
  - Uniform loop nests and unimodular space-time transformations