Communication Optimization for Parallel Processing

Lecture 1

Pangfeng Liu, Department of Computer Science and Information Engineering, National Taiwan University.

Importance of Parallel Processing

- The users demand solution for larger problems with faster turnaround time.
- Weather forecast, military simulation, computational physics.
- The Google search engine consists of thousand of computers.
- The deep blue and deep thought also used parallel processing techniques.

Parallel Computing

- Having multiple processors working *simultaneously* on a problem. The processor work collectively to solve the single problem. Usually for the maximum performance.
- Temporal concurrency.

Distributed Computing

- Having multiple systems working in different places, usually for reliability purposes, or when the data the system needs to process are scatter in different geographical regions.
- Spatial distribution.

Adding Numbers Together

- A simple experiment
- There are 200 numbers to be added.
- How to add them in the least amount of time?

Speedup

- The ratio between the parallel time and the sequential time.
- The sequential time must be the best possible.
- How to improve your speedup?
  - Make your parallel version faster.
  - Make your sequential version slower. ?

Efficiency

- The speedup divided by the number of processor.
- A measure of “how are doing in parallel”.
- The efficiency is always between 0 and 1.
  - Why is that?

Cost

- The product of parallel time and number of processors
- A measurement of “how much resource we have put in”.

Emulation of Parallel Algorithms

- Use a sequential method to emulate the behavior of a parallel algorithm.
- We emphasize that the computation for the speedup must be from the “best possible sequential algorithm”.
- If that is the case, we cannot have super linear speedup.

Upper Bound Analysis

- How fast we know we can do it.
- Usually derived from an algorithm.
  - There exists an algorithm so that for all inputs the running time/space used is bounded by a function of the input size.
- Usually in big-O notation.

Lower Bound Analysis

- How much time we know we must spend in order to do it
- Usually derived from mathematical analysis/adversary arguments.
- For every algorithm, there exists an input such that the algorithm must use X amount of resource in order to solve it.
- Usually in Omega notation

An Example

- How much time do we need to find the largest number among N integers?
- Use “the number of comparison” as the cost metric.
- Upper bound is?
- Lower bound is?

Another Example

- Add N numbers together in *parallel*.
- How do we improve
  - The execution time?
  - The efficiency?

Tree Method

- Using 2n-1 processors as a complete binary tree.
- The computation takes log n in time.
- The cost is O(nlogn).

A Better Tree Method

- Use 2(n/log n) –1 processors to form a tree.
- Each leaf has log n numbers to add, then it passes the result up the tree as before.
  - Computation
    - O(log n) time
  - Communication
    - O(log n) time
- The time is O(log n) time.
- The cost is O(n).

A Equally Good Algorithm

- Using 1 processor takes O(n) time.
  - The cost is O(n).
- A sequential algorithm indeed.

Optimal in What Sense?

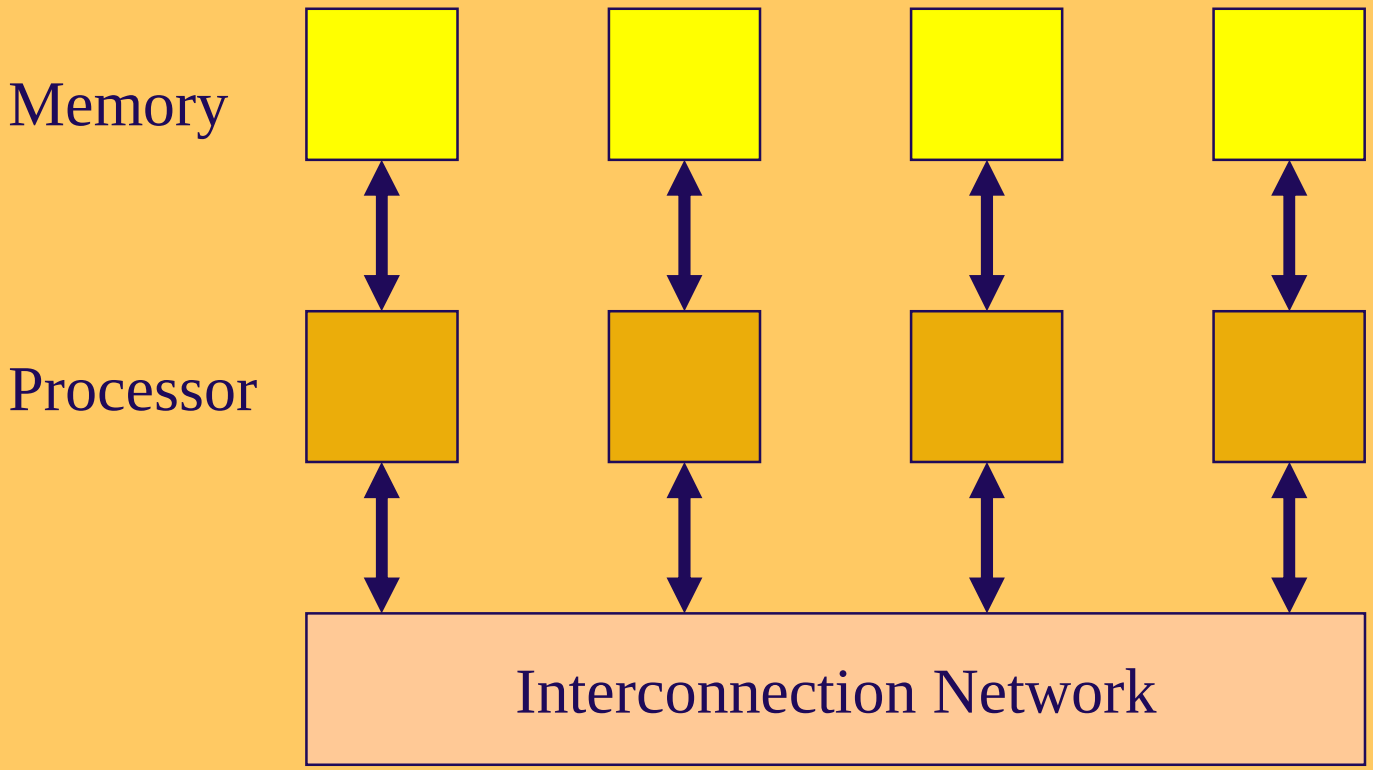
- Now we have three algorithms to choose from.
- Time optimal
- Cost optimal
- Programming effort optimal

Model of Parallel Computation

- Distributed Memory Machines
- Shared Memory Machines
- Computation Network

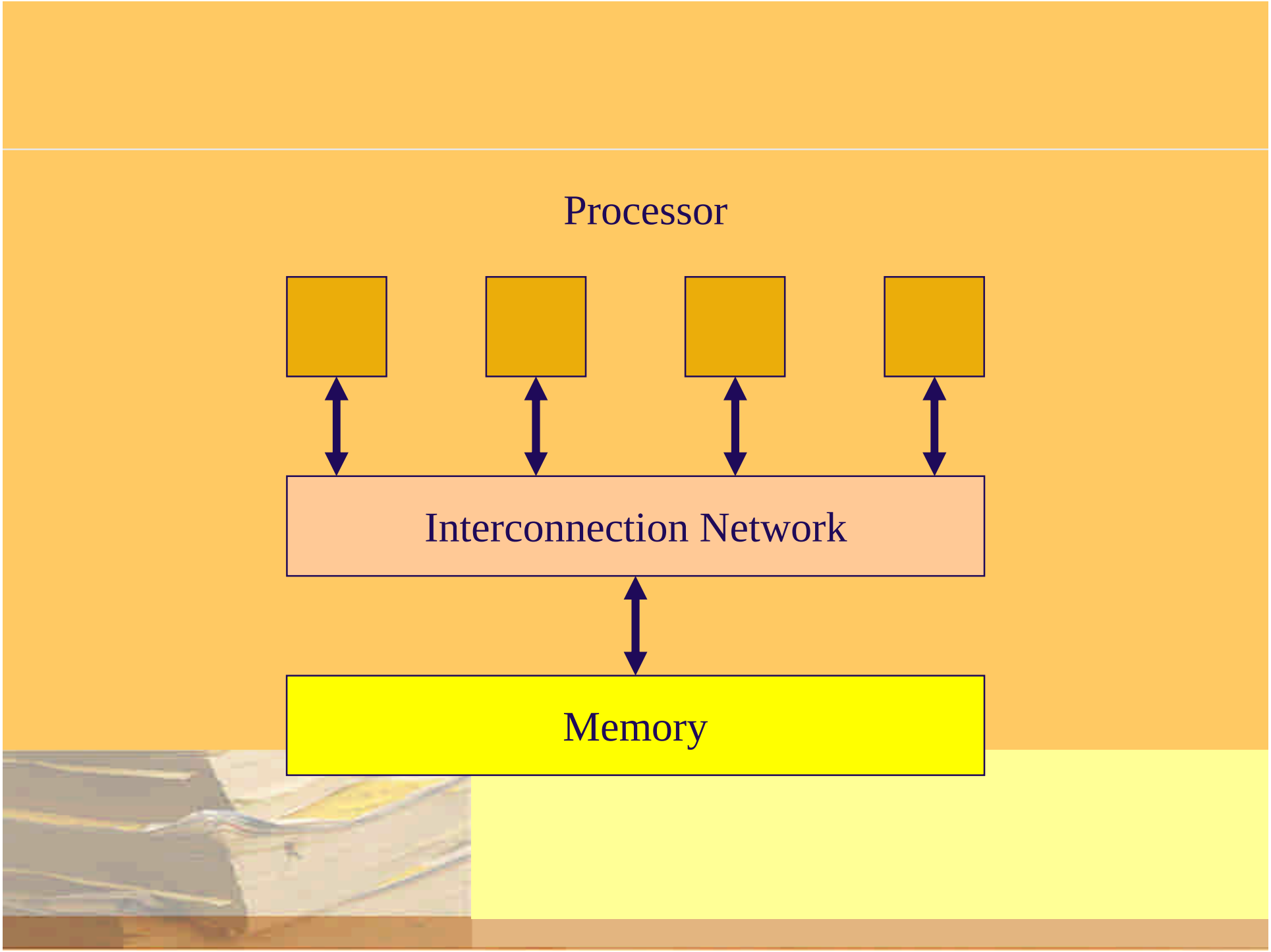
Distributed Memory

- Processors are connected loosely by an “interconnection network”.
- Each processor has its own memory.
- Processors communicate by message passing.
- The minimization of communication cost is the most important issue.
- The topology of the interconnection network is very application-dependent.


Memory
Processor
Interconnection Network

Shared Memory

- Processors are connected by a shared-memory.
- The processors can communicate by shared-memory since one processor can see the changes made in the global memory by other processors.
- The memory access conflict is the most important issue.


Processor
Interconnection Network
Memory

Computation Network

- Hardwired connection for a dedicate purpose.
- Specially designed hardware for a special computation.
  - FFT
- Sorting and permutation networks