Models of Parallel Computers

Outlines

- A Taxonomy of Parallel Architectures
- Dynamic Interconnection Networks
- Static Interconnection Networks
- Evaluating Static Interconnection Networks

A Taxonomy of Parallel Architectures

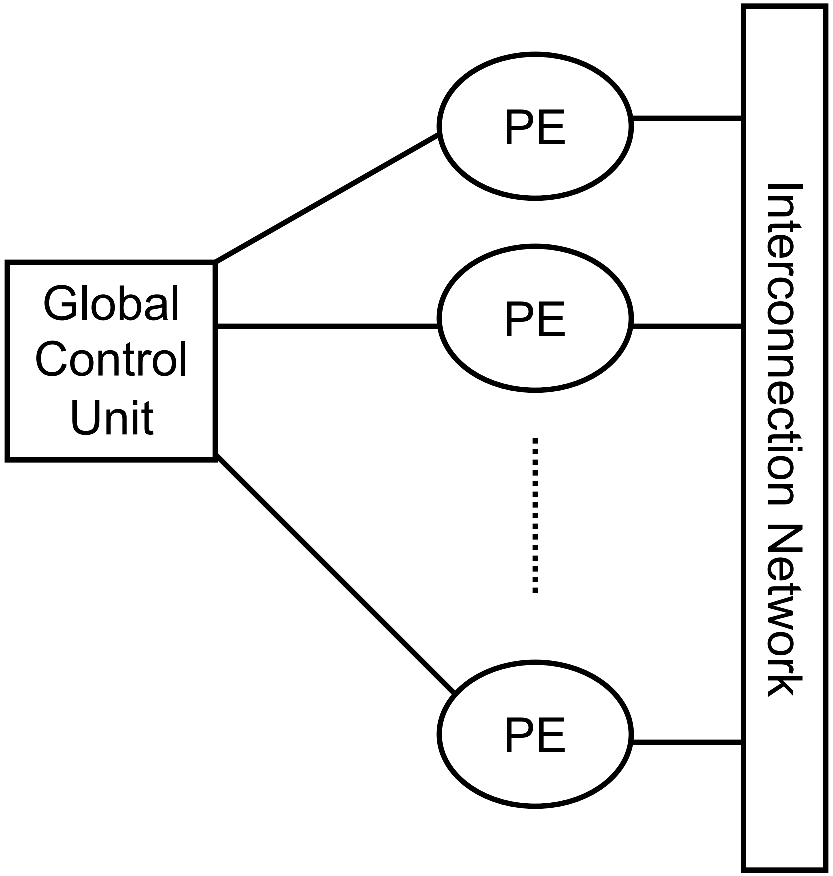
- Parallel Computers differ along various dimensions such as:
  - **Control Mechanism:** *single global control unit or multiple independent control units*
  - **Address-Space Organization:** *distributed or shared memory*
  - **Interconnected Network:** *static or dynamic*
  - **Granularity of Processors:** *course or fine grain*

Control Mechanism

- Processing units in parallel computers either operate under centralized control of a single control unit or work independently.
- Two Models:
  1) Single Instruction Stream, Multiple Data Stream (SIMD).
  2) Multiple Instruction Stream, Multiple Data Stream (MIMD).

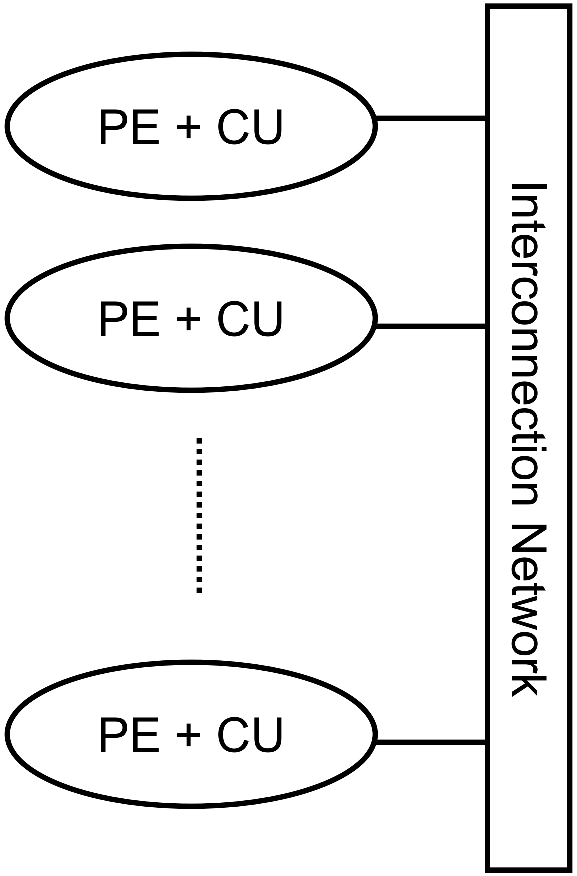
Single Instruction Stream, Multiple Data Stream (SIMD)

- A single control unit dispatches instructions to each processing unit.
- The same instruction is executed synchronously by all processing units.
- Examples of SIMD parallel computers: MPP, CM-2, MasPar MP-1 and MP-2.



Multiple Instruction Stream, Multiple Data Stream (MIMD)

- Each processor is capable of executing a different program independent of other processor.
- Examples of MIMD computers: Cosmic Cube, nCUBE2, CM-5.



Comparing SIMD with MIMD Computers

1) SIMD requires less hardware than MIMD because they have only one global control unit.

2) SIMD requires less memory because only one copy of program needs to be stored.

3) MIMD stores program and operating system at each processor.

4) SIMD computers are naturally suited for data-parallel programs; that is programs in which the same set of instructions are executed on a large data set.

Comparing SIMD with MIMD Computers (Continues)

5) SIMD computers require less start up time for communicating with neighbouring processors.

6) A drawback of SIMD computers is that different processors cannot execute different instructions in same clock cycle.

Comparing SIMD with MIMD Computers (Continues)

7) Data-parallel programs in which significant parts of the computation are contained in conditional statements are better suited to MIMD computers than to SIMD computers.

8) Individual processors in an MIMD computer are more complex, because each processor has its own control unit.

9) CPU used in SIMD computers has to be specially designed.

10) Processors in MIMD computers may be both cheaper and more powerful than processors in SIMD computers.

Comparing SIMD with MIMD Computers (Continues)

11) SIMD computers are better suited to parallel programs that require frequent synchronization.

12) Many MIMD computers have extra hardware to provide fast synchronization, which enables them to operate in SIMD mode as well.

- ➢ Examples: CM-5 and DADO.

Address-Space Organization

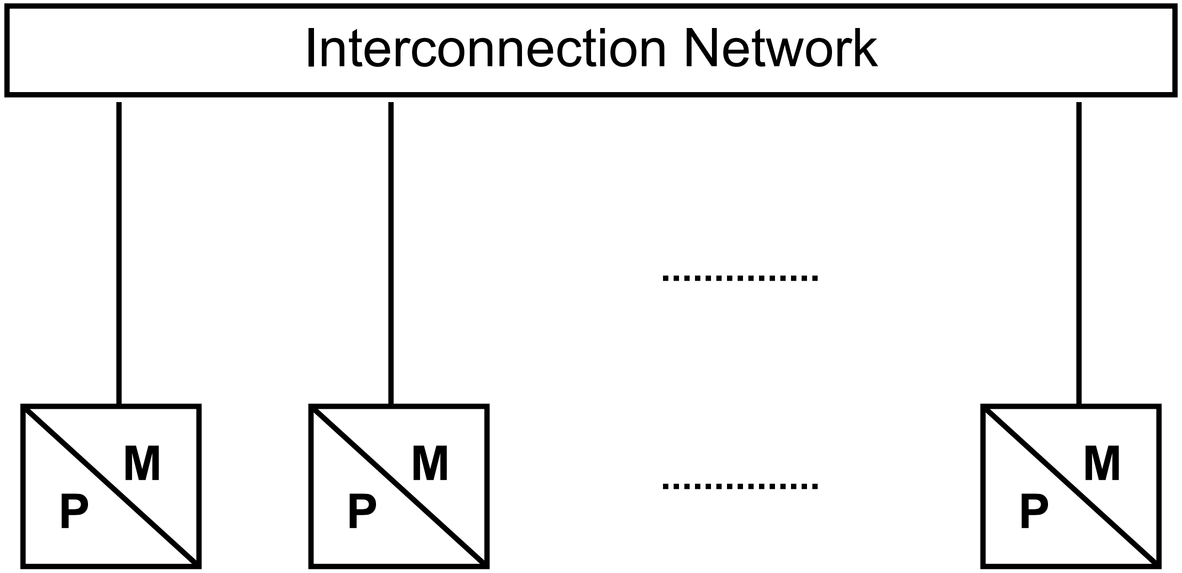
- Two Models:

  1) Message-Passing Architecture.
  2) Shared-Address Space Architecture.

Message-Passing Architecture

- Processors are connected using a message-passing interconnection network.
- Each processor has its own memory called local or private memory, which is accessible only to that processor.
- This architecture is referred to as a distributed-memory or private-memory architecture.

A Typical Message-Passing Architecture



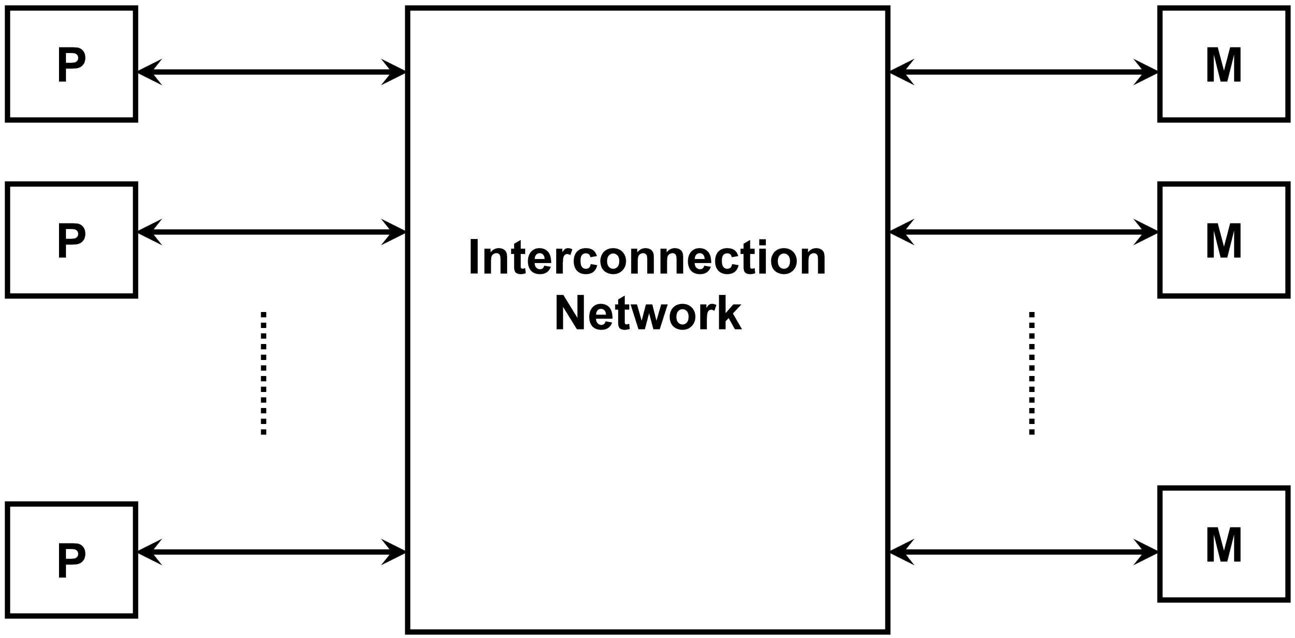
P: Processor   M: Memory

- MIMD message-passing computers are referred to as multi-computers.
- Examples: Cosmic Cube, CM-5, and nCube2.

Shared-Address Space Architecture

- MIMD shared-address space computers are referred to as multiprocessors.
- Three Models:
  1) Uniform-Memory-Access (UMA).
  2) Non-Uniform-Memory-Access with local and global memories (NUMA).
  3) Non-Uniform-Memory-Access with local memory only (NUMA).

Uniform-Memory-Access Shared-Address Space Computer (UMA)



- These architectures are called shared-memory parallel computers.
- Examples: C.mmp and NYU Ultracomputer

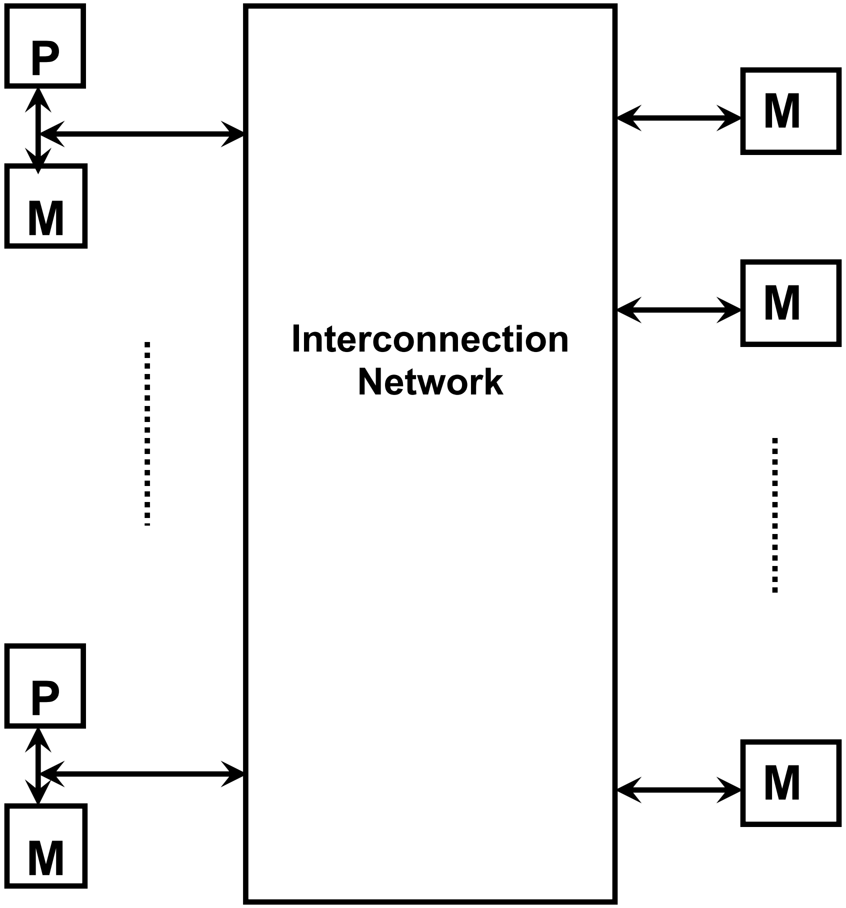
Uniform-Memory-Access Shared-Address Space Computer (UMA) Cont…

- Drawbacks:
  - The bandwidth of interconnection network must be substantial to ensure good performance.
  - Memory access through interconnection network can be slow, since a read or write request may have to pass through multiple stages in network.

Non-Uniform-Memory-Access Shared-Address Space Computer with local and global memories (NUMA)

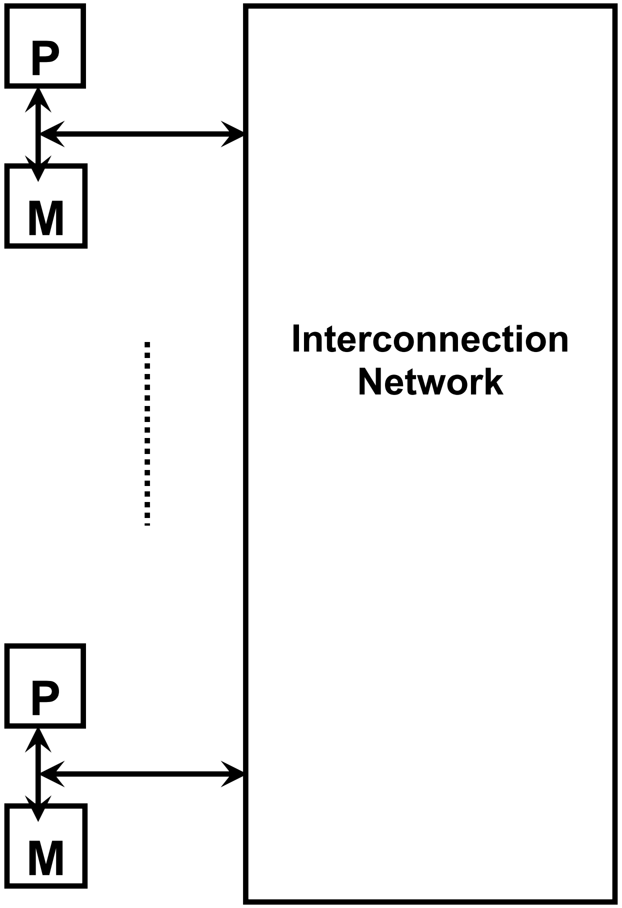
- Provide each processor with a local memory, which stores program being executed on processor and any non-shared data structures.

- Global data structures are stored in shared memory, which eliminates repeated memory references across interconnection network and improves performance.



Non-Uniform-Memory-Access Shared-Address Space Computer with local memory only (NUMA)

- Local memory access time is much smaller than remote memory access time.



Shared-Address Space Computers

- Shared-address space computers are classified into two categories based on amount of time a processor takes to access local and global memories:

1) UMA computer: If the time taken by a processor to access any memory word in the system is identical.

2) NUMA computer: If the time to access a remote memory bank is longer than the time to access a local one.

Shared-Address Space Computers (Cont…)

- Most shared-address space computers have a local cache at each processor to increase their effective processor-memory bandwidth.

- Cache coherence occurs when a processor modifies a shared variable in its cache; after this modification, different processors have different values of the variable, unless copies of the variable in the other caches are simultaneously updated.

NUMA versus Message-Passing

- NUMA architecture is similar to a message-passing architecture; that is memory is physically distributed in both.

- The major difference between NUMA and Message-Passing is that a NUMA provides hardware support for read and write access to other processors’ memories, whereas in a Message-Passing, remote access must be emulated by explicit message passing.

Shared-Address Space versus Message-Passing

- Shared-Address Space computers provide greater flexibility in programming than Message-Passing.

- A Shared-Address Space tends to be more expensive than Message-Passing.

Interconnection Networks

- Shared-Address Space computers and Message-Passing computers can be constructed by connecting processors and memory units using a variety of interconnection networks

- Interconnection networks can be classified as static or dynamic

Static & Dynamic Networks

- Static networks consist of point-to-point communication links among processors and are also referred to as direct network
- Static networks are used to construct message-passing computers
- Dynamic networks are built using switches and communication links
- Dynamic networks are referred to as indirect networks and are used to construct shared-address space computers

Processor Granularity

- Coarse-Grain computers composed of a small number of very powerful processors
  - Example: Cray Y-MP 8-16 processors each capable of Gflops (10^9 flop per second)
- Fine-Grain computers composed of a large number of less powerful processors
  - Example: CM-2 up to 65,536 one-bit processors

Processor Granularity Cont…

- The granularity of a parallel computer can be defined as the ratio of the time required for a basic computation operation over the time required for a basic communication

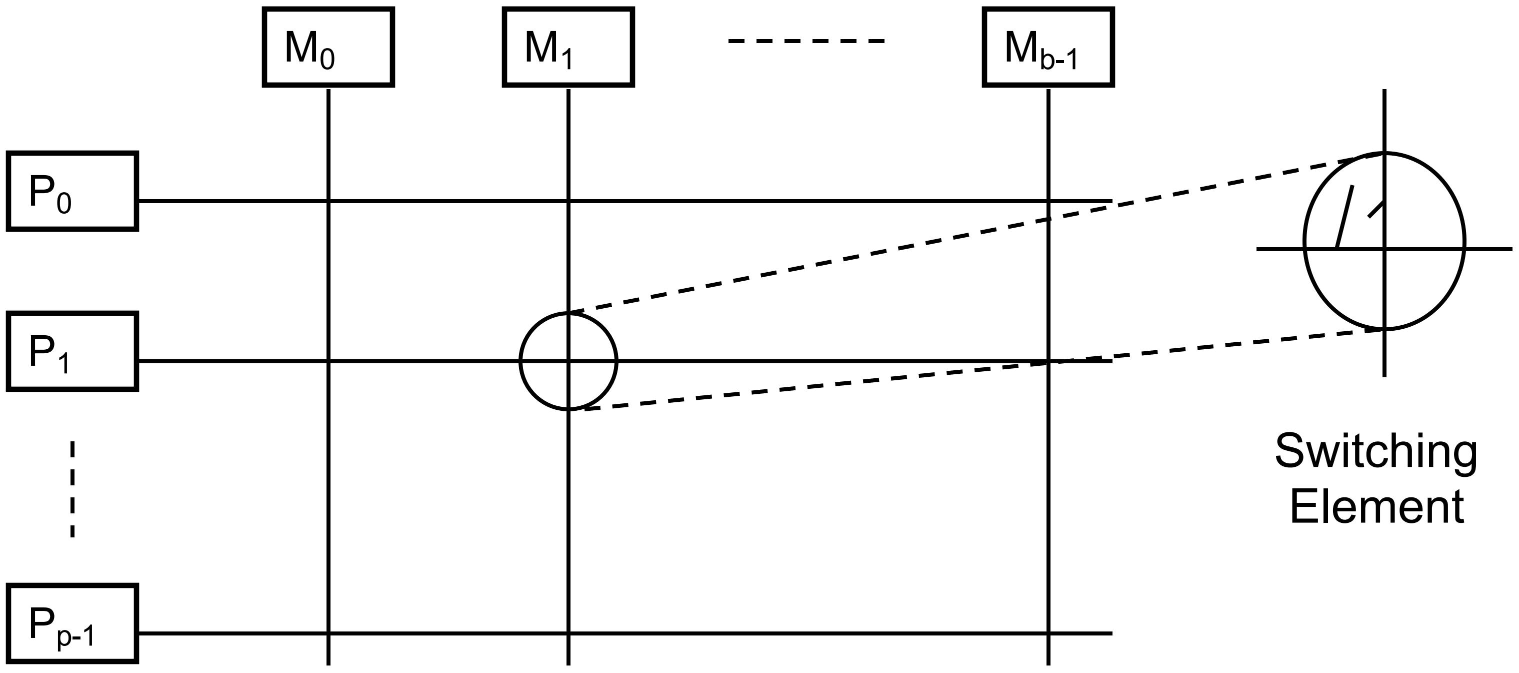
  (That is computation / communication)
- Parallel computers for which this ratio is small are suitable for algorithms requiring frequent communication; that is, algorithms in which the grain size of the computation is small (fine-grain)
- Parallel computers for which this ratio is large are suited to algorithms that do not require frequent communication (coarse-grain)

Dynamic Interconnection Networks

1) Crossbar Switching Networks
2) Bus-based networks
3) Multistage Interconnection Networks

Crossbar Switching Networks

- A completely non-blocking crossbar switch connecting ***p*** processors to ***b*** memory banks.

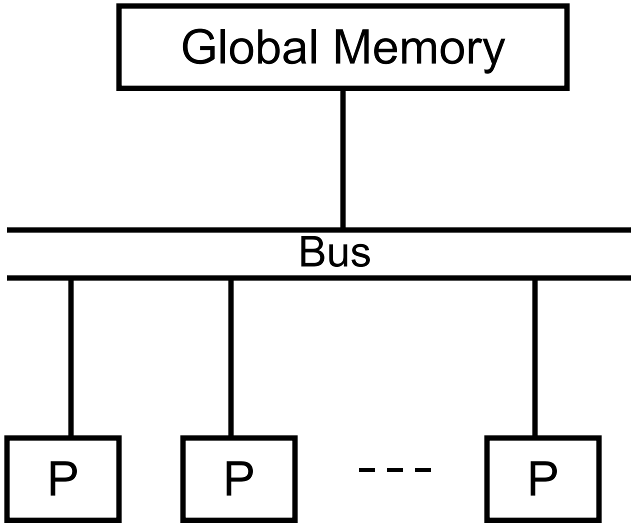


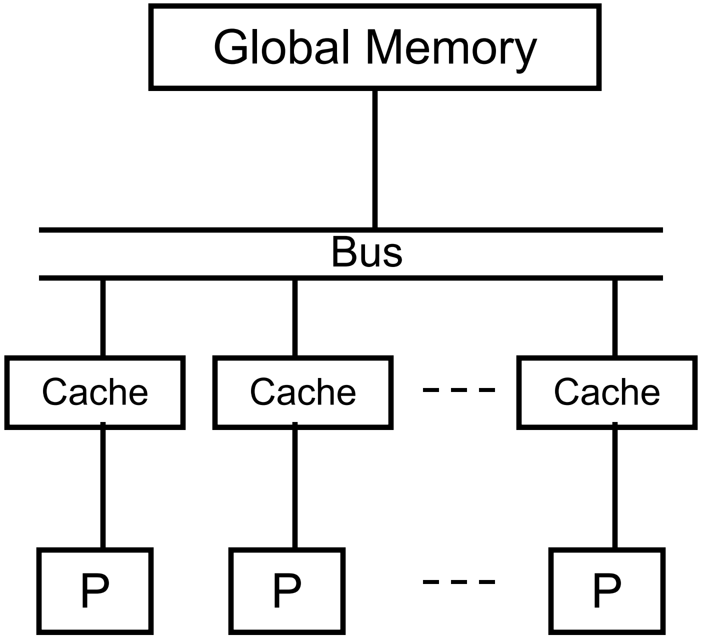
Crossbar Switching Networks (Continues)

- Crossbar switching network is a non-blocking network in the sense that the connection of a processor to a memory bank does not block the connection of any other processor to any other memory bank.
- Normally, ***b*** memory banks is greater than or equal to ***p*** processors, so that each processor has at least one memory bank to access.
- Total number of switching elements required to implement such a network is Θ(p x b).

Bus-Based Networks

- A typical bus-based architecture with no cache; and with cache memory at each processor.





Bus-Based Networks (Continues)

- Whenever a processor accesses global memory, that processor generates a request over the bus. The data is then fetched from memory over the same bus.

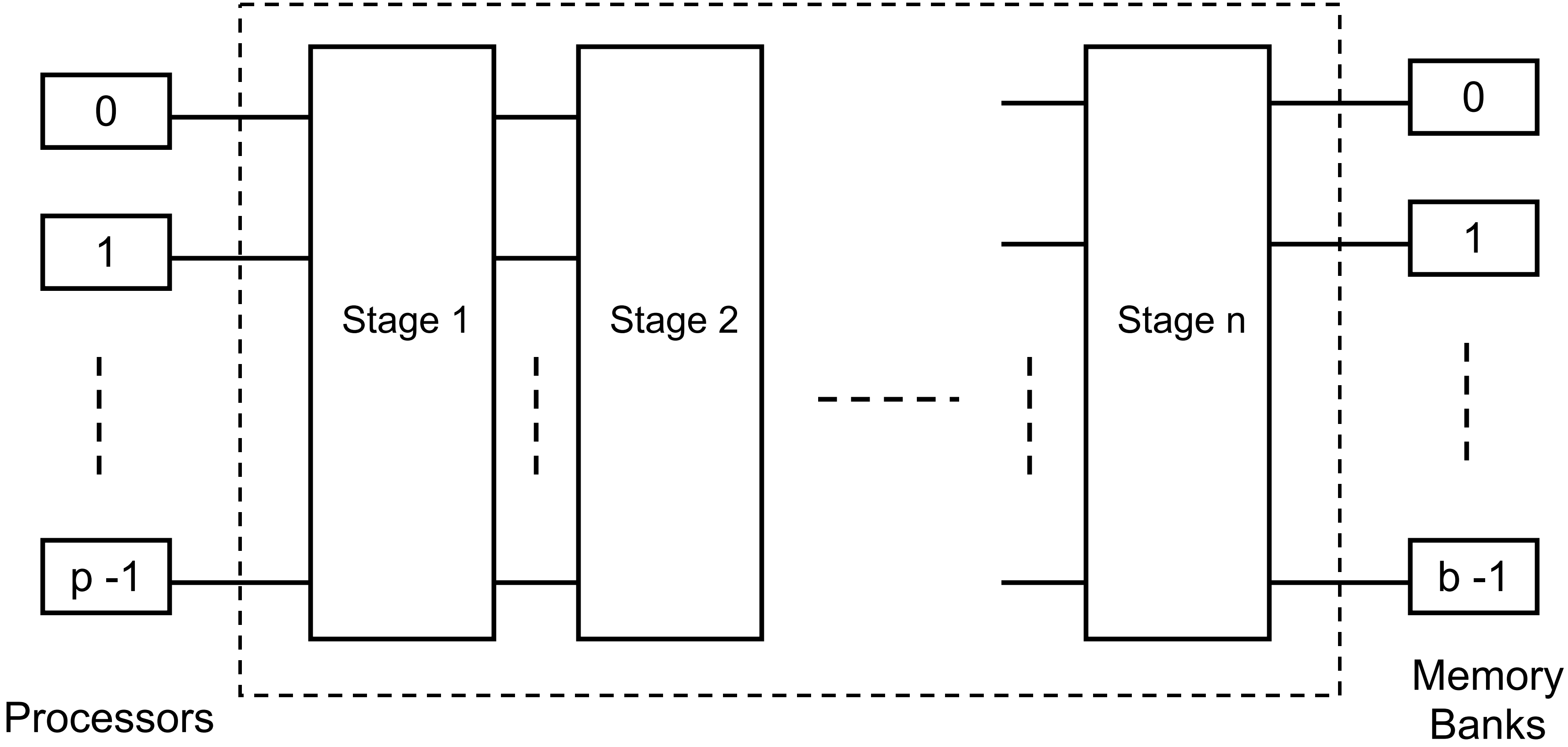
- Bus contention: each processor spends an increasing amount of time waiting for memory access while the bus is in use by other processor due to large number of processors sharing same bus.

Bus-Based Networks (Continues)

- To reduce bus contention, each processor has a local cache memory:
  - Local cache memory reduces the total number of accesses to global memory, because when a reference is made to a memory location, subsequent references are likely to be made to memory locations in the neighborhood of this location, were fetched into a processor cache memory too (locality in space).

Multistage Interconnection Networks

- Schematic of a multistage network consisting of ***p*** processors and ***b*** memory banks.

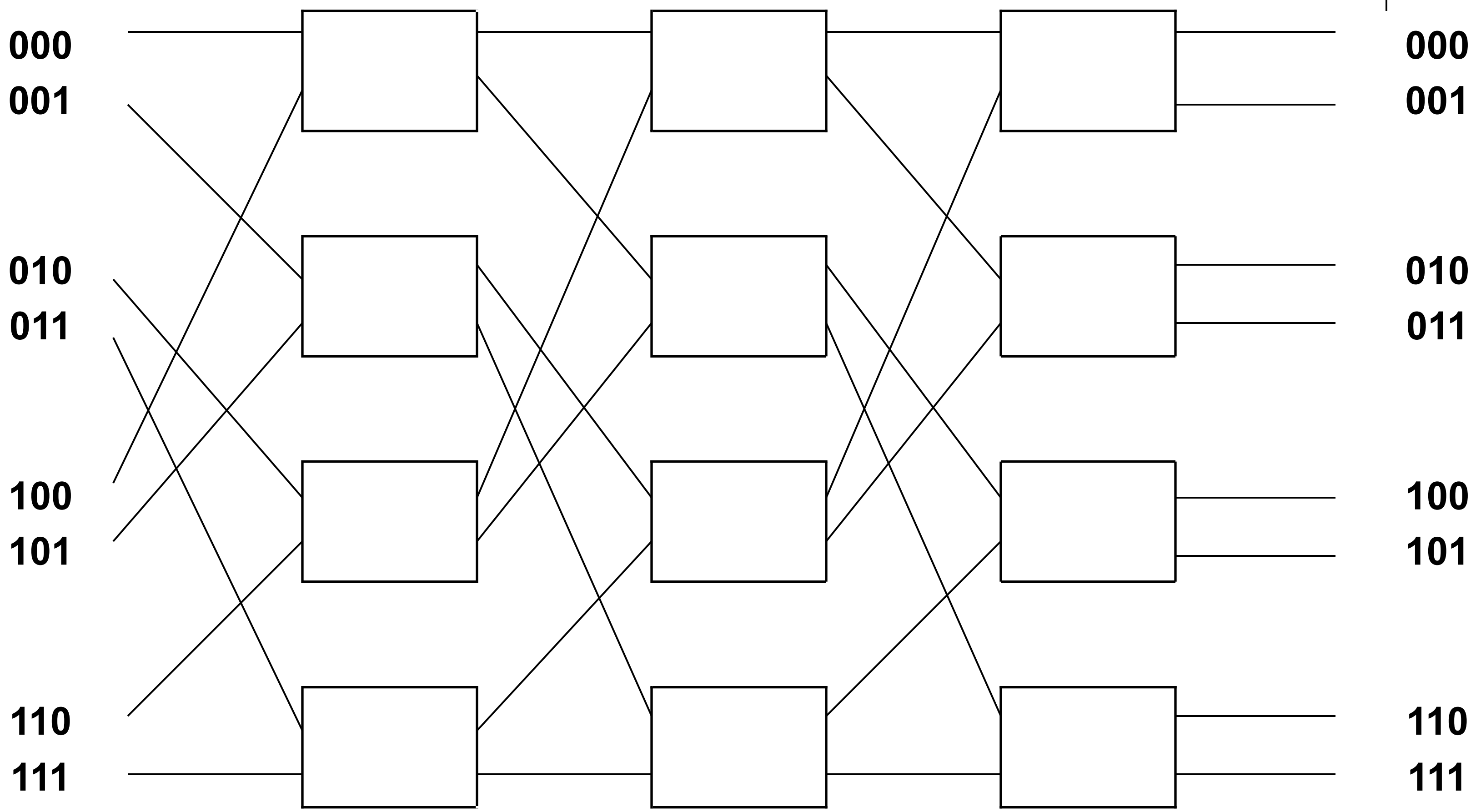


Multistage Interconnection Network

Example: Omega Network

- Omega network consists of log p stages; where p is the number of processors and the number of memory banks
  - Each stage of Omega network consists of an interconnection pattern that connects ***p*** inputs and ***p*** outputs.
  - This interconnection pattern is called perfect shuffle.
- In each stage of an Omega Network, a perfect shuffle interconnection pattern feeds into a set of (p / 2) switching elements.

A Complete Omega Network connecting 8 inputs (processors) and 8 outputs (memory banks)



Processors
(Input nodes)

Interconnection Network
(Switching elements)

Memory Banks
(Output nodes)

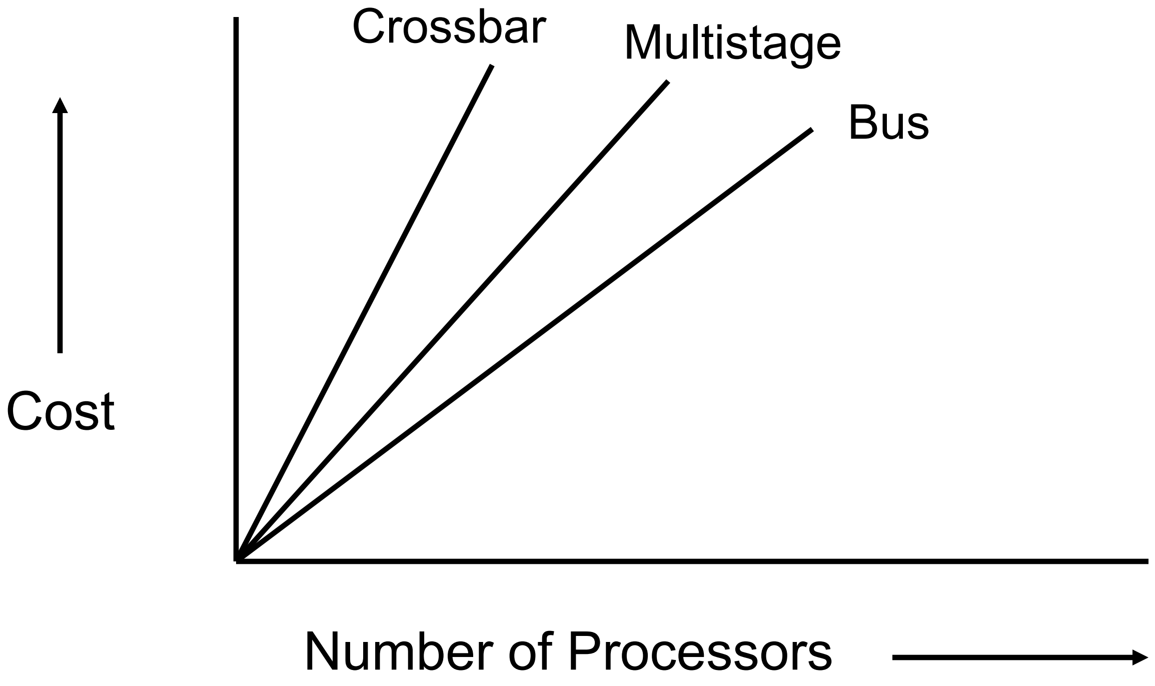
Omega Network (Continue)

- An Omega Network has $(p / 2) \times (\log_2 p)$ switching elements and the cost of such a network grows as $\Theta(p \log_2 p)$.
- Example: For a Complete Omega Network connecting 8 inputs (8 processors) and 8 outputs (8 memory banks) we have $(8 / 2) \times (\log_2 2^3) = 12$ switching elements and a cost of $\Theta(8 \times \log_2 2^3) = 8 \times 3 = 24$.
- Note: this cost $\Theta(p \log_2 p)$ is less than $\Theta(p^2)$ cost of a complete crossbar switch.

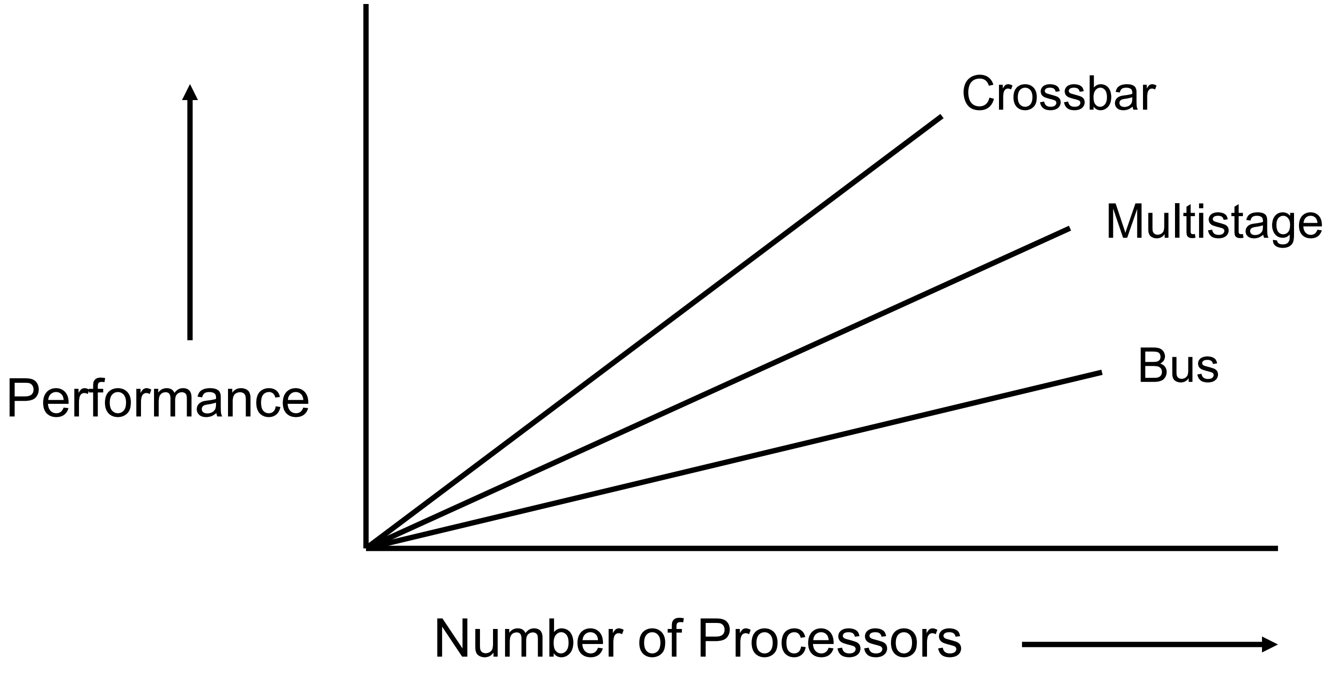
Crossbar, Shared Bus, & Multistage Interconnection Networks

- The crossbar interconnection network is scalable in terms of performance but un-scalable in terms of cost.
- The shared bus network is scalable in terms of cost but un-scalable in terms of performance.
- The multistage interconnection networks is more scalable than the bus in terms of performance and more scalable than crossbar in terms of cost.

Cost versus Number of Processors



Performance versus Number of Processors

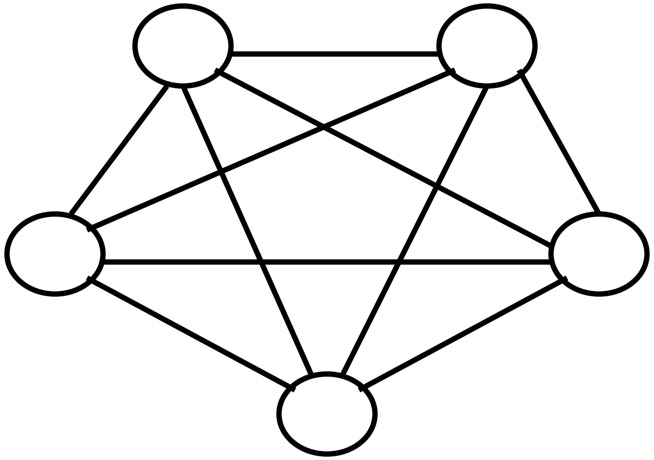


Static Interconnection Networks

- Message-Passing architecture typically use static interconnection networks to connect processors.
- Types of Static Interconnection Networks:
  - Completely-Connected Network
  - Star-Connected Network
  - Linear Array and Ring
  - Mesh Network
  - Tree Network
  - Hypercube Network
- Evaluating Static Interconnection Networks:
  - Diameter; Connectivity; Bisection Width; Bisection Bandwidth; and cost.

Completely-Connected Network

- Each processor has a direct communication link to every other processor in the network.
- A Completely-Connected Network of 5 Processors:

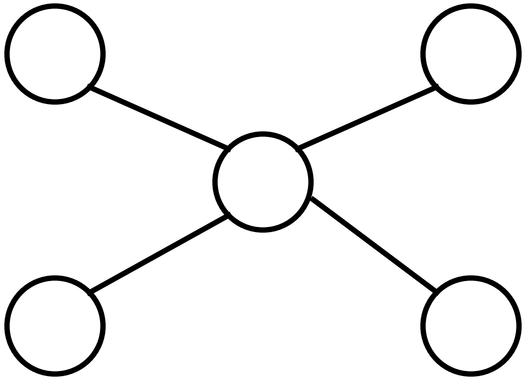


Advantages of Completely-Connected Network

1) In this network a processor can send a message to another processor in a single step, since a communication link exists between them.
2) The communication between any input / output pair does not block communication between any other pair.
3) The network supports communications over multiple channels originating at the same processor.

Star-Connected Network

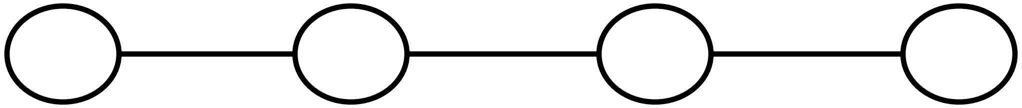
- One processor acts as the central processor and every other processor has a communication link connecting it to this processor.
- A Star-Connected Network of 5 Processors:



- Communication between any pair of processors is routed through the central processor.
- The central processor is the bottleneck in the star topology.

Linear Array and Ring

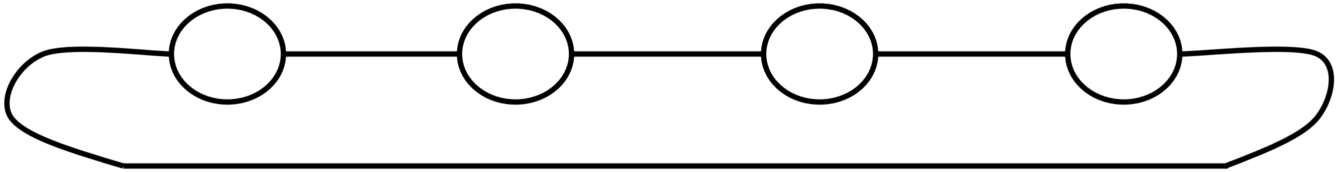
- A 4-Processor Linear Array:



- Each processor in this network (except processors at the ends) has a direct communication link to 2 other processors.
- A wraparound connection is provided between the processors at the ends.
- A linear array with a wraparound connection is called a ring.

Ring (Continues)

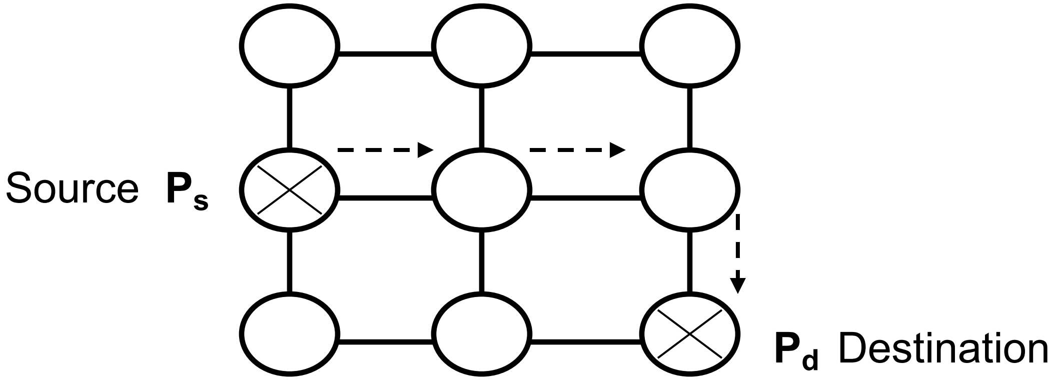
- A 4-Processor Ring:



- One way of communicating a message between processors is by repeatedly passing it to the processor immediately to the right (or left, depending on which direction yields a shorter path) until it reaches its destination.

Mesh Network

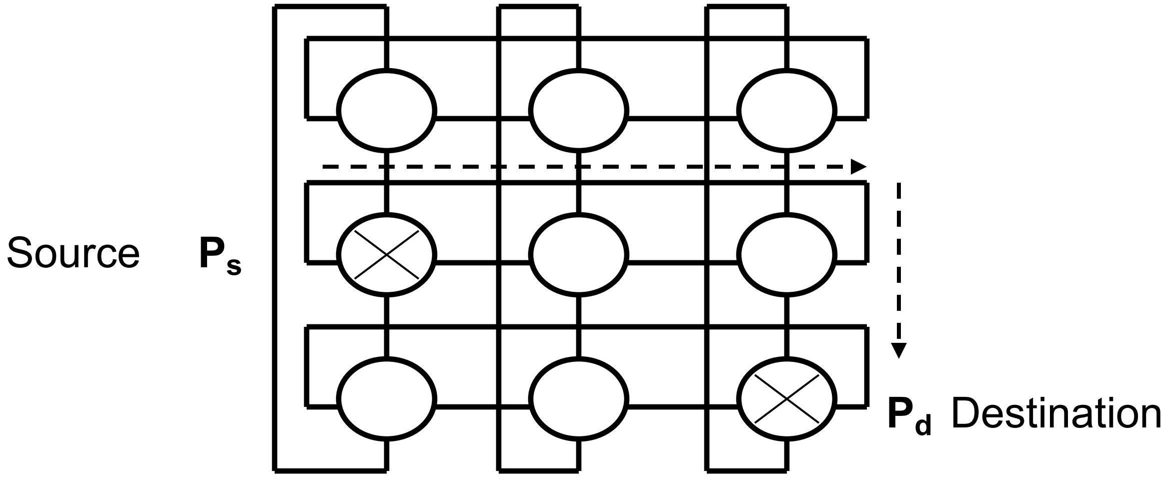
- A two-dimensional mesh of 9 processors:



- Routing a message from processor P_s to processor P_d.
- In a 2-D mesh, each processor has a direct communication link connecting it to 4 other processors.

Mesh Network (Continues)

- A wraparound mesh or torus (2-D):
  - Routing a message from processor P_s to processor P_d.



Mesh Network (Continues)

- The Cray T3D is an example of 3-D Mesh:

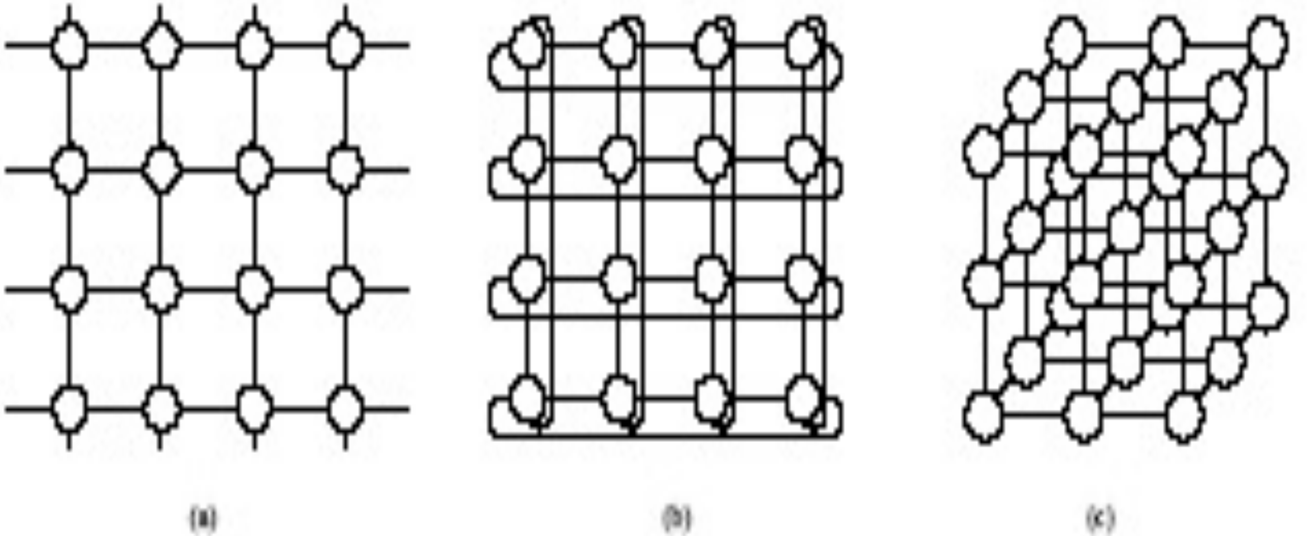


Figure 2.16 Two and three dimensional meshes: (a) 2-D mesh with no wraparound; (b) 2-D mesh with wraparound link (2-D torus); and (c) a 3-D mesh with no wraparound.

Tree Network

- A tree network is one in which there is only one path between any pair of processors.
- Both linear arrays and star-connected networks are special cases of tree network.
- Complete binary tree networks and message routing in them:
  - Static tree networks: a processor at each node of the tree.
  - Dynamic tree networks: nodes at intermediate levels are switching elements and the leaf nodes are processors.

Static vs. Dynamic Tree Networks

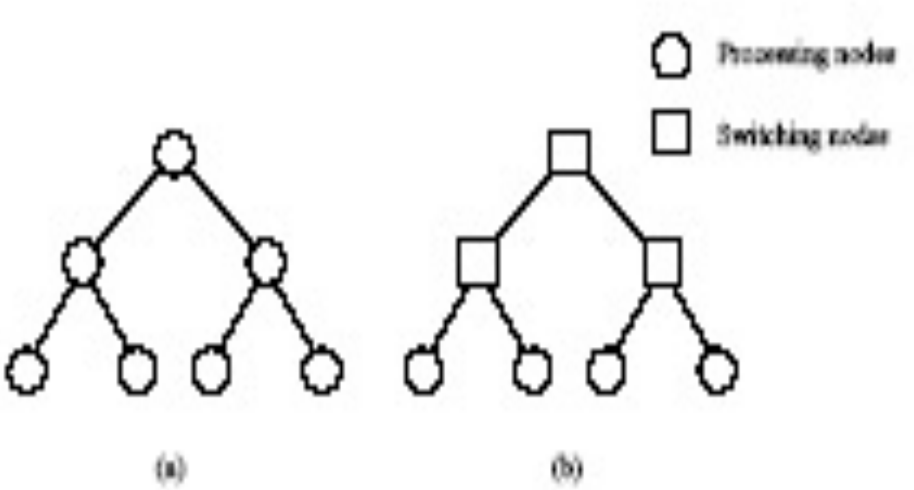


Figure 2.18 Complete binary tree networks: (a) a static tree network; and (b) a dynamic tree network.

Tree Networks (Continues)

- Tree networks suffer from a communication bottleneck at higher levels of the tree.
- For example, when many processors in the left sub-tree of a node communicate with processors in the right sub-tree, the root node has to handle all the messages.
- This problem can be solved by increasing the number of communication links between processors that are closer to the root.

Example: A Fat Tree Network

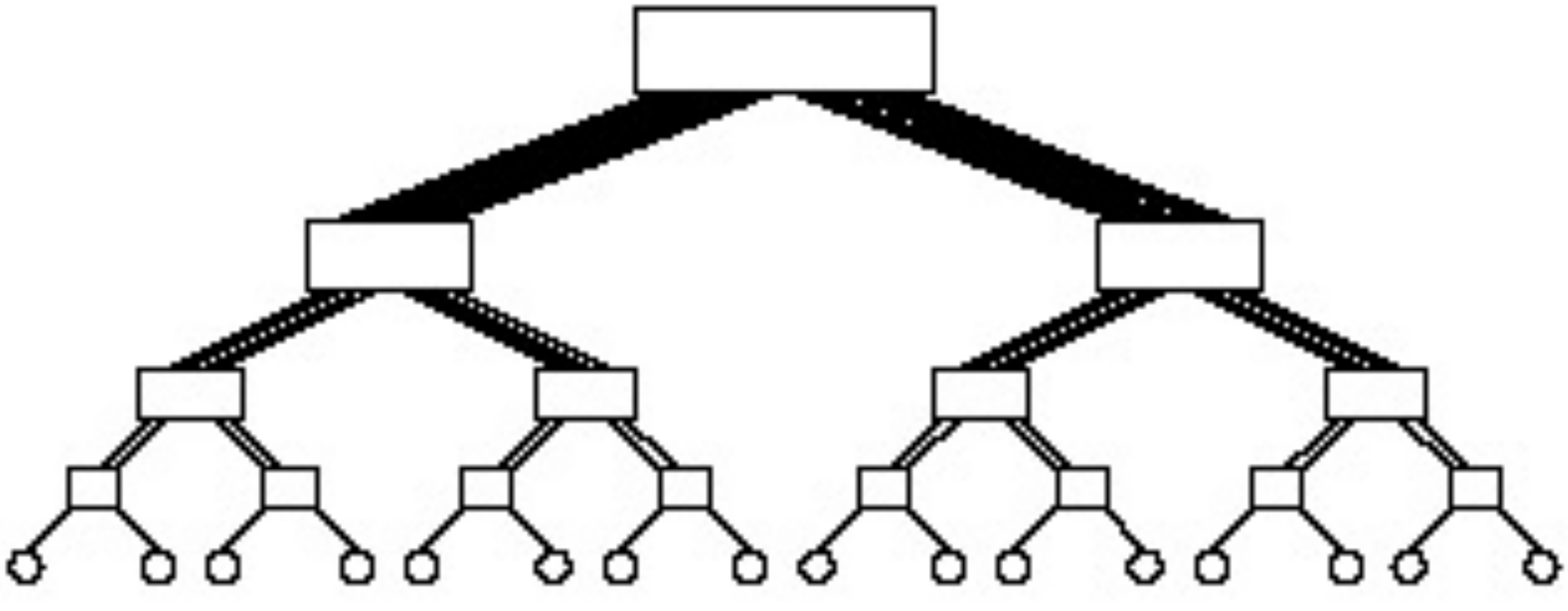
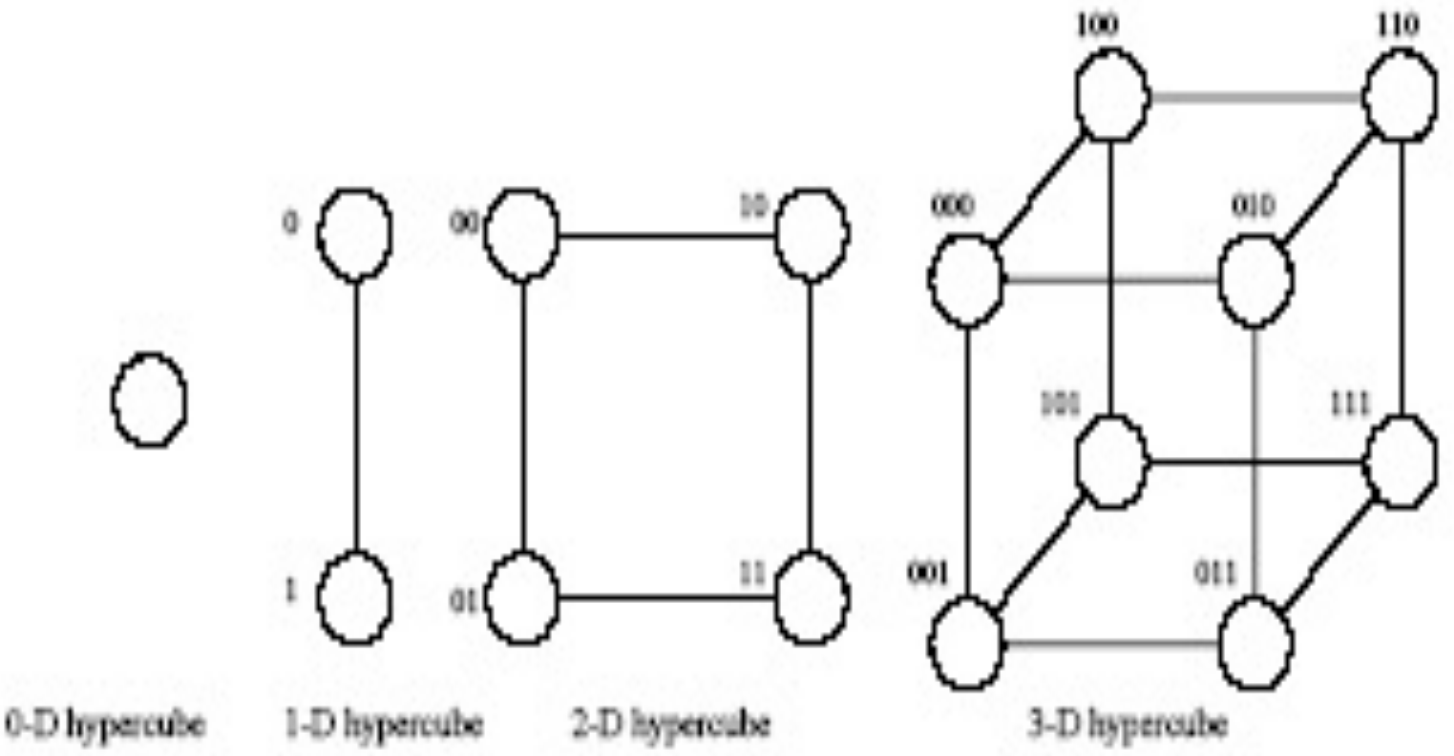


Figure 2.19 A fat tree network of 16 processing nodes.

Hypercube Network

- A hypercube is a multidimensional mesh of processors with exactly 2 processors in each dimension.
- A d-dimensional hypercube consists of $p = 2^d$ processors.
- A zero-dimensional hypercube is a single processor.
- A one-dimensional hypercube is constructed by connecting 2 zero-dimensional hypercubes.
- In general, a (d+1)-dimensional hypercube is constructed by connecting the corresponding processors of 2 d-dimensional hypercubes.

Hypercubes of Dimensions Zero to Three



Hypercube Labels

- When a (d+1)-dimensional hypercube is constructed by connecting 2 d-dimensional hypercubes, the labels of the processors of one hypercube are prefixed with 0 and those of the second hypercube are prefixed with a 1.

Properties of Hypercube Network

1) Two processors are connected by a direct link if and only if the binary representation of their labels differ at exactly one bit position.
2) In a d-dimensional hypercube, each processor is directly connected to d other processors.
3) A d-dimensional hypercube can be partitioned into 2 (d – 1)-dimensional sub-cubes.

Properties (Continues)

4) Hamming distance between 2 labels (***s*** & ***t***) of 2 processors in hypercube is the total number of bit positions at which these 2 labels differ.

Example: Hamming Distance:

$\underline{01}1$ and $\underline{10}1 = 2$

$\underline{101}$ and $\underline{010} = 3$

Properties (Continues)

- Hamming distance between ***s*** and ***t*** labels is the number of bits that are 1 in the binary representation of ***s*** X-OR ***t***.
- The number of communication links in the shortest between 2 processors is the Hamming Distance between their labels.

Evaluating Static Interconnection Networks

1) Diameter: is the maximum distance between any 2 processors in the network.
   - The distance between 2 processors is defined as the shortest path (in terms of number of links) between them.
   - Networks with smaller diameters are better, since distance largely determines communication time.

Evaluating Static Interconnection Networks

2) Connectivity: is a measure of the multiplicity of paths between any 2 processors.

- A network with high connectivity is desirable, because it lowers contention for communication resources.
- Arc connectivity: is the minimum number of arcs that must be removed from the network to break it into 2 disconnected networks.

Evaluating Static Interconnection Networks

3) Bisection Width: is the minimum number of communication links that have to be removed to partition the network into 2 equal halves.
4) Bisection Bandwidth of a network is the minimum volume of communication allowed between any 2 halves of the network with an equal number of processors.

   OR

   Bisection Bandwidth = Bisection Width x Channel Bandwidth

Evaluating Static Interconnection Networks

- Channel bandwidth: is the peak rate at which data can be communicated between the ends of a communication link.

  OR

  Channel Bandwidth = Channel rate x channel width
- Channel width is equal to the number of physical wires in each communication link.

  OR is the number of bits that can be communicated simultaneously over a link connecting 2 processors
- Channel rate: is the peak rate at which a single physical wire can deliver bits.

Evaluating Static Interconnection Networks

5) Cost: is the number of communication links or the number of wires required by the network.
6) Size: is the number of nodes in the network.

Characteristics of various static network topologies connecting p processors

Network	Diameter	Bisection Width	Arc Connectivity	Cost
Completely Connected	1	$p^2/4$	p – 1	p(p-1)/2
Star	2	1	1	p - 1
Complete Binary Tree	2log((p+1)/2) Or 2FL(logp)	1	1	p - 1
Linear Array	P-1	1	1	p - 1
Ring	FL(p/2)	2	2	p
2-D Mesh	2(SqR(p)–1)	SqR(p)	2	2 (p-SqR(p))
2-D Mesh wraparound	2FL(SqR(p)/2)	2 SqR(p)	4	2p
Hypercube	logp	p/2	logp	(plogp)/2