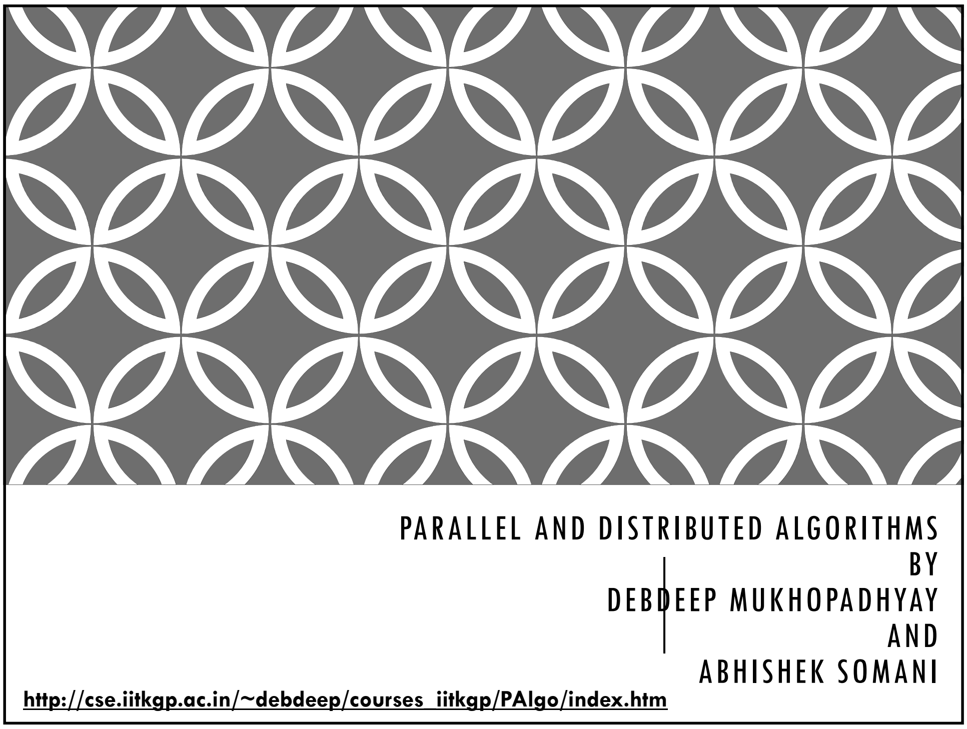
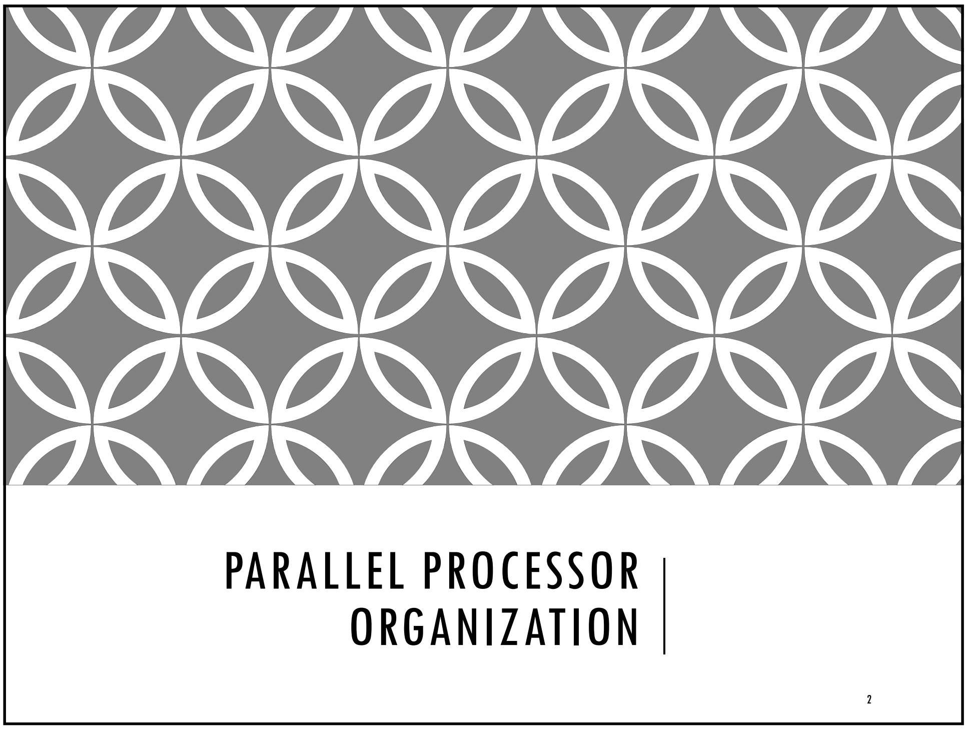


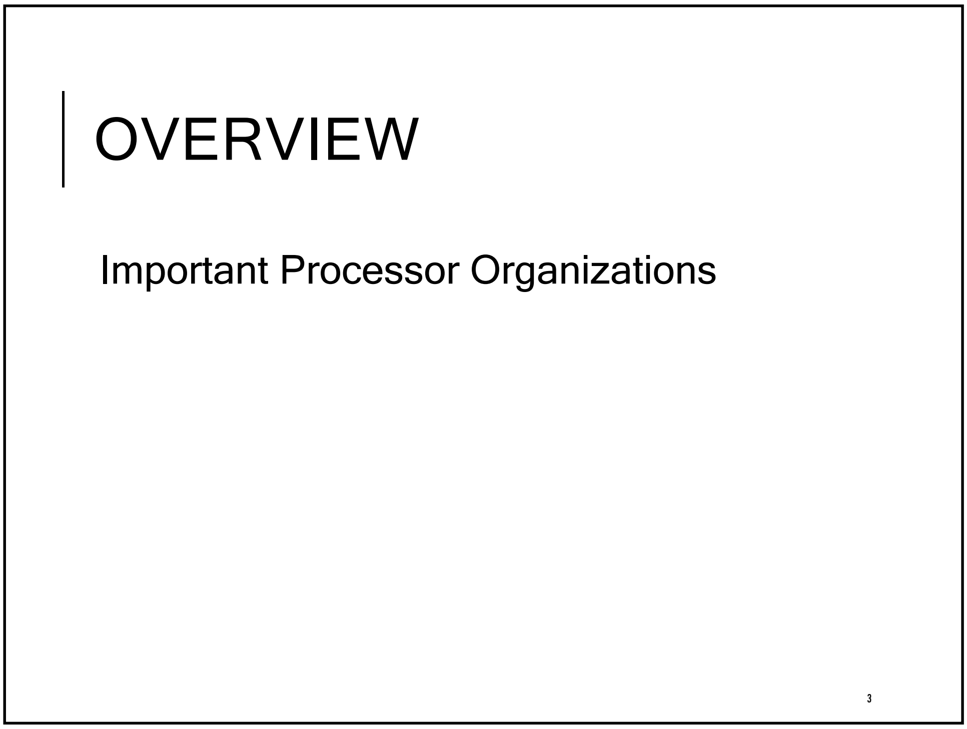
PARALLEL AND DISTRIBUTED ALGORITHMS BY DEBDEEP MUKHOPADHYAY AND ABHISHEK SOMANI

http://cse.iitkgp.ac.in/~debdeep/courses_iitkgp/PAlgo/index.htm



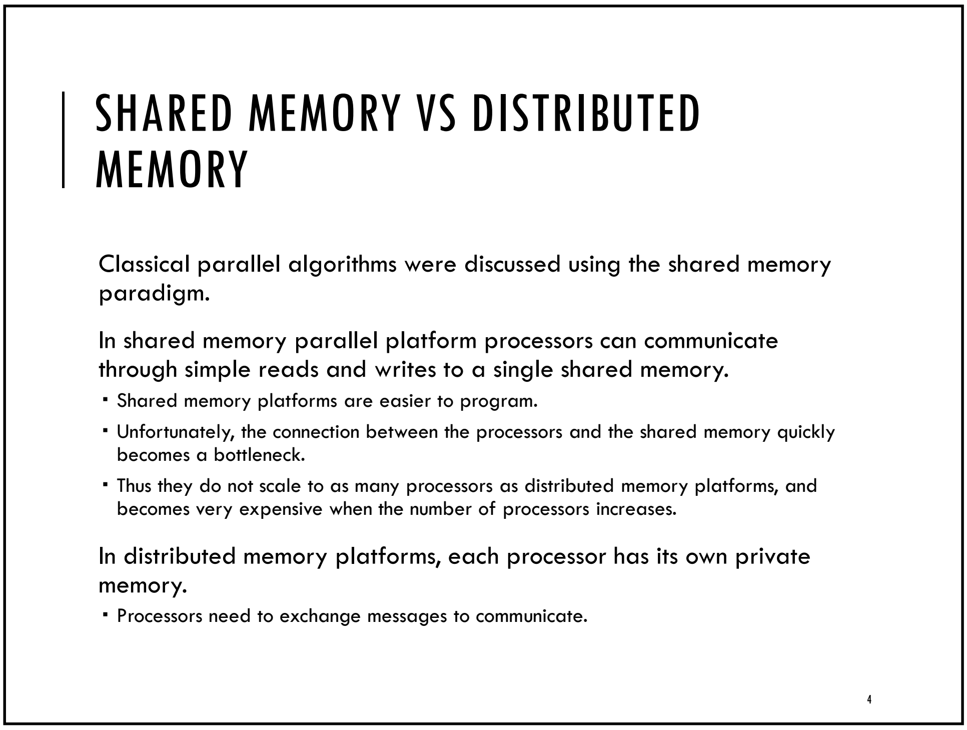
PARALLEL PROCESSOR ORGANIZATION

2



OVERVIEW

Important Processor Organizations



SHARED MEMORY VS DISTRIBUTED MEMORY

Classical parallel algorithms were discussed using the shared memory paradigm.

In shared memory parallel platform processors can communicate through simple reads and writes to a single shared memory.

- Shared memory platforms are easier to program.
- Unfortunately, the connection between the processors and the shared memory quickly becomes a bottleneck.
- Thus they do not scale to as many processors as distributed memory platforms, and becomes very expensive when the number of processors increases.

In distributed memory platforms, each processor has its own private memory.

- Processors need to exchange messages to communicate.

IMPORTANCE OF NETWORK TOPOLOGIES

One cannot really design parallel algorithms without understanding

- the underlying parallel architectures and
- by means of which components are connected to each other.

NETWORK TOPOLOGY

Ways in which set of nodes are connected to each other.

- Essentially a discrete graph : a set of nodes and edges.

Network topologies arise in parallel architectures and parallel algorithms in several contexts:

- Could describe the interconnection among multiple processors and memory modules.
- Can also describe the communication pattern among a set of parallel processes.

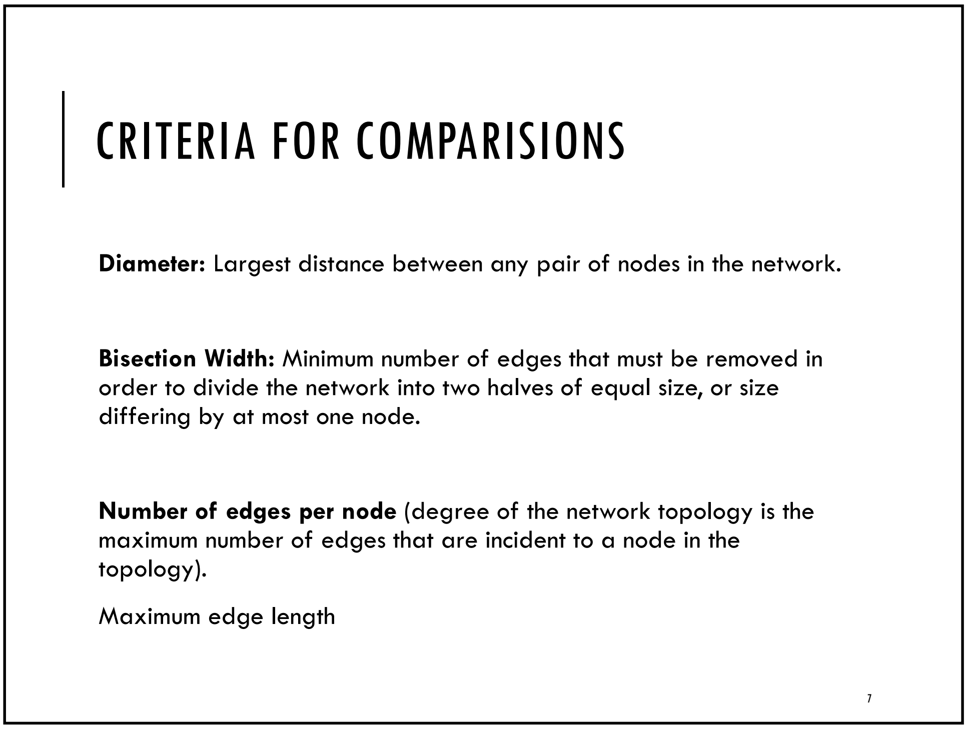
We hence first observe the properties of the network as mathematical entities, agnostic of these details.

Can be represented by a graph in which:

- the nodes (vertices) represent processors and
- edges represent communication paths between a pair of processors.

Our goal will be to implement and analyze the parallel algorithms on these organizations.

How do we compare among the organizations?



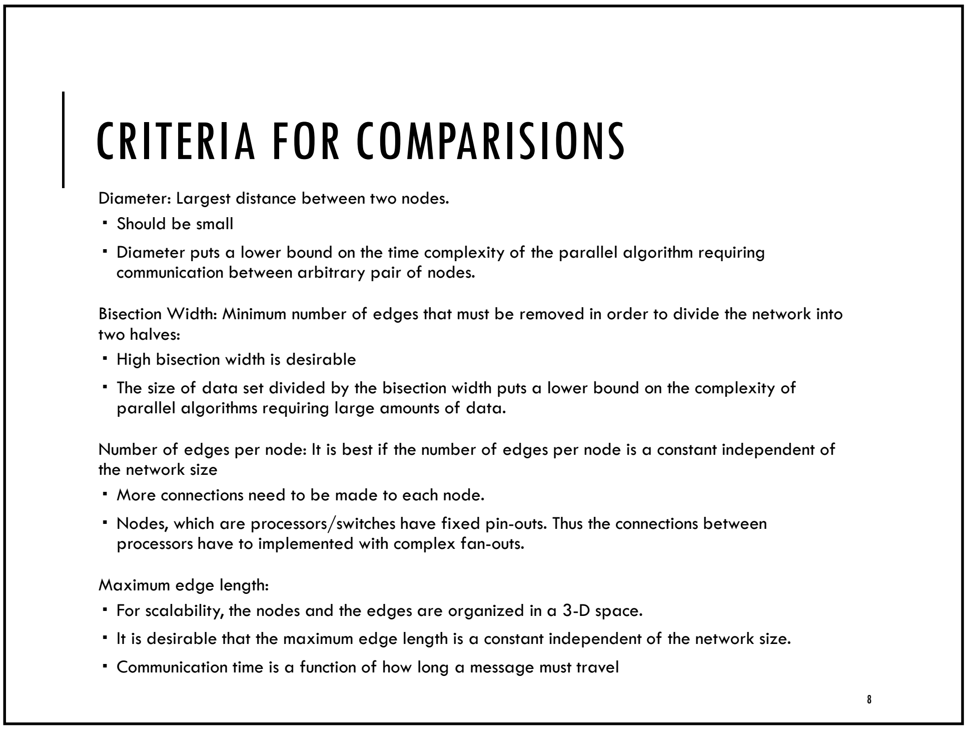
CRITERIA FOR COMPARISIONS

Diameter: Largest distance between any pair of nodes in the network.

Bisection Width: Minimum number of edges that must be removed in order to divide the network into two halves of equal size, or size differing by at most one node.

Number of edges per node (degree of the network topology is the maximum number of edges that are incident to a node in the topology).

Maximum edge length



CRITERIA FOR COMPARISIONS

Diameter: Largest distance between two nodes.

- Should be small
- Diameter puts a lower bound on the time complexity of the parallel algorithm requiring communication between arbitrary pair of nodes.

Bisection Width: Minimum number of edges that must be removed in order to divide the network into two halves:

- High bisection width is desirable
- The size of data set divided by the bisection width puts a lower bound on the complexity of parallel algorithms requiring large amounts of data.

Number of edges per node: It is best if the number of edges per node is a constant independent of the network size

- More connections need to be made to each node.
- Nodes, which are processors/switches have fixed pin-outs. Thus the connections between processors have to implemented with complex fan-outs.

Maximum edge length:

- For scalability, the nodes and the edges are organized in a 3-D space.
- It is desirable that the maximum edge length is a constant independent of the network size.
- Communication time is a function of how long a message must travel

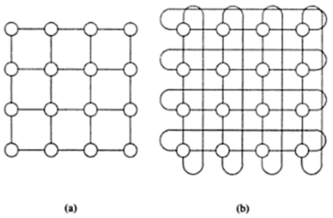
MESH NETWORKS

The nodes are arranged into a q-dimensional lattice.

Communication is only allowed between neighbouring nodes.

- Interior nodes communicate with 2q other processors.

Example: 2D mesh (a) No-wrap around, (b) with wrap around



CHARACTERISTICS OF THE MESH NETWORK

Let k be the number of processors in one dimension.

Diameter of a q-dimensional mesh with k^q nodes is q(k-1)

When k is even, Bisection width of the mesh is k^{q-1}

Maximum number of edges per node is 2q.

Maximum edge length is a constant, independent of the number of nodes for 2 and 3-D meshes.

BINARY TREE NETWORK

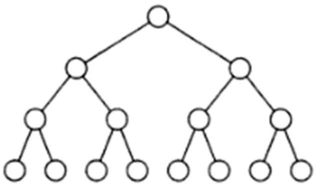
The 2^k-1 nodes are arranged into a complete binary tree of depth k-1

A node has at most three links: every node can communicate with its two children and and every node (other than the root) with its parent.

Low Diameter: 2(k-1)

Poor Bisection width: 1

As the number of nodes increase, the length of the longest edge increase.



Size=15
Depth = 3

11

HYPER TREE NETWORKS

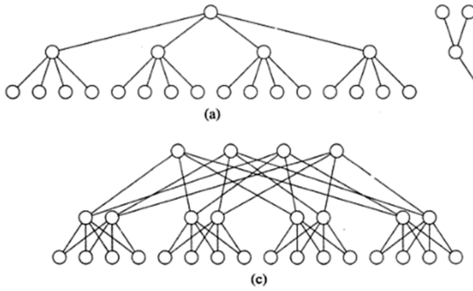
An approach to build a network with the low diameter of a binary tree but with an improved bisection width.

From “front” looks like *k*-ary tree of height *d*

From “side” looks like upside down binary tree of height *d*

12

HYPER TREE OF DEGREE 4 AND DEPTH 2



A 4-ary hypertree with depth d has 4^d leaves and $2^d(2^{d+1}-1)$ nodes in all.

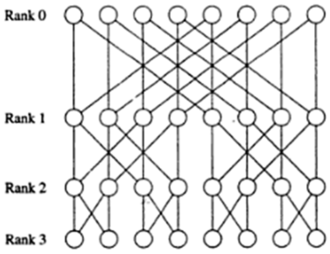
Diameter = 2d

Bisection width = 2^{d+1}

Number of edges per node is never more than 6.

Maximum edge length is an increasing function of the problem size.

BUTTERFLY NETWORK



Note if node (i,j) is connected to node (i-1,m), then node (i,m) is connected to node (i-1,j).

the entire network is made of such butterfly patterns.

Consists of $(k+1)2^k$ nodes divided into k+1 rows or ranks.

Each row contains $n=2^k$ nodes.

Node (i,j) refers to the jth node on the ith rank, $0 \leq i \leq k$, $0 \leq j \leq n$.

Node (i,j) on rank i>0 is connected to node (i-1,j) and node (i-1,m), where m is the integer found by inverting the ith most significant bit in the binary representation of j.

CHARACTERISTICS

As the rank number decrease, the widths of the wings of the butterflies increase exponentially.

- Thus the length of the longest network edge, increases with the number of network nodes.

Diameter with $(k+1)2^k$ nodes is 2k.

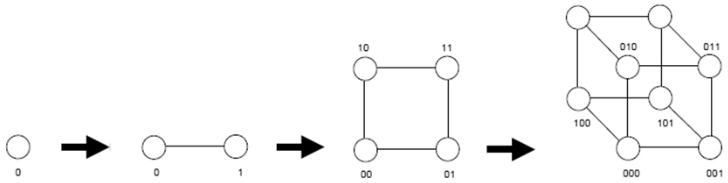
Bisection width = 2^k

HYPERCUBES

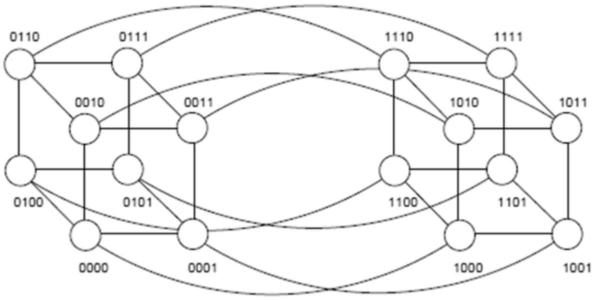
A binary n-cube or hypercube network is a network with 2^n nodes arranged as the vertices of a n-dimensional cube.

We can start thinking from a single point. Let its label be 0, and called as the 0-cube. We replicate the 0-cube, and place it one unit away. We label it as 1.

Likewise, we extend this as below:



AND FURTHER…



4-Cube: By extending the 3-cubes

PROPERTIES

The labels of two nodes differ by exactly one bit change if they are connected by an edge

In an k-dimensional hypercube, each node label is represented by k bits.

- Each of these bits can be inverted (0→1, 1→0), meaning there are exactly n incident edges.
- Degree of the k-dimensional hypercube is thus k.

DIAMETER

A k-bit integer can be transformed to another k-bit number by changing at most k bits (one bit at a time).

This corresponds to a walk across k edges in a hypercube.

BISECTION WIDTH OF A HYPERCUBE

Realize all nodes can be thought of lying on one of 2 planes:

- Consider the tth bit position. Depending on whether it is 0 or 1, the node is in either plane.
- To split the network into two sets of nodes, one in each plane, we have to remove edges which connects these two planes.
- Remember: The labels of two nodes differ by exactly one bit change if they are connected by an edge.
    - Thus every node in the 0-plane is connected to exactly one node in the 1-plane.
    - Thus, there are 2^{k-1}edges which connect these two planes (one edge for every pair of nodes).
- **Bisection width is thus 2^{k-1}**

NUMBER OF EDGES

Number of edges is $k.2^{k-1}$ (the proof is left as an exercise)

Comments:

- Bisection width is very high (half the number of nodes)
- Diameter is low
- Drawbacks:
  - Number of edges per node is a logarithmic function of the network size.
  - Maximum edge length increases as the network size increases.

INTERCONNECTION NETWORKS

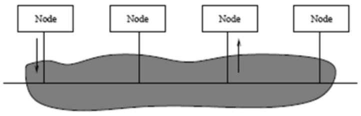
Interconnection network is a system of links that connects one or more devices to each other for the purpose of inter-device communication.

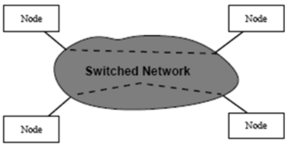
Usage:

- Connect processors to processors
- Allow multiple processors to access one or more shared memory modules
- Used to connect processors with locally attached memories to each other

Interconnection network can be of two types:

Shared Network: Can have at most one message on it at any time. Eg: A bus





Switched Network: Allows point to point messages among pairs of nodes and therefore supports the transfer of multiple concurrent messages.

Eg: Switched Ethernet.

INTERCONNECT NETWORK TOPOLOGIES

Notation: Squares to represent processors and/or memories, circles to represent switches

Direct Topology: There is exactly one switch for each processor node

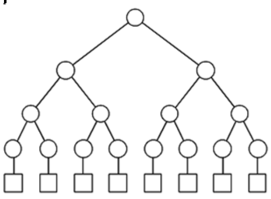
Indirect Topology: Number of switches is greater than the number of processor nodes

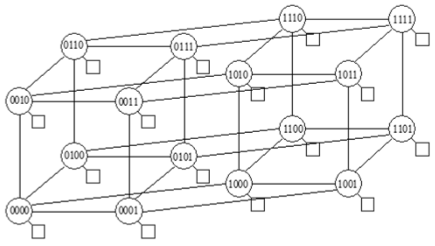
Certain topologies are direct, while others are indirect:

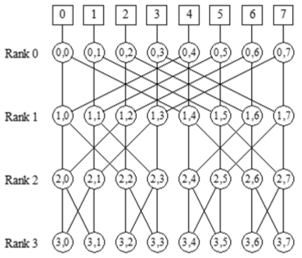
- The 2D mesh is almost always used as a direct topology
- Binary trees are always indirect topologies
- Butterfly networks are indirect topologies: processors are connected to rank 0, and either memory modules or switches back to the processors are connected to the last rank.
- Hypercubes are direct topologies

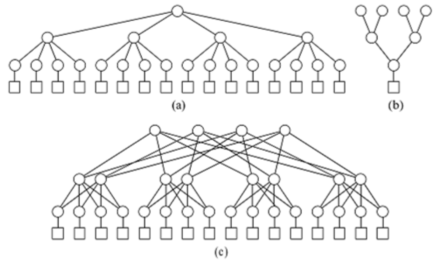
23

INTERCONNECT NETWORK TOPOLOGIES









24

VECTOR PROCESSORS AND PROCESSOR ARRAYS

Vector computer is a computer that has an instruction that can operate on a vector.

A pipelined vector processor is a vector processor that can issue a vector instruction that operates on all the elements of the vector in parallel by sending these elements through highly pipelined functional units with a fast clock.

A processor array is a vector processor that achieves parallelism by having a collection of identical synchronized processing elements, each of which executes the same instruction on different data, and which are controlled by a single control unit.

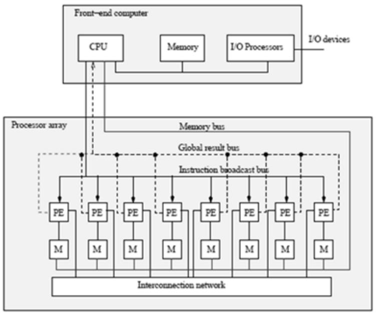
PROCESSOR ARRAY

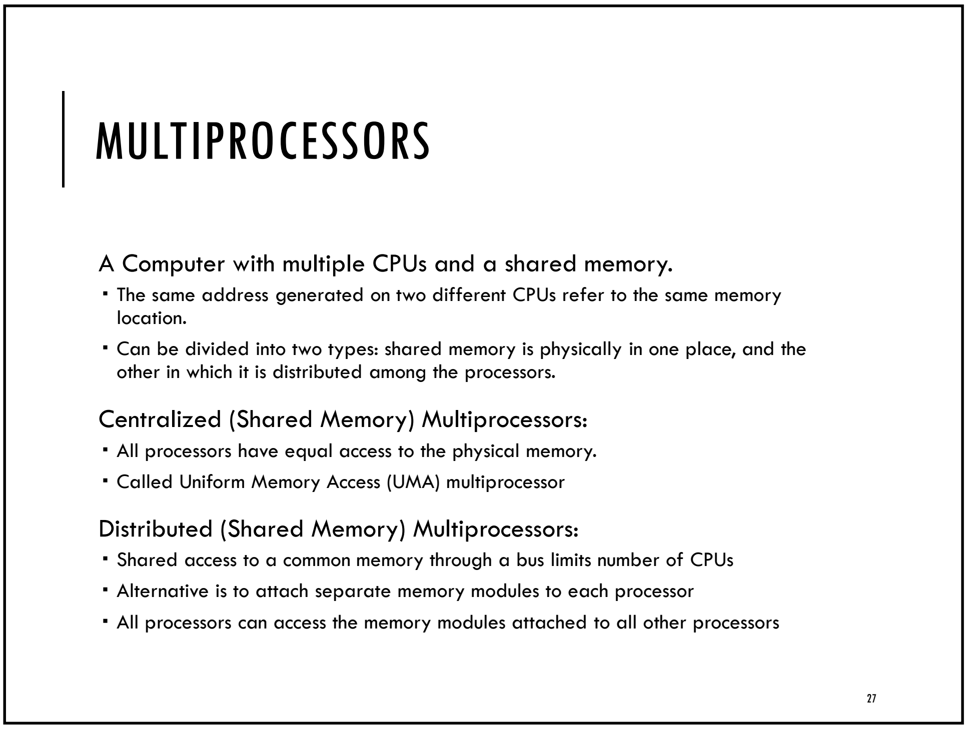
Each PE has a unique identifier, its processor id.

Each PE has a small local memory in which its own private memory can be stored.

- The data on which each PE operates is distributed along the PE's local memories at t start of the computation (provided it fits in!)
- The control unit, broadcasts the instruction to executed to the PEs.
- The PEs execute it on the data in the local memory, and can store the results in the local memories, or can return the result back to the CPU.
- The PEs are connected to each other through interconnection network, that allows them to interact data between each other as well.

A global result line is usually a separate, parallel bus that allows each PE to transmit values back to the CPU to be combined by a parallel, global operation, such as logical-and, logical-or, depending on the hardware support in the CPU.





MULTIPROCESSORS

A Computer with multiple CPUs and a shared memory.

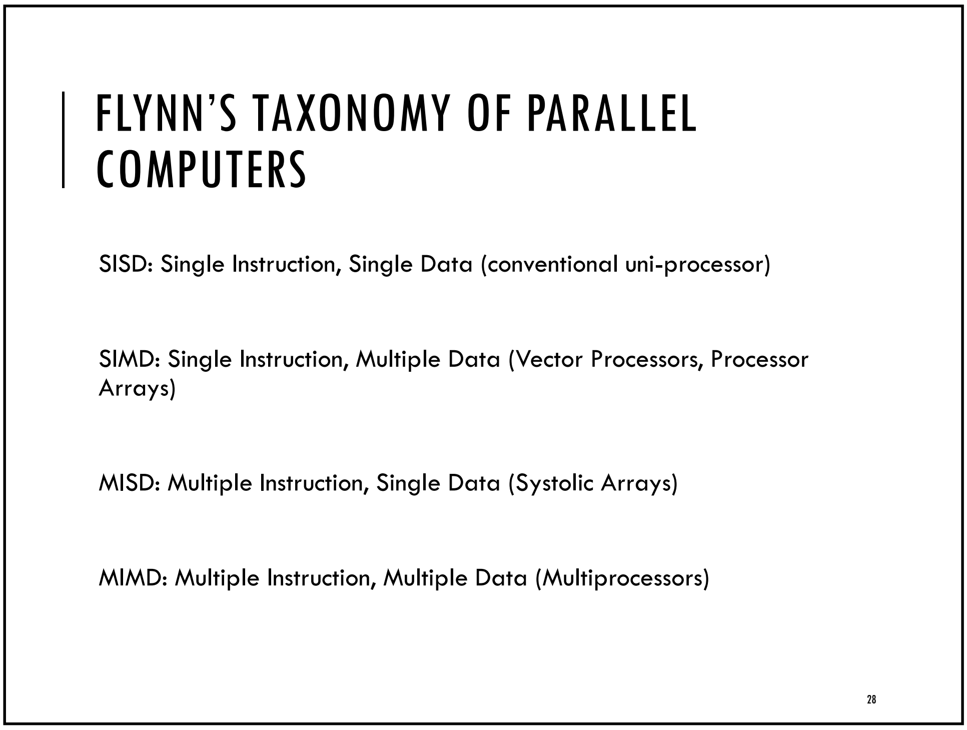
- The same address generated on two different CPUs refer to the same memory location.
- Can be divided into two types: shared memory is physically in one place, and the other in which it is distributed among the processors.

Centralized (Shared Memory) Multiprocessors:

- All processors have equal access to the physical memory.
- Called Uniform Memory Access (UMA) multiprocessor

Distributed (Shared Memory) Multiprocessors:

- Shared access to a common memory through a bus limits number of CPUs
- Alternative is to attach separate memory modules to each processor
- All processors can access the memory modules attached to all other processors



FLYNN’S TAXONOMY OF PARALLEL COMPUTERS

SISD: Single Instruction, Single Data (conventional uni-processor)

SIMD: Single Instruction, Multiple Data (Vector Processors, Processor Arrays)

MISD: Multiple Instruction, Single Data (Systolic Arrays)

MIMD: Multiple Instruction, Multiple Data (Multiprocessors)