Interconnect Network Topologies

Characteristics of a network

- Topology (what)
  - Physical interconnection structure of the network graph.
  - Physically limits the performance of the networks.
- Routing algorithm (which)
  - Restricts the set of paths that messages can follow.
- Switching strategy (how)
  - How data in a message traverses a route (passing routers)
- Flow control mechanism (when)
  - When a message or portions of it traverse a route
  - What happens when traffic encountered

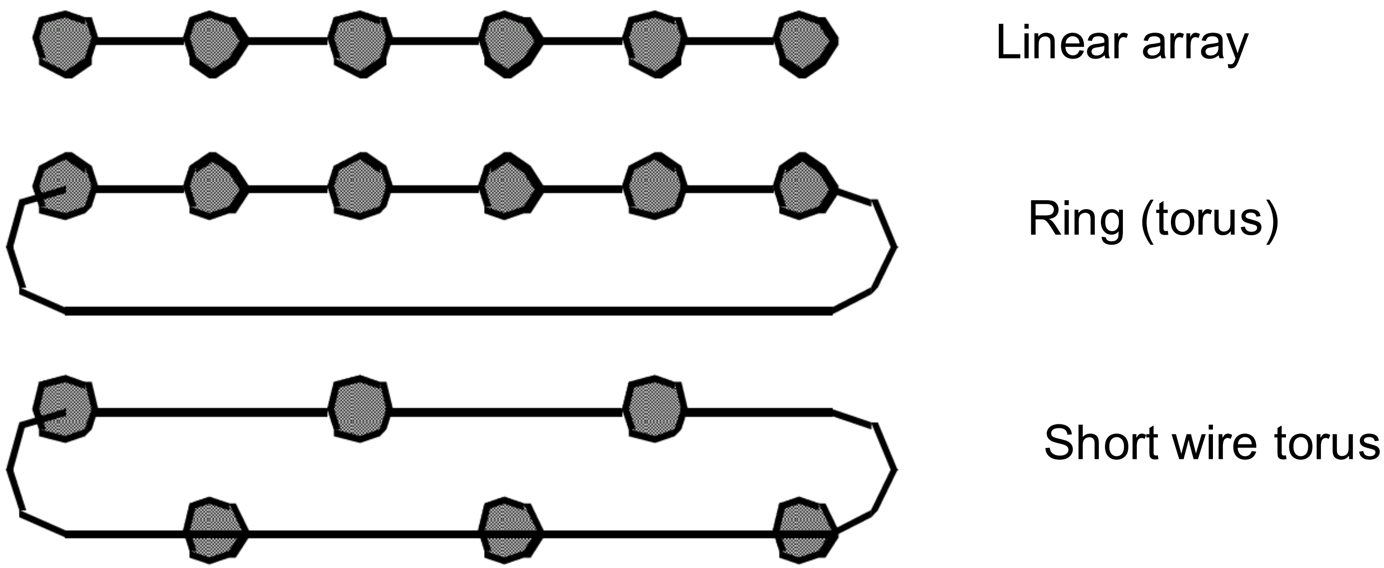
Topology

- How the components are connected.
- Important properties
  - Diameter: maximum distance between any two nodes in the network (hop count, or # of links).
  - Nodal degree: how many links connect to each node.
  - Bisection bandwidth: The smallest bandwidth between half of the nodes to another half of the nodes.
- A good topology: small diameter, small nodal degree, large bisection bandwidth.

Topology

- Regular topologies
  - Nodes are connected with some kind of patterns.
    - The graph has a structure.
  - Nodes are identified by coordinates.
  - Routing can usually pre-determined by the coordinates of the nodes.
- Irregular topologies
  - Nodes are connected arbitrarily.
    - The graph does not have a structure, e.g. internet
    - More extensible in comparison to regular topology.
  - Usually use variations of shortest path routing.

Linear Arrays and Rings

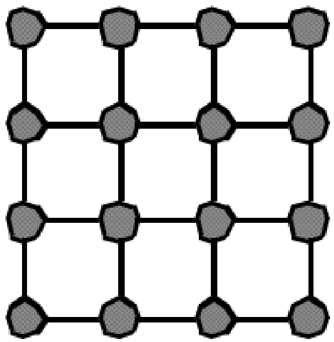


Diameter = ?, nodal = ? Bisection bandwidth = ?

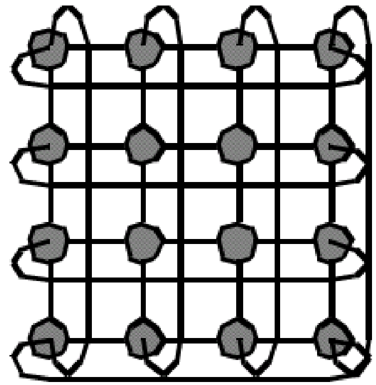
Describing linear array and ring

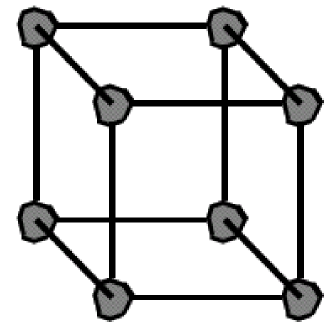
- Array: nodes are numbered from 0, 1, …, N-1
  - Node i is connected to node i+1, $0<=i<=N-2$
- Ring: nodes are numbered from 0, 1, …, N-1
  - Node i is connected to node (i+1) mod N, for all $0<=i<=N-1$

Multidimensional Meshes and Tori



2D Grid





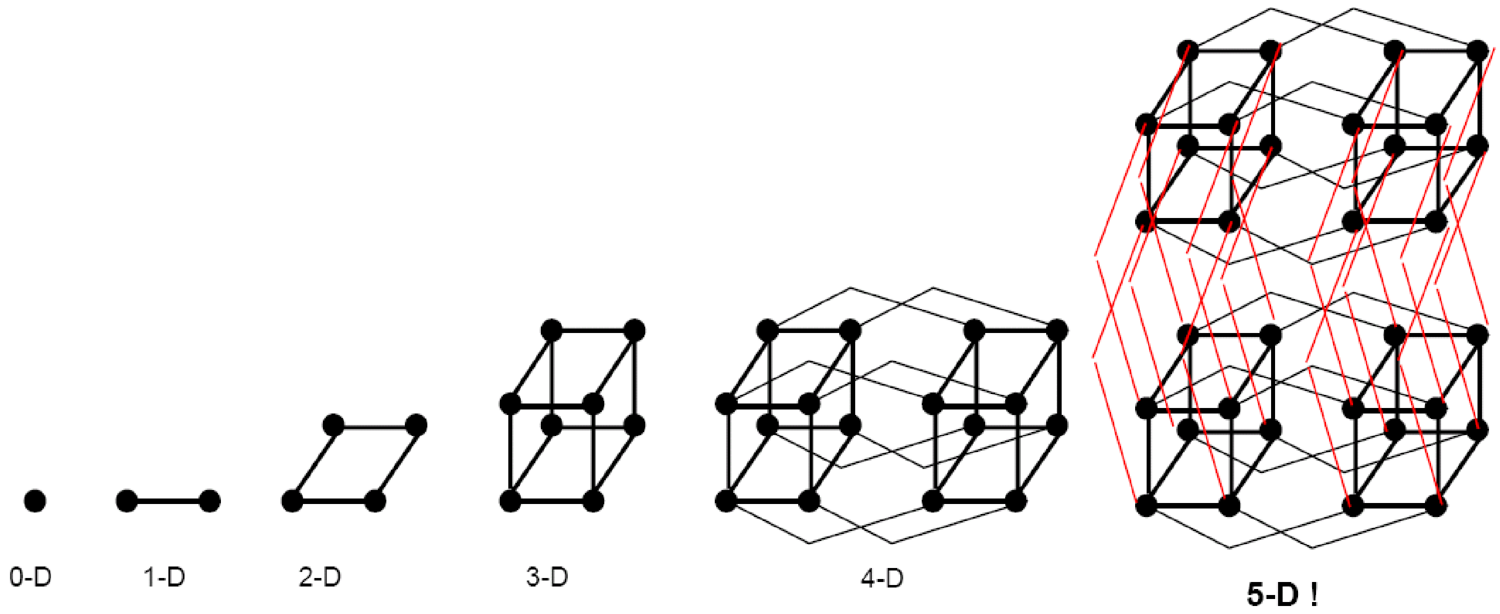
3D Cube

- d-dimensional array/torus
  - N = k_{d-1} x k_{d-2} x … x k_0
  - Each node is described by a d-vector of coordinate
  - Node (i_{d-1} x i_{d-2} x …x d_0) is connected to ???

More about multi-dimensional mesh and tori

- d-dimension k-ary mesh (torus)
  - Each node is described by a d-vector of coordinates.
    - The value of each item in the vector is between 0 and d_i-1.
  - Diameter = ?
  - Nodal degree = ?
  - Bisection bandwidth = ?

Hypercubes

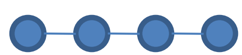


- Also called binary n-cubes. # of nodes = N = 2^n
- Each node is described by its binary representation.
  - There is a link between two nodes whose binary representations differ by one bit.
- Diameter=? Nodal degree = ? Bisection bandwidth = ?

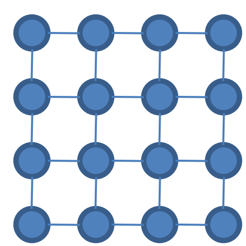
K-ary n-cube (n-dimensional, k-ary mesh/torus)

- Extended from binary (hypercube) to k-ary
- Each dimension has k elements, n dimensions
- Each node is identified by a k-based number (n digits).
  - Dimension order routing

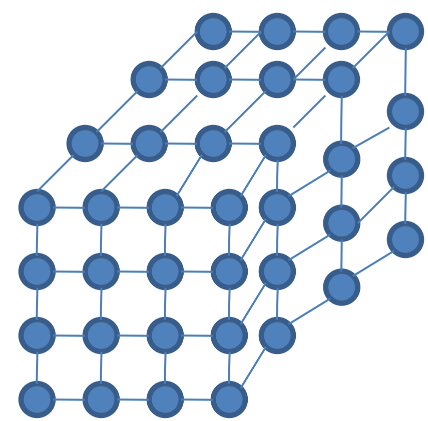
4-ary 0-cube



4-ary 1-cube

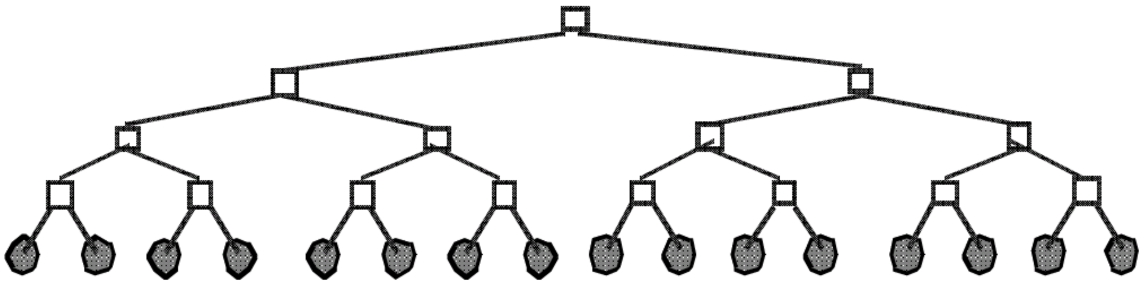


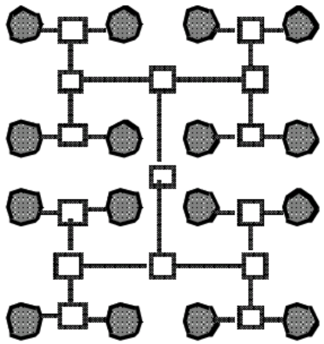
4-ary 2-cube



4-ary 3-cube

Trees





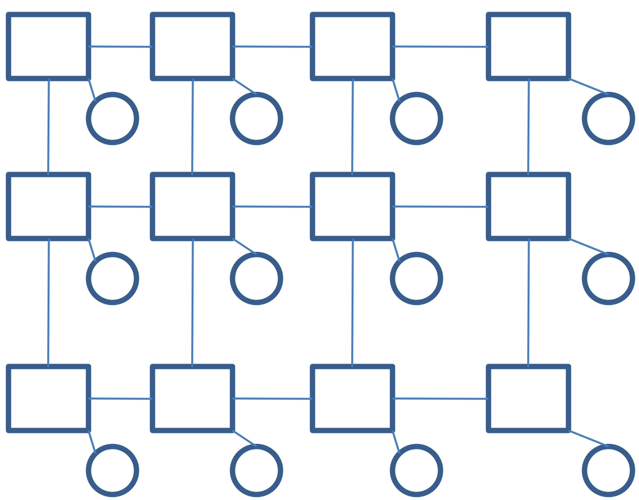
- Fixed degree, log(N) diameter, O(1) bisection bandwidth.
- Routing: up to the common ancestor than go down.

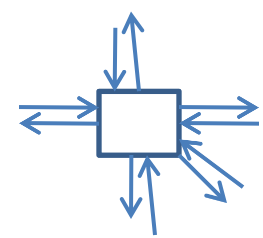
Irregular topology

- Irregular topology does not any special mathmetic properties
  - Can be expanded in any way.
  - No easy way for routing: routes need to be computed like in the Internet.
    - Routes can usually be determined in a regular network by using the coordinates of the source and destination.

Direct and indirect networks

- All the previously discussed networks are direct networks in that the compute nodes are directly attached to the nodes in the topology.
  - An example mesh system.



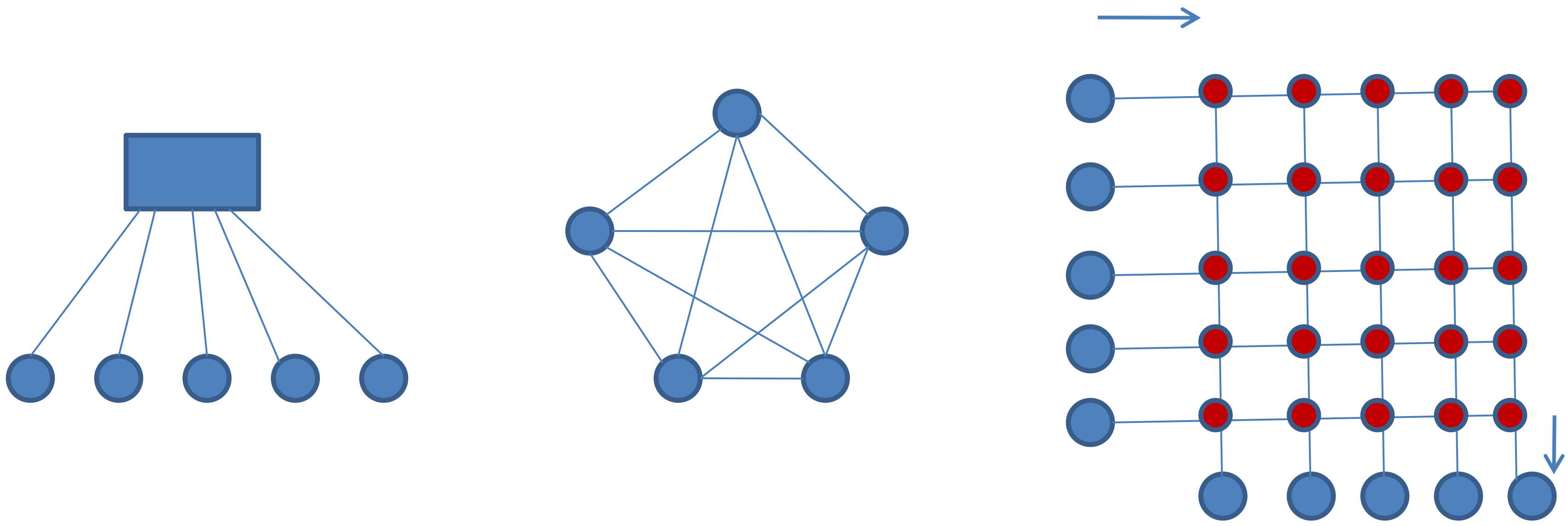


Each switch is a 5x5 switch

Indirect networks

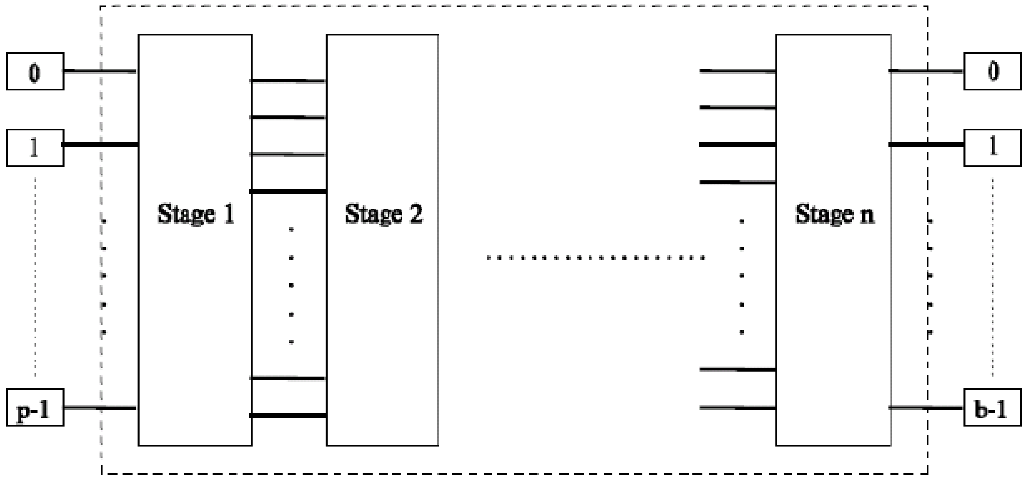
- Compute nodes are not directly attached to each switch, but are rather attached to the whole network.
  - Using a central interconnect to connect all compute nodes
  - The network emulate the cross-bar switch functionality.

Fully connected network



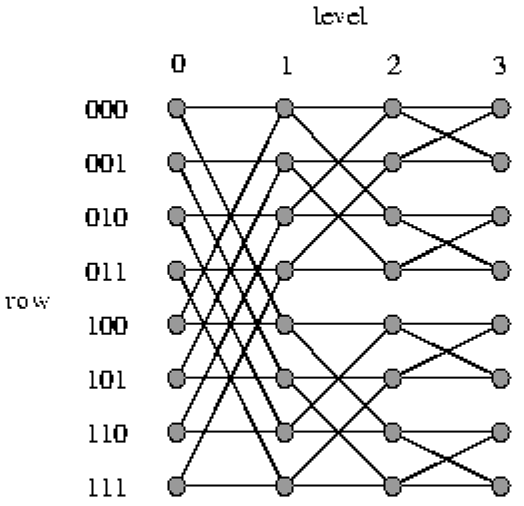
- Different organizations:
  - Connected by one switch (crossbar switch), connecting all nodes, connected with a crossbar.
- All permutation communication (each node sends one message and receives one message) can be realized.

Multistage interconnection networks (MIN)

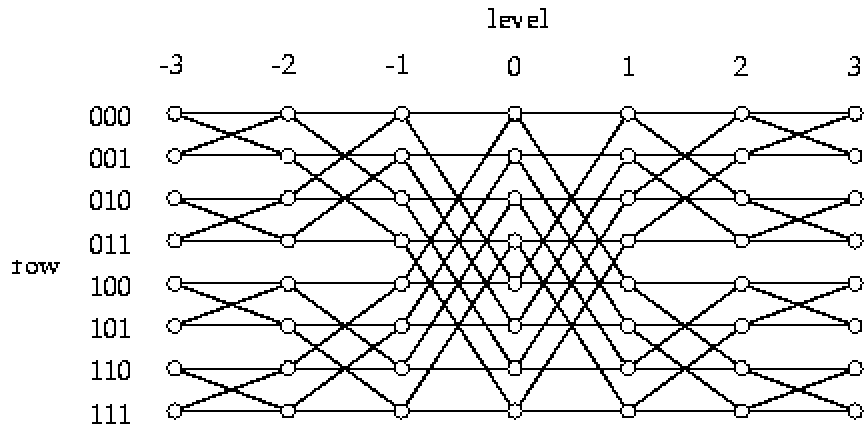


- Try to emulate the cross-bar connection.
  - Realizing permutation without blocking
  - Using smaller cross-bar(2x2, 4x4) switches as the building block. Usually O(Nlg(N)) switches (lg(N) stages.

Multi-stage networks examples



(a) An 8-input butterfly network

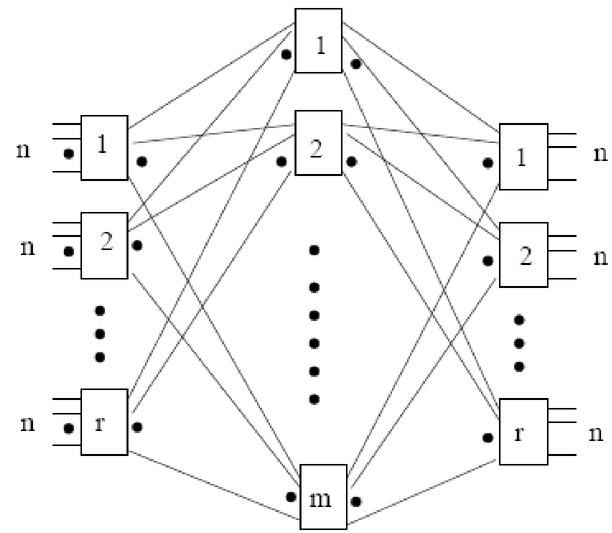


(b) An 8-input Benes network

- MINs can be blocking or non-blocking
  - Blocking: there exist some permutation that results in link contention.
  - Non-blocking: any permutation can be realized without link contention
- Butterfly network is blocking.
- Benes network is non-blocking.

Clos Network

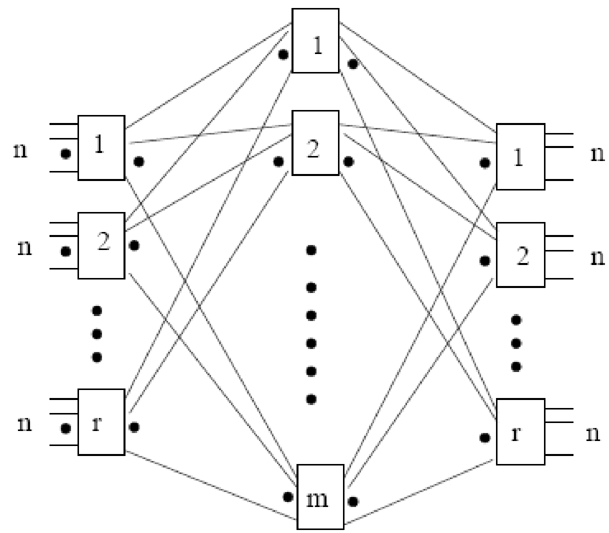
- Three stages: ingress stage, middle stage, and egress stage
  - Ingress/egress stage has r n X m switches
  - Middle stage has m r X r switches
  - Each switch at ingress/egress stage connects to all m middle switches (one port to each switch).



(a) Clos(n, m, r)

Clos Network

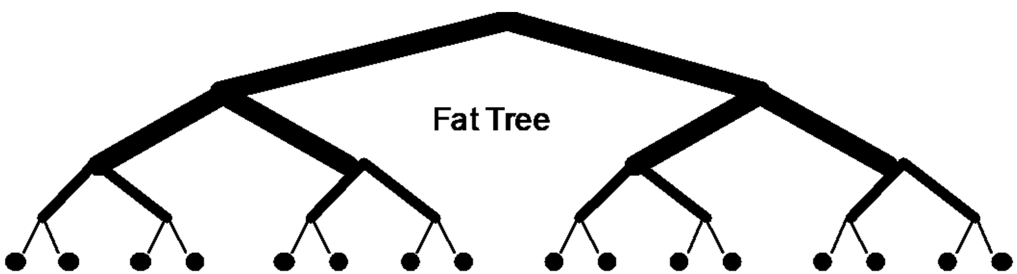
- Clos network is non-blocking when m>=2n-1.

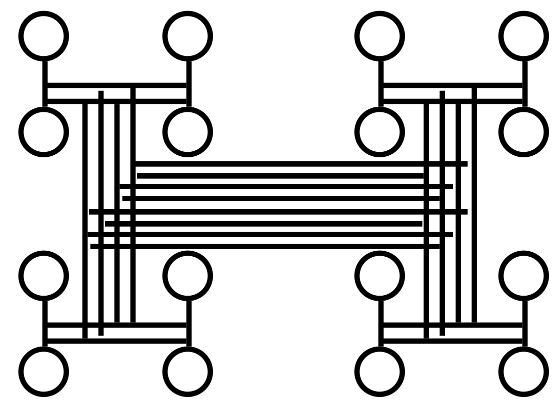


(a) Clos(n, m, r)

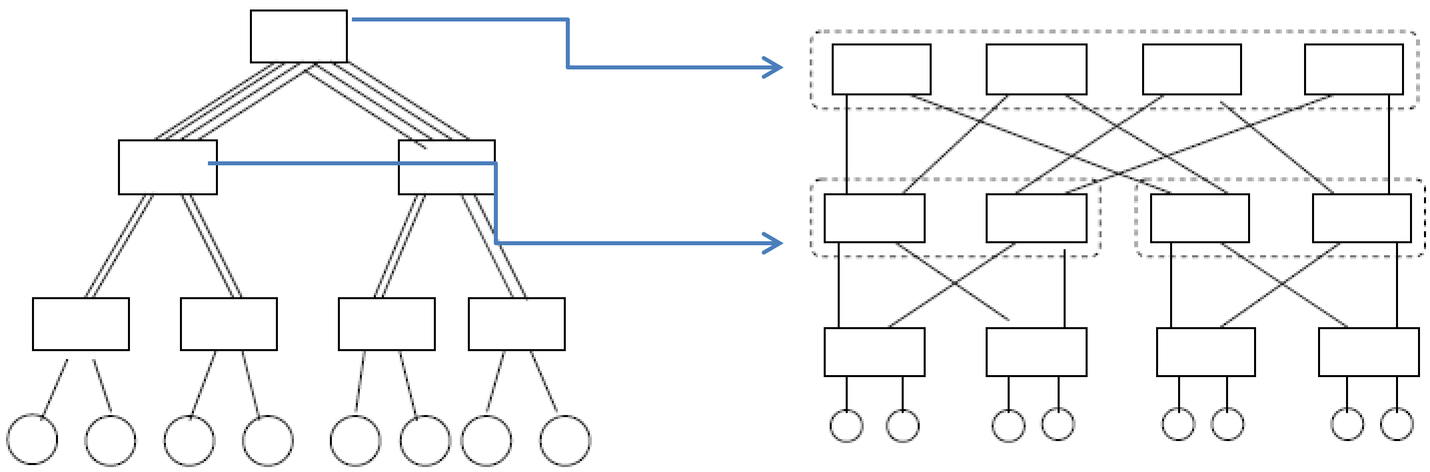
Fat-Trees

- Fatter links (really more of them) as you go up, so bisection BW scales with N
  - Not practical, root is an NxN switch



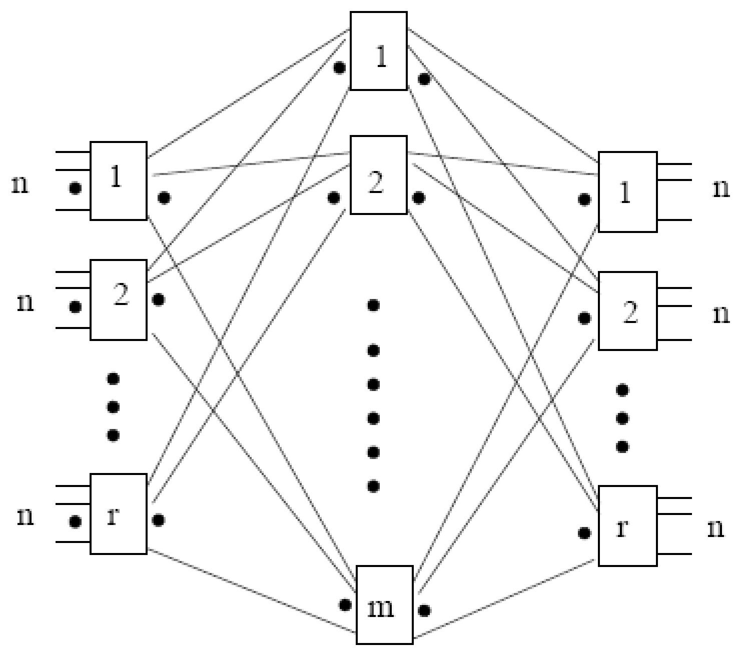


Practical Fat-trees

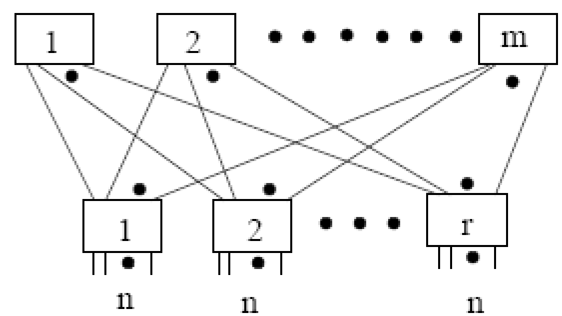


- Use smaller switches to approximate large switches.
  - Connectivity is reduced, but the topology is not implementable
  - Most commodity large clusters use this topology. Also call constant bisection bandwidth network (CBB)

Clos network and fat-tree (folded Clos)



A generic 3-stage Clos network



A generic 2-level fat-tree (folded Clos)

Physical constraint on topologies

- Number of dimensions.
  - 2 or 3 dimensions
    - Can layout physically
    - Short wires, easy to build
    - Many hops, low bisection bandwidth
  - >=4 dimensions
    - Harder to build, longer wires
    - Fewer hops, better bisection bandwidth
  - K-ary n-cubes provide a good framework for comparison.