Interconnection networks

When more than one processor needs to access a memory structure, interconnection networks are needed to route data—

- from processors to memories (concurrent access to a shared memory structure), or
- from one PE (processor + memory) to another (to provide a message-passing facility).

Inevitably, a large bandwidth is required to match the combined bandwidth of the processing elements.

» One extreme is a *shared bus*. How does the cost scale as the number of processors N increases?

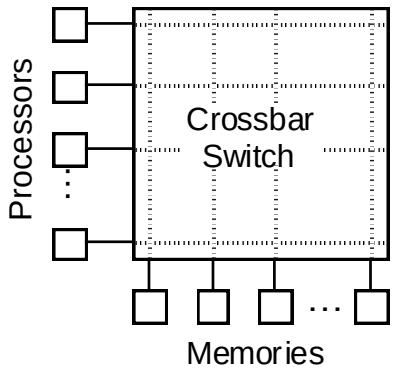
How does the bandwidth scale?

» For concurrent access to shared memory, the ideal structure is a _____, which can simultaneously connect any set of processors to any set of distinct memory modules.

All N processors can access all M memory units with an $N \times M$ crossbar switch.

Since there are usually about as many processors as memories, as processors are added, the complexity of a crossbar switch grows as N^2.

How does the bandwidth scale?



For reasonably large values of N, the crossbar switch may be more expensive than the processors and memories.

» For message passing, the most general is the *complete interconnection network*—a path from each processor to every other processor.

Unfortunately, this requires _____ bidirectional links. Cost grows with the square of N.

Measures of interconnection performance

Several metrics are commonly used to describe the performance of interconnection networks:

- *Degree*, the number of links ("edges") that are attached to a node.
- *Diameter*, the maximum number of nodes through which a message must pass on its way from source to destination.

  Diameter measures the maximum delay in transmitting a message from one processor to another.
- *Average distance*, where the distance between two nodes is defined by the number of hops in the shortest path between those nodes. Average distance is given by

$$d_{avg} = \frac{\sum_{d=1}^{r} (d \cdot N_d)}{N-1}$$

  where N is the number of nodes, N_d is the number of nodes at distance d apart, and r is the diameter.
- *Bisection width*, the smallest number of wires you have to cut to disconnect the network into two equal halves (±1).

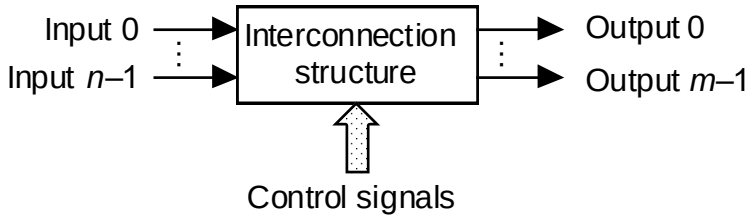
For a crossbar, give all of these metrics: Degree, diameter, average distance, bisection width.

Which of these metrics are measures of performance, and which are measures of cost?

Interconnection topologies

[§10.4] An idealized interconnection structure—

- takes a set of n input ports labeled 0, … , n–1 and
- sets up connections between them and a set of m output ports 0, … , m–1,
- with the connections determined by control signals.



Usually we will assume that $m = n$.

Here are some sample topologies.

1. Ring.

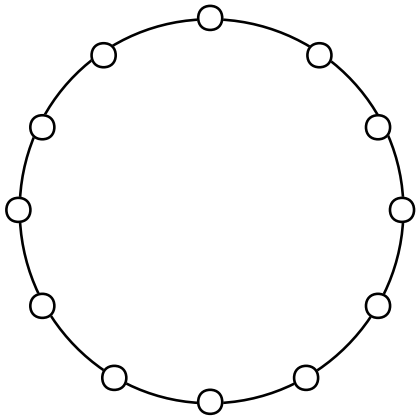
Processor i directly connected to processors i+1(mod N) and i–1 (mod N). Data can be moved from any processor to any other by a sequence of cyclic shifts.

Motivation: Many parallel algorithms include calculations of the form

$$X[i] := \frac{X[i-1] + X[i]}{2}$$

Usually every item of an array except the first and last is updated in this way.

The processor interconnections can be diagrammed as a bidirectional ring:



The diameter of a bidirectional ring is ____. Its bisection width is

What about average distance and degree?

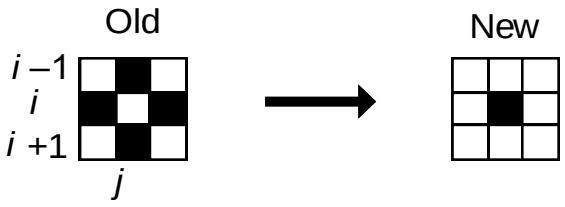
2. Mesh interconnection network

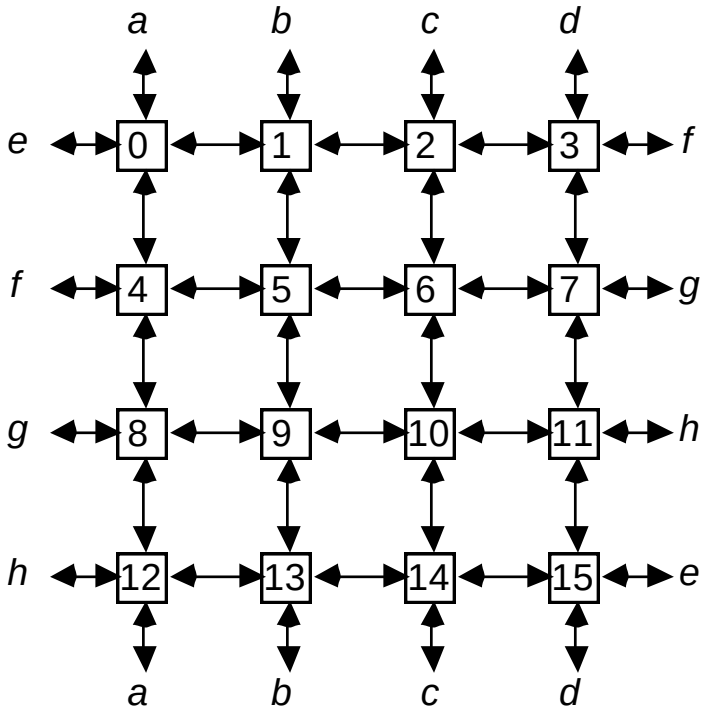
A mesh is like having “row & column” cyclic shifts.

One motivation: *Four-point iteration* is common in the solution of partial differential equations. Calculations of the form

$$X[i,\ j]\ := (X[i+1, j] + X[i-1, j] + X[i\ , j-1] + X[i\ , j+1]) \div 4)$$

are performed frequently.





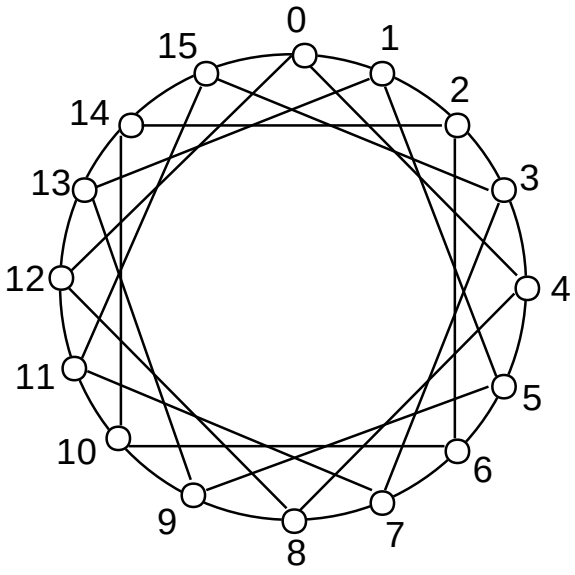
Here is an example of a 16-node mesh. Note that the last element in one row is connected to the first element in the next.

If the last element in each row were connected to the first element in the same row, we would have a *torus* instead.

In the Illiac IV, each processor *i* was connected to processors:

$$\{i+1,\ i-1,\ i+8, \text{ and } i-8\} \pmod{64}.$$

The diameter of an Illiac IV mesh is $\sqrt{N} - 1$. For example, in a 16-node mesh structure, it takes a maximum of 3 steps. To see that, let us look at the mesh interconnection network shown in the form of a *chordal ring:*



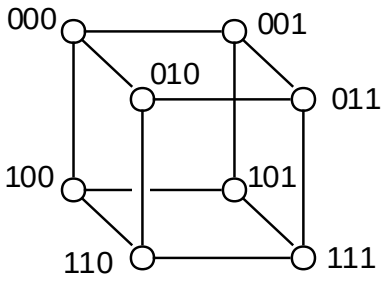
In a 64-element mesh, any node can be reached from any other in no more than 7 of these shifts.

Without the end-around connections (a “pure” 2D mesh), the diameter is $2(\sqrt{N} - 1)$.

It is also possible to have a multidimensional mesh. The diameter of a d-dimensional mesh is $d(N^{1/d})-1$ and its bisection width is $N^{(d-1)/d}$

The average distance is $d \times 2(N^{1/d})/3$ (without end-around connections).

3. Hypercube



A hypercube is a generalized cube. In a hypercube, there are 2^n nodes, for some n. Each node is connected to all other nodes whose numbers differ from it in only one bit position.

What is the degree of a hypercube?

What is the diameter of a hypercube?

What is the average distance?

What is the bisection width?

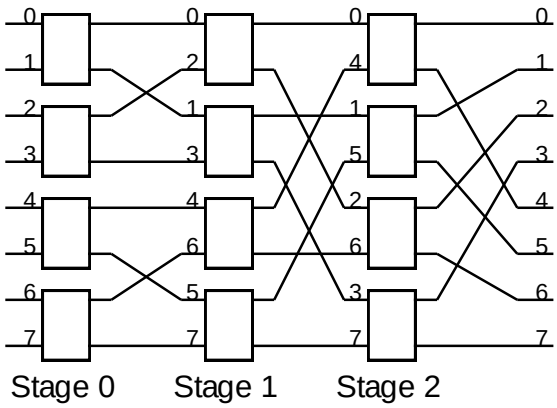
An interconnection network can be either single stage or *multistage.*

- If it is single stage, then the individual control boxes must be set up to *n* times to get data from one node to another.

  Data may have to pass through several PEs to reach its destination.

- *Multistage* networks have several sets of switches in parallel, so data only needs to pass through several *switches*, not several nodes.

For a multistage cube network, we can diagram the paths from one cell to another like this:



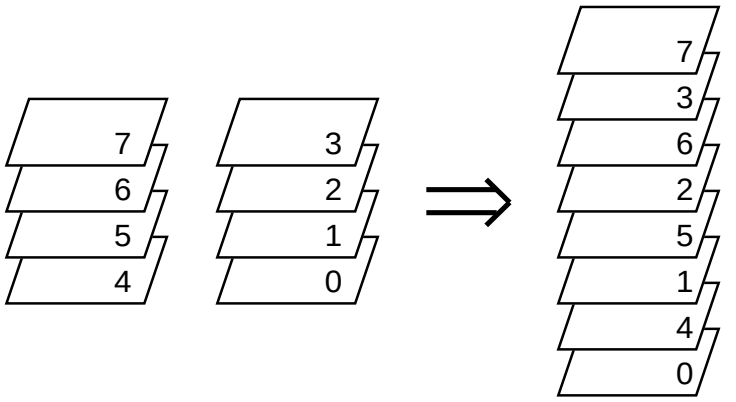
A multistage cube network is often called an *indirect binary n-cube.*

4. Perfect-shuffle interconnection

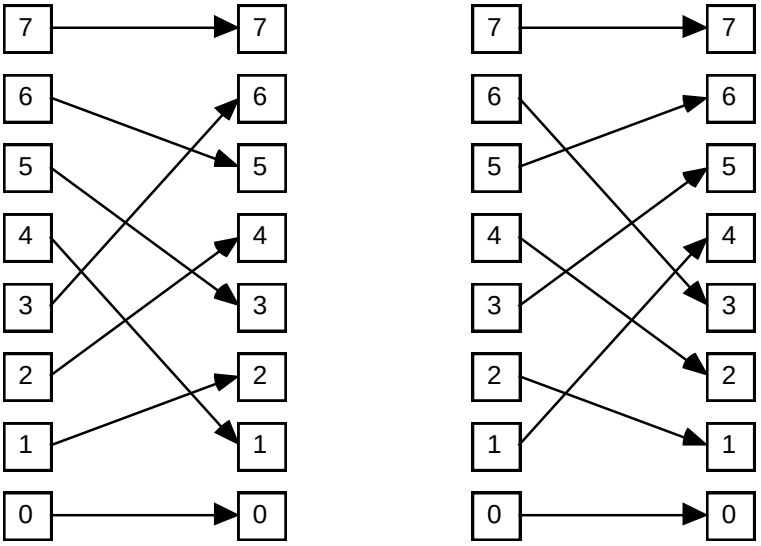
This interconnection network is defined by the routing function

$$S\left((a_{n-1} \ \ldots \ a_1 a_0)_2\right) \equiv (a_{n-2} \ \ldots \ a_1 a_0 \, a_{n-1})_2$$

It describes what happens when we divide a card deck of, e.g., 8 cards into two halves and shuffle them “perfectly.”



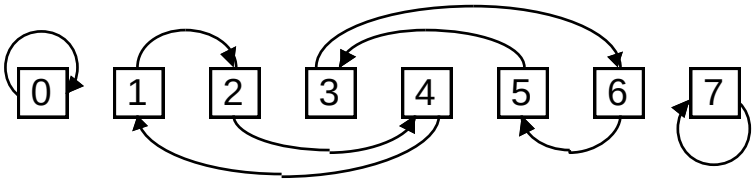
We can draw the processor interconnections required to obtain this transformation (at near right):



If the links are bidirectional, the *inverse perfect shuffle* is obtained (above, right).

5. Shuffle-exchange network

By itself, a shuffle network is not a complete interconnection network. This can be seen by looking at what happens as data is *recirculated* through the network:

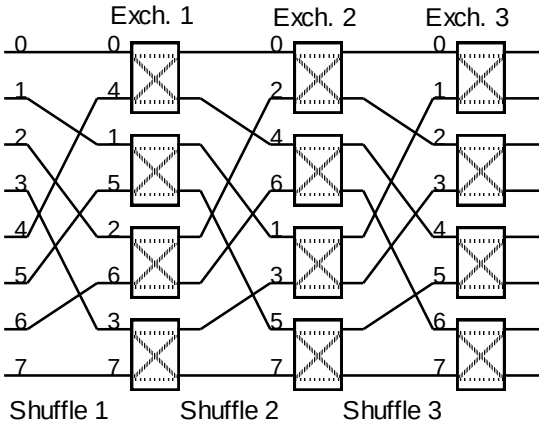


An *exchange* permutation can be added to a shuffle network to make it into a complete interconnection structure:

$$E\,(a_{n-1}\ \ldots\ a_1\,a_0)_2\ +\ a_{n-1}\ \ldots\ a_1\,a_0$$

A shuffle-exchange network is isomorphic to a cube network, with a suitable renumbering of boxes.

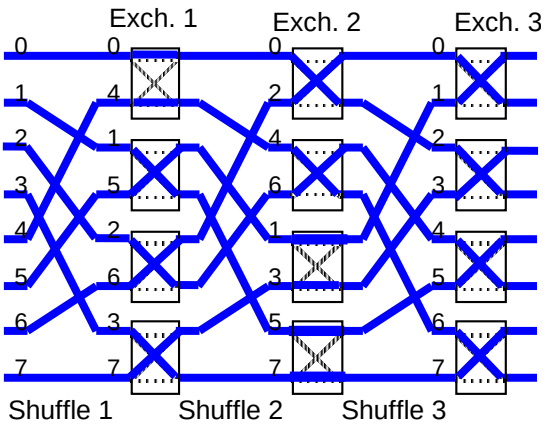
Here is a diagram of a multistage shuffle-exchange network for $N = 8$.



Sums (or other operations involving all the elements) can be performed in log N steps.

In addition, with a shuffle-exchange network, arbitrary cyclic shifts of an N-element array can be performed in log N steps.

This diagram shows how the switches in a shuffle-exchange network can be set to route input *k* to output *k* + 3 (mod 8).



Switches are set to pass through or cross over depending on the exclusive-or of the input and output port numbers.

0 **xor** 3 = 000_2 **xor** 011_2 = 011 ➔ the first switch is set to pass through; the next two along the route are set to cross over.

1 **xor** 4 = 001_2 **xor** 100_2 = 101 ➔ the first switch is set to cross over, the next one to pass through, and the last one to cross over.

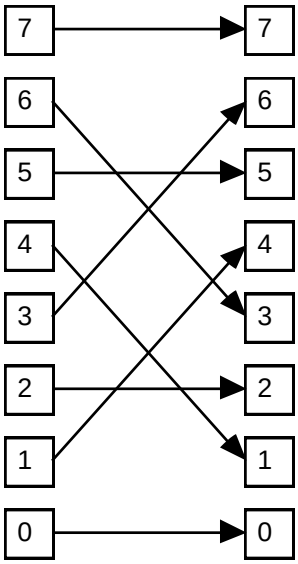
2 **xor** 5 = 010_2 **xor** 101_2 = 111 ➔ all three switches along the route are set to cross over.

The diameter of a shuffle-exchange network is

The bisection width is

6. Butterfly network

A butterfly network is closely related to shuffle-exchange networks.



The butterfly permutation is defined as—

$$B(a_{n-1}\ a_{n-2}\ \ldots\ a_1\ a_0)\ \equiv\ a_0\ a_{n-2}\ \ldots\ a_1\ a_{n-1}$$

i.e., the permutation formed by interchanging the most- and least-significant bits in the binary representation of the node number.

This permutation can be diagrammed as shown at the right:

Two variants of the butterfly permutation are the kth sub-butterfly, performed by interchanging bits 0 and k–1 in the binary representation—

$$B_k(a_{n-1}\ a_{n-2}\ \ldots\ a_1\ a_0)\ \equiv\ a_{n-1}a_{n-2}\ a_k\ a_0\ldots\ a_1\ a_{k-1}$$

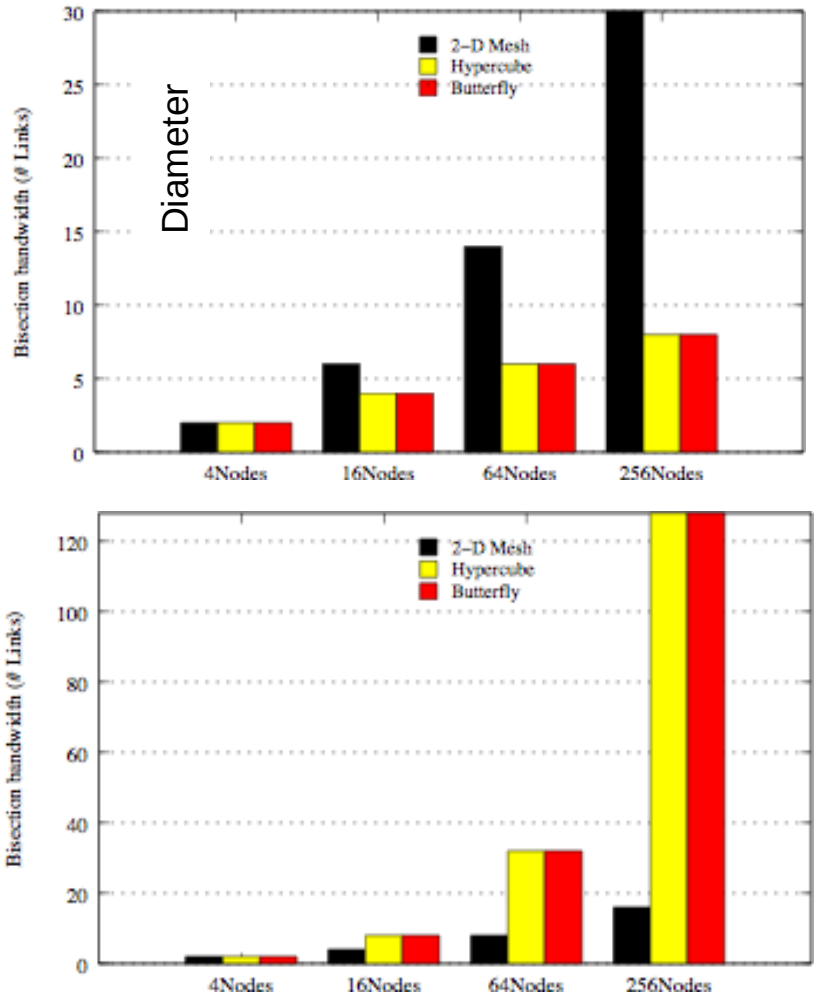
and the kth super-butterfly, peformed by interchanging bits n–1 and k–1:

$$B^k(a_{n-1}\ a_{n-2}\ \ldots\ a_1\ a_0) \equiv a_{k-1}\ a_{n-2}\ a_k\ a_{n-1}a_{k-2}\ \ldots\ a_0$$

The textbook has an interesting diagram showing how metrics change with size for 2D meshes, hypercubes, and butterflies.

Explain what it says about increasing arity (k) vs. increasing dimension (d). Given the numbers here, which network would be more desirable for larger multiprocessors?

But why is this not the whole story?



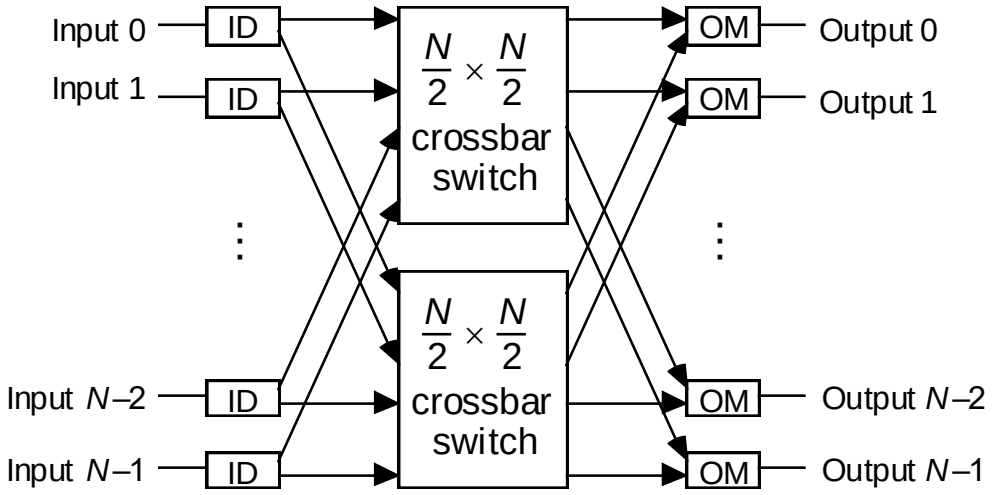
7. Beneš network

As we have seen, a crossbar switch is capable of connecting a set of inputs to *any* set of distinct outputs simultaneously.

A shuffle-exchange, or multistage cube, network is not capable of doing this. (It is easy to come up with an example.)

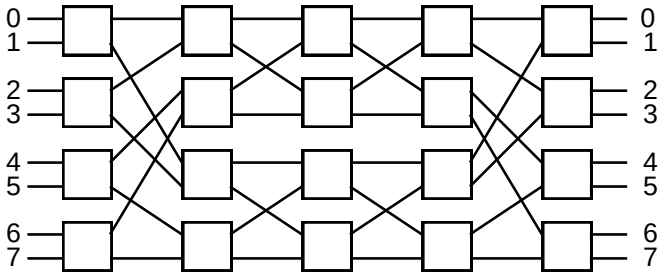
Is it possible to achieve an *arbitrary* permutation of input-output combinations with less than a full crossbar switch?

Yes. The Beneš network substitutes two $N/2 \times N/2$ crossbar switches, plus an N-input exchange switch for a full crossbar switch, as shown below.



The resulting $N/2 \times N/2$ crossbar switches can be similarly reduced.

Through this process, a full connection network can be produced from 2×2 switches at significantly lower cost than a full crossbar:



The stages of a Beneš network are connected by shuffle and inverse-shuffle permutations.

The network is called *rearrangeable*, since the switch settings can always be rearranged to accommodate any input-output mapping.

In some Beneš networks, the switches are capable of performing broadcasts, as well as pass-through or interchange.

Such Beneš networks can achieve all N^N possible input/output mappings.

Trees

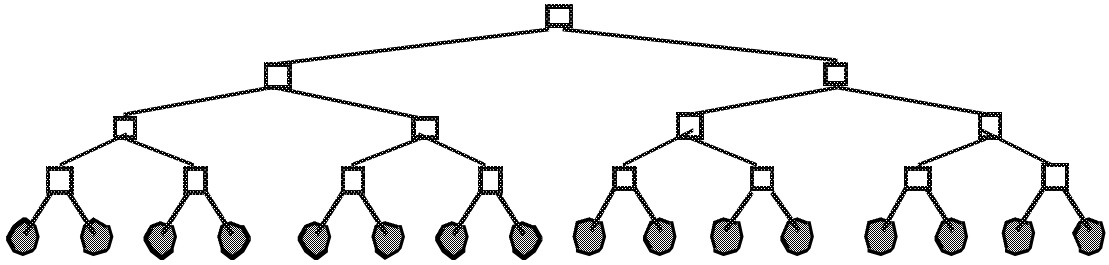
In meshes and hypercubes, the average distance increases with the dth root of N.

In a tree, the average distance grows only logarithmically.

A simple tree structure, however, suffers from two problems.

- Congestion

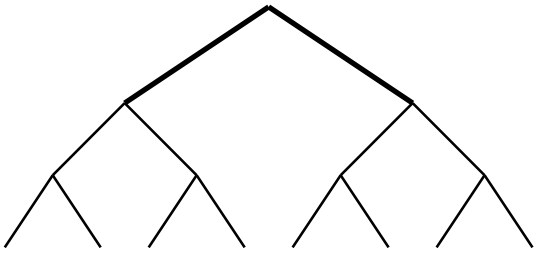
- Its fault tolerance is low.



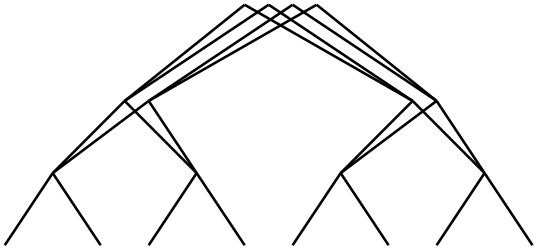
8. Fat trees

One approach to overcoming the limitations of the tree topology was devised by Leiserson and implemented in the Thinking Machines CM-5 data network.

The idea is that the edges at level *k* should have two or more times the capacity of the edges at level *k*+1 (the root is at level 0).



In reality, the links at higher levels are formed by replicating connections.



The algorithm for routing a message from processor *i* to processor *j* is as follows:

- Starting from processor *i*, a message moves up the tree along the path taking it to the first common ancestor of *i* and *j*.
- There are many possible paths, so at each level the routing processor chooses a path at random, in order to balance the load.
- Upon reaching the first common ancestor, the message is then routed down along the unique path connecting it to processor *j*.

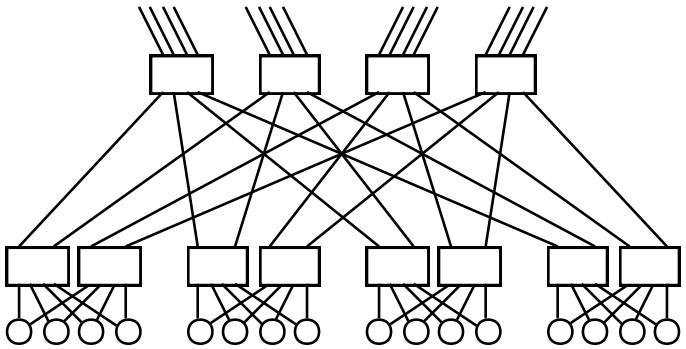
What are some metrics for a fat tree?

The diameter is

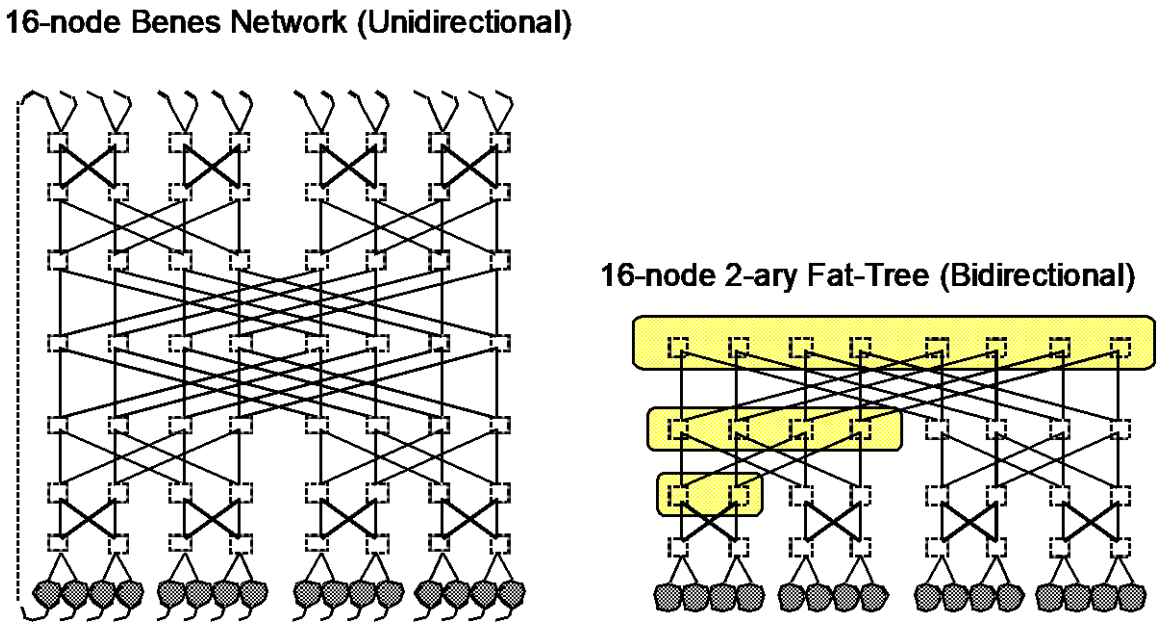
and its bisection width is

What is its degree?

We have shown a fat tree based on a binary tree. It may also be based on a *k*-ary tree. The CM-5 used fat trees based on 4-ary trees:



A *k*-ary fat tree can also be viewed as a *k*-ary Beneš network that is folded back on itself in the high-order dimension:



The collection of $N/2$ switches at level i is viewed as 2^{d-i} "fat nodes" of 2^{i-1} switches, where d is the *dimension* of the switch (where d is the number of levels in the tree—4 in the picture).