# Lesson 2-3 Topology

### Intro to Network Models: Links and Diameter

Network Types:

1D → Linear
2D → Mesh
Fully Connected = every node has a direct connect to every other node.

Links : number of connections (Links).

Linear Network: P-1 links
Mesh Network: 2P links
Fully Connected Network: P(P-1) / 2 links $\approx P^2 / 2$

The number of links is important because it is a proxy for cost - the more links the higher the price.

Diameter: the longest shortest path. Take all pairs of nodes and compute the shortest path, then take the longest of these paths.

Linear Network: P-1 diameter
Mesh Network: $2\sqrt{P} - 1$ diameter
Fully Connected Network: 1 diameter

### Improve the Diameter of a Linear Network

Diameter of a Ring network is: $floor$ (P / 2)

### Improve the Diameter of a 2-D Mesh

*Anything method that connects the corners will cut the diameter in half*.
Wrapping the rows and columns leads to a Torus network.

### Bisection (Band)Width

Bisection Width = the minimum number of links that have to be removed to cut the network into two equal groups of nodes.

Linear Network: bisection width = 1
Ring Network: bisection width = 2
2D Mesh Network: bisection width = $\sqrt{P}$ where P is the number of nodes

Fully Connected Network: $P^2$ / 4

Bandwidth: if equal link speed ( $\beta$ ) then: $\beta 2D(P) = \beta 2D(P) * \beta$

**Improve the Bisection of a 2D Mesh**

Where should links be added to double the bisection width of a 2D Mesh network?
Wrap around links will double the bisection width.

**Some Other Network Topologies**

Other topologies:

Tree:

Links: P links
Diameter: log P links
Bisection: 1

This is a terrible bisection width. So designers add many more wires near the top of the tree.

D dimensional Mesh or Torus:

If it has d root(P) nodes per dimension

Links: dP
Diameter: ½ $d(P)^{1/d}$
Bisection: $2(P)^{(d-1)/d}$

D dimensional meshes and tori are very important, many of the worlds supercomputers use low dimensional meshes.

Hyper Cubes: are log(P) dimensional torus

Links: P*log(P)
Diameter: log(P)
Bisection: P/2

Hypercubes have a high cost of wires but they also have a high bisection width.

| Type Network | Links | Diameter (small = good) | Bisection (large = good) |
|---|---|---|---|
| Linear | P-1 | P-1 | 1 |
| Mesh | 2P | $2\sqrt{P} - 1$ | 2 |
| Fully Connected | $P^2$ / 2 | 1 | $\sqrt{P}$ |
| Tree | P | log(P) | 1 |
| D-Dimensional Mesh | dP | ½ $d(P)^{1/d}$ | $2(P)^{(d-1)/d}$ |
| Hyper Cube | P*log(P) | log(P) | P/2 |

**Mappings and Congestion**

Mappings: running the logical network on an actual physical network.

For example: An algorithm is designed for a ring network, but the underlying physical network is a 2D torus. There should be little to no congestion.

If the algorithm is designed for a 2D torus network, but the underlying phsyical network is a ring, there will be a lot of congestion.

Congestion = maximum number of logical edges that map to a physical edge.

Congestion of a ring going to a torus = 1
Congestion of a torus going to a ring => 6

**Lower Bound on Congestion**

$B_x$ → physical bisection width
L → logical edges cut

$C \geq L/Bx$ and $L \geq BL$

So …. $C \geq L/Bx \geq B_L / B_x$

If you know the congestion. you'll know how much worse the cost of your algorithm will be on a physical network with a lower bisection capacity.

**Exploiting Higher Deimensions**

Exploit the extra capacity of a network that has extra links.