Administrivia

One-sentence proposals due today (now)

Lecture on Friday!!! (but none next Wednesday)

One-page proposals due Monday

Guest lecture on Monday:
Bob Felderman, Myrinet, *The Myrinet Network*

EECS Colloquium also on Monday at 3:00:
Jack Turner, UW, *Gigabit ATM Switches*

Projects

I will check proposals today

Potential problems:

- Too many groups
- Unattached students (i.e. still need a project)
- Too much overlap among certain projects

One-page proposals (due Monday at 5pm):

- Thesis statement
- Methodology
- Deliverables
- Milestones

Multiprocessor Topologies

Review

General properties

Bisection wiring

The Express Cube

The Impact of Locality

Picking a topology

Review of Monday's Lecture

Topology metrics:

- Long vs. short wires
- Complexity (easy to build, manage?)
- Latency: how many hops?
  - Switches: > *50* ns
  - Wire: *15* cm/ns $\Rightarrow$ one switch $\cong$ *7.5* m of wire
- Bisection bandwidth
- Fault tolerance – depends on routing algorithms
- Internal load balancing
- The impact/requirement of locality
- Easy to partition into sub machines?

Review

Mesh vs. Torus:

- Torus has twice the bisection bandwidth
- Torus takes about half as many hops
- Mesh suffers from central congestion
- Fault tolerance depends on routing
  - Torus has slightly better fault tolerance
  - No current implementation uses adaptive routing
- Mesh can be partitioned, torus cannot (?)

Review

Metric	3D Torus	Fat Tree	Mult. Meta.
Requires long wires	No	Yes	Yes
Easy to build	Easiest	Easy	Easy
Latency (hops)	$\Omega(n^{1/3})$	$\Omega(\log n)$	$\Theta(\log n)$
Bisection bandwidth	$\Theta(n^{2/3})$	$\Theta(n)$	$\Theta(n)$
Locality	Required	Allowed	Ignored
Load balancing	OK	Good	Great
Fault tolerance	Difficult	Good	Great
Can be partitioned	No	Easy	Ok
Adjustable bandwidth	No	Yes	Yes
Routers	None	$\Theta(n \lg n)$	$\Theta(n \lg n)$

General Properties

Randomness provides fault/congestion tolerance

Congestion ≅ short-term or intermittent fault

- Avoiding congestion ⇒ can also avoid faults

In the limit, bisection bandwidth is $\Theta(n^{2/3})$

Self-similar ⇔ partitionable

Fat cables are better:

- Fewer cables ⇒ easier to build
- Better queueing properties by pooling resources
- Can be multiplexed to reduce cost & adjust bandwidth

Fault Tolerance

Requires adaptive routing:

- Mult. Meta is always adaptive
- Fat tree is adaptive on the way up
- Torus could be adaptive

Adaptive routing ruins message order:

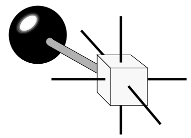
- Different paths ⇒ out-of-order arrival
- Circuit-switched routing preserves order during connection
  - Adaptive routing to set up connection
  - Fixed path after that
  - Reroute connection if a fault occurs
  - But, retransmission also ruins message order

Channel Width

Key torus assumption:
  Can always make channel wider

This is often false – channels are pin limited

- Less true for routers – they can be *cascaded*
- Maybe a torus *should* have routers...



3D torus node must have pins for 6 directions:

- J-Machine limited by pins to 9-bit channel

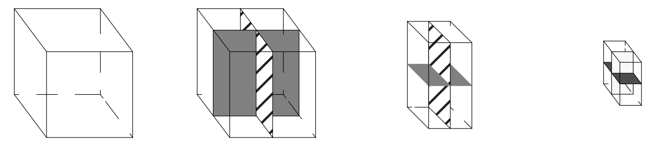
This partially invalidates bisection wiring arguments

Bisection Wiring a Problem?

Assume: ***16K*** nodes, ***300*** cm^3/node, ***25*** cm^3 router

- ***4***-ary butterfly $\Rightarrow \log_4 16384 - 1 = 6$ stages, ***4K*** rows
- Total volume: $16384\,(300) + 4096\,(6)\,(25) = 5.5$ m^2
- Placement into a cube:

1) initial cube 2) first stage 3) second stage 4) third stage



$S \cdot S \cdot S$ | $S \cdot \frac{S}{2} \cdot \frac{S}{2}$ | $\frac{S}{2} \cdot \frac{S}{4} \cdot \frac{S}{2}$ | $\frac{S}{4} \cdot \frac{S}{4} \cdot \frac{S}{4}$

Wire length: S | Wire length: S | Wire length: $S/2$

Stage Placement (continued)

After placing 3 stages:

- Cumulative wire length: $S + S + \frac{S}{2}$
- Place remaining stages in $\frac{S}{4}$-sided cube $\Rightarrow$ recurse

After all stages:

- Total wire length:
  $$\left(S + S + \frac{S}{2}\right)\left(1 + \frac{1}{4} + \frac{1}{16} + \ldots\right) = \frac{5S}{2}(1.33) = 3.33S$$
- ***5.5*** m^3 $\Rightarrow S = 1.76 \Rightarrow \mathrm{TWL} = 5.88$ m
- With ***10*** mm^2 cables: $16384 \cdot 3.33S \cdot 10^{-5} = 0.55S$ m^3
- With $S = 1.76$, volume is ***0.97*** m^3 (< ***5.5***)

Long Fat Wires

Available bisection area: $S^2 = 3.1$ m^2

- Area of bisection wires: $32196 \cdot 10^{-5} = 0.32$ m^2
- Plenty of room, issue is just cost (and power)
- For 32x32x16 torus, only $32 \cdot 16 \cdot 2 = 1024$ cables cross

Latency: 15 cm/ns for wire, 10 ns per switch

- Butterfly: 6 hops, 6 m $\Rightarrow$ 60 + 40 = 100 ns
- 3D Torus, 32x32x16: 20 hops, ignore wires = 200 ns
- Nearest neighbor only $\Rightarrow$ torus only takes 10 ns

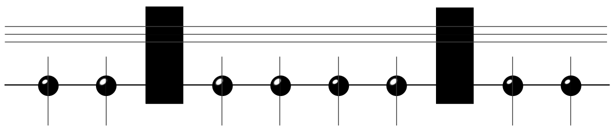
Can use long fat wires as needed

Express Cubes

Goal: fix problems with 3D torus

- ◆ Too many hops
- ◆ Can't really make channels wider because of pins
- ◆ Can't adjust bisection bandwidth

Solution: add routers for longer hops (interchanges)

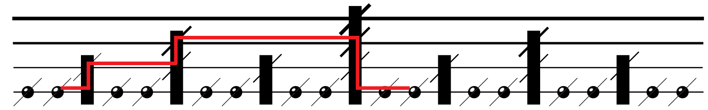


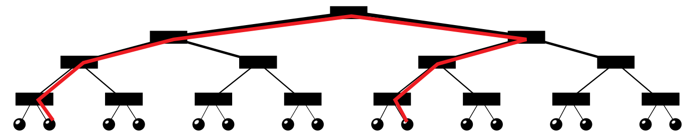
- ◆ Can adjust bandwidth between interchanges

Apply hierarchically to get $\Omega(\log n)$ hops

Comparison with Fat Trees

Example routing pattern:





- ◆ Express cube: 5-9 hops depending on implementation
  - • Seperate tree for each dimension?
- ◆ Fat tree: 8 hops for 2-ary, 4 hops for 4-ary

Express Cubes: Summary

Analysis:

- ◆ Mixture of short and long wires
- ◆ Use long wires to tune bandwidth
- ◆ $\Omega(\log n)$ hops, but probably more than 4-ary fat tree or 4-ary metabutterfly
- ◆ Uses routers to avoid pin limitations and to control bandwidth
- ◆ Number of routers: depends on implementation
  - • $\Theta(n)$ if interchange can cover all levels
  - • $\Theta(n \log n)$ if interchange can switch one level at a time
- ◆ Partitioning depends on mesh/torus
- ◆ Can exploit locality
- ◆ Fault tolerance depends on routing and is unknown

The Impact of Locality

Primary effect: increase the effective bandwidth

- ◆ If a message ties up 2 channels instead of 4, the effective bandwidth doubles
- ◆ Effective bandwidth proportional to average number of hops

Secondary effect: reduce the number of hops

Example: metabutterfly

- ◆ Long packets always tie up $\log n$ channels (versus 1 for nearest neighbor in a torus)
- ◆ Short packets might only tie up one channel at a time, but still tie up $\log n$ channels over their lifetime

Forget adding more wires — worry about reducing the hops!

Picking a Topology

Do you have unlimited money? ⇒ metabutterfly

- Can achieve full bandwidth for any communication pattern
- Lower latency, better load balancing than fat tree

Do you have locality?

- Zero locality ⇒ metabutterfly
- Lots ⇒ 3D torus is the most cost effective
- Less locality favors the express cube or the fat tree, with the fat tree improving as locality decreases

Fault tolerance?

- Lots of faults ⇒ metabutterfly
- Some faults ⇒ fat-tree, *maybe* express-cube with adaptive routing

Picking a Topology

Partitioning:

- Fat tree is the best choice, or express-cube *mesh*
- Metabutterfly is OK, but still pay full $\log n$
- Tori are bad

Alternative: minimize number of components

- 3D torus with separate routers and fat channels
- 3D torus express cube with a few fat interchanges
- Hope for locality

ATM Switches

Think of each ATM switch as a BIG router

- 16 inputs, 16 outputs (or bigger)
- Crossbar or Benes network inside

Very few routers per topology:

- 1024-node metabutterfly requires 64 rows, 3 stages
- = 192 switches
- Each node gets one fiber in and out

Why no planned topologies for NOW?

- No physical locality (so far)
- Incremental construction