EE382C
Lecture 4

High-Radix and Non-Blocking Networks
4/7/11
Question of the day

- What topology has an average hop count, $H_{\text{avg}}$ for load-balanced traffic that is close to the $\log_{d/2}N$ bound and is able to route arbitrary traffic with an $H_{\text{max}}$ of twice this amount?
High-Radix Networks
Bandwidth Trend (ISCA '05)

[Graph showing the bandwidth trend from 1985 to 2010 for various computer architectures, including Torus Routing Chip, Intel iPSC/2, J-Machine, CM-5, Intel Paragon XP, Cray T3D, MIT Alewife, IBM Vulcan, Cray T3E, SGI Origin 2000, AlphaServer GS320, IBM SP Switch2, Quadrics QsNet, Cray X1, Velio 3003, IBM HPS, SGI Altix 3000, Cray XT3, and YARC.]
Router bandwidth
As bandwidth increases ...
Low-Radix vs. High-Radix Router

Low-radix (small number of fat ports) vs. High-radix (large number of skinny ports)
Latency vs. Radix

Optimal radix ~ 40

Optimal radix ~ 128

Header latency decreases

Serialization latency increases
Determining Optimal Radix

Latency = Header Latency + Serialization Latency
         = \( H t_r + \frac{L}{b} \)
         = \( 2t_r \log_k N + \frac{2kL}{B} \)

where \( k = \text{radix} \)
\( B = \text{total Bandwidth} \)
\( N = \# \text{of nodes} \)
\( L = \text{message size} \)

Optimal radix
\[ k \log_2 k = \frac{B t_r \log N}{L} \]

Aspect Ratio
Higher Aspect Ratio, Higher Optimal Radix
$K_N$ – The Ultimate High-Radix Network
High-Radix Butterfly

- Just build a butterfly with large $k$
- For $k = 128$
  - 128 in 1 stage
  - 16K in 2 stages
  - 2M in 3 stages
- But – vulnerable to adversarial traffic
High-Radix Clos

- Not vulnerable to adversarial traffic,
- But twice the number of stages – even on benign traffic.
Flattened Butterfly
Dragonfly Topology

Interconnection Network

R_0, R_1, R_2, ..., R_n

Inter-group Interconnection Network

G_0, G_1, G_{g-1}

intra-group interconnection network

R_0, R_1, ..., R_a
Dragonfly by the numbers

• Consider \( k = 128 \)

• Divide into \( g, l, p \)
  – \( p \) processors per router
  – \( l \) connections to other routers in same group
  – \( g \) global connections – to other groups

• Design method
  – Pick \( g \)
  – Let \( l = 2g - 1 \)
  – Let \( p = 128 - 2g \)
  – Compute concentration factor \( c = \frac{p}{g} \)
An Example

<table>
<thead>
<tr>
<th>d</th>
<th>128</th>
<th>128</th>
<th>128</th>
<th>128</th>
</tr>
</thead>
<tbody>
<tr>
<td>p</td>
<td>93</td>
<td>75</td>
<td>51</td>
<td>33</td>
</tr>
<tr>
<td>l</td>
<td>23</td>
<td>35</td>
<td>51</td>
<td>63</td>
</tr>
<tr>
<td>g</td>
<td>12</td>
<td>18</td>
<td>26</td>
<td>32</td>
</tr>
<tr>
<td>p/g</td>
<td>7.75</td>
<td>4.17</td>
<td>1.96</td>
<td>1.03</td>
</tr>
<tr>
<td>(l+1)g</td>
<td>288</td>
<td>648</td>
<td>1352</td>
<td>2048</td>
</tr>
<tr>
<td>group</td>
<td>2139</td>
<td>2625</td>
<td>2601</td>
<td>2079</td>
</tr>
<tr>
<td>N</td>
<td>618,171</td>
<td>1,703,625</td>
<td>3,519,153</td>
<td>4,259,871</td>
</tr>
</tbody>
</table>
Non-Blocking Networks
Non-blocking networks

- Non-blocking: able to connect any unconnected input to any unconnected output.

- Circuit switching vs. packet switching

- Strictly vs. rearrangeably non-blocking

- Non-interfering networks
Crossbar Switch

in0
in1
in2
in3
out0
out1
out2
out3
out4
input line
crosspoint
output line
Old Mechanical Crossbar
Implemented with Multiplexers
Crossbar Expansion

in0

in(n-1)

inn

in(2n-1)

out0

out(n-1)

outn

out(2n-1)
Clos Networks
Basic Clos Structure (3,3,4)

- **n = 3 ports per switch**
- **r = 4 n x m** input switches
- **m = 3 r x r** middle switches
- **r = 4 m x n** output switches

<table>
<thead>
<tr>
<th>Input Switches</th>
<th>Middle Switches</th>
<th>Output Switches</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1, 1.2, 1.3</td>
<td>Middle Switch 1</td>
<td>1.1</td>
</tr>
<tr>
<td>2.1, 2.2, 2.3</td>
<td>Middle Switch 2</td>
<td>2.1</td>
</tr>
<tr>
<td>3.1, 3.2, 3.3</td>
<td>Middle Switch 3</td>
<td>3.1</td>
</tr>
<tr>
<td>4.1, 4.2, 4.3</td>
<td>Middle Switch 4</td>
<td>4.1</td>
</tr>
</tbody>
</table>

EE 382C - S11 - Lecture 4
Simple Clos (2,2,2)

- **Route**
  - 1.1 to 1.1
  - 2.2 to 2.2
  - 1.2 to 2.1
  - 2.1 to 1.2
Routing example

- Route
  - 1.1 to 1.1
  - 2.2 to 2.2
  - 1.2 to 2.1
  - 2.1 to 1.2
Edge Coloring

![Graph](image)
Edge Coloring

- Route
  - 1.1 to 1.1
  - 2.2 to 2.2
  - 1.2 to 2.1
  - 2.1 to 1.2
Rearrangement

- Route
  - 1.1 to 1.1
  - 2.2 to 2.2
  - 1.2 to 2.1
  - 2.1 to 1.2

- Route next call violating rule
• Route
  – 1.1 to 1.1
  – 2.2 to 2.2
  – 1.2 to 2.1
  – 2.1 to 1.2

• Route next call violating rule
• Fix color conflict at b2 by flipping other edge
Rearrangement

- Route
  - 1.1 to 1.1
  - 2.2 to 2.2
  - 1.2 to 2.1
  - 2.1 to 1.2

- Route next call violating rule
- Fix color conflict at b2 by flipping other edge
- Final circuit no longer conflicts
Rearrangement
Strictly non-blocking

\[ \begin{align*}
\text{a1} & \quad \text{b1} \\
\text{a2} & \quad \text{b2}
\end{align*} \]
Strictly non-blocking Routing

• Number of stages

• Vector of free middle stages at every input and output switch

• E.g. with 15 middle stages:
  – Input $i$ : 0 0 0 1 0 0 0 1 0 1 1 1 1 1 1
  – Output $j$ : 0 1 1 0 1 1 1 0 1 0 0 0 0 1 1

---------------------------------------
3D View of a Clos Network

input switches

middle switches

output switches
Rearrangement Example
(3,3,4) Routing Problem
Before Setting Up (2,4)
Connect (2,4) using RED
Switch (1,4) to **BLUE**

The red link being moved cannot lead back to i2 since i2's red link is already accounted for.
Connect (4,3) using BLUE
Chain of calls to be switched

BLUE/RED

Chain is guaranteed to be acyclic since input blue and output red links of previous nodes are already accounted for.
After Switching 5 calls
Complete Colored Graph
Question of the day

- What topology has an average hop count, $H_{\text{avg}}$ for load-balanced traffic that is close to the $\log_{d/2}N$ bound and is able to route arbitrary traffic with an $H_{\text{max}}$ of twice this amount?
Flattened Butterfly Topology
Non-Blocking Summary

- Non blocking
  - Can connect any unconnected input to any unconnected output
    - Strictly non-blocking: without moving any other connections
    - Rearrangeably non-blocking: may require moving other connections
  - Applies to circuit switching
    - For packet switching you usually want a non-interfering network
- Crossbar
  - Trivially non-blocking
  - Also has stiff backpressure
- Clos network
  - 3 stages of crossbars – each switch of stage i connects to all switches of stage i+1
  - (m,n,r)
  - Strictly non blocking if $m \geq 2n-1$, rearrangeable if $m \geq n$
  - Schedule with the looping algorithm – augmented paths
  - Multicast
    - Conflict vector representation
    - Can fanout in input or middle stage (split calls)