Topologies - II

Overview

- Express Cubes
- Flattened Butterfly
- Fat Trees
- DragonFly

Express Cubes

- The Problem
  - Node delay dominates wire delay
  - Pin densities are not as high as wiring densities
  - In a \( k \)-ary \( n \)-cube, \( n \), \( k \), and \( W \) are all related
- Consequently networks are node limited rather than wiring limited
  - Unused capacity
- Goals:
  - Design a network that can make use of the unused wiring capacity
  - Approach the wire delay lower bound

Express Links

\[
\text{Interchange box} \quad \text{express channel} \quad \text{Wire delay} = i \cdot T_w
\]

\[
\text{Non-express latency} \quad T_a = (T_n + T_w)D + \frac{L}{W} T_p
\]

Express latency \( T_b = \left( \frac{D}{i} + i \right) T_n + T_w D + \frac{L}{W} T_p \)
Balancing Node and Wire Delay

- Reduce the node delay component of latency
- Express channel length chosen to balance wire delay and node delay
- For large D, the latency is within a factor of 2 of dedicated Manhattan wire latency
- Pick $i$ based on average distance and relationship between wire delay and node delay

Balancing Throughput and Wiring Density

- Exploit available wiring density to move beyond pin-out limitations
- Consider wiring density of the substrate
- More complex interchanges
- Good utilization of express channels
- Routing on express links

Balancing Throughput and Wiring Density

- Simpler interchange box design
- Routing in a dimension
- Uneven traffic distribution across links

Hierarchical Express Channels

- Messages in the express segment are dominated by node latency
  - Reduce by making this latency growth as logarithm
  - Logarithmic for short distances and linear for long distances
- Port assignment for messages at each level
- Combine advantages of direct and indirect networks
- Routing in three phases: ascent, cruise, descent
Latency Behavior

- Saw tooth pattern reflects latency jumps
- Hierarchical channels smoothes out latency variations

Reducing Pin Out

- Implementing interchanges with constant pin-out
  - Small latency penalty
- These can be further reduced at the expense of a few more hops

Implementing Multiple Dimensions

- Pinout of interchange boxes is kept constant
- Messages have to descend to local channels to change dimensions

Summary

- Push to create networks limited by wire delay and wire density
  - For long distances, latency approaches that of a wire
  - Increase the bisection width of the network
- Baseline of k-ary n-cubes – efficient use of bisection width
- Hierarchical express cubes combines logarithmic delay properties with wire efficiency and locality exploitation
- What happens when high radix routers become feasible?
Flattened Butterfly

- Pin bandwidth and pin-out have improved over the years
  - Low dimensional networks do not make good use of this greater I/O bandwidth
- Implications of improved off-chip bandwidth?

Reading Assignment


Topology Tradeoff

- Retain the desirable properties of MINs
  - Logarithmic diameter
  - Cost
- Exploit the properties of folded Clos Networks
  - Path diversity
  - Ability to exploit locality

Analysis

- Logarithmic path lengths of MINs are offset by lack of path diversity and ability to exploit locality
- Need to balance cost (switches, links/cables/ connectors), latency and throughput
  - Reduce the channel count by reducing diameter
  - Reduce the channel count by concentration
- Better properties are achieved by paying for the above with increased switch radix
Another Look at Meshes

- Trade-off wire length and hop count
  - Latency and energy impact
- Take advantage of high radix routers

Key Insight

- Goal is to balance serialization latency and header latency
- Meshes only reduce serialization latency
- FB trades serialization latency for header latency via higher radix switches

Using Concentrators

- Better balance of injection bandwidth and network bandwidth
  - How often do all processors want to communicate concurrently?
- Significant reduction in wiring complexity

Construction

- Note that the inter-router connections are determined by permutations of the address digits
  - For example in the above figure (a) \( (21)_4 \rightarrow (12)_4 \)
### Properties

- For a $k$-ary $n$-flat with $N = k^n$ nodes we have
  - $N/k$ routers in $n-1$ dimensions with $k' = n(k-1) + 1$ router radix
  - Built for size
  - Need high radix routers

### On-Chip Structure

- Structurally similar to generalized hypercube
- Attempts to approach the wire bound for latency
- Latency tolerant long wires (pipelined, repeated)
- Deeper buffers
- Radix $k$-based routing

### Wire Delay

- Longer physical wires
- Topology packaging has minimal physical (manhattan) distance
  - Wire length close to minimal physical wire length

### Router Optimizations

- Non-minimal routing
  - Using Bypass Channels

---

Figure from J. Kim, J. Balfour, and W. J. Dally, "Flattened Butterfly Topology for On-Chip Networks," Proceedings of MICRO 2007
Connectivity

\[ j = i + m \left( m - \left( \left\lfloor \frac{i}{k^{d-1}} \right\rfloor \right) \mod k \right) k^{d-1} \]

- Each switch \( i \) in a stage is connected to \( j \) for \( m = 0 \) to \( k-1 \)

Routing

- Basic dimension order traversal
  - However in MINs all paths are of length
- In FB, only necessary dimensions need be traversed (remember binary hypercubes!)
- Number of shortest paths is factorial in the number of differing dimensions
  - In the MIN, dimension order is fixed
- Non-minimal routing for better network-wide load balancing
  - Enables performance equivalent to flattened Clos

Packaging for High Node Count

- For a 16-ary 4-flat network
- Can scale to 64K nodes

Packaging in three dimensions
Tradeoff bisection bandwidth and node delays

Construction with Fixed Radix

\[ \left[ \frac{k}{n^d+1} \right] \geq N \]
\[ k' = \left( \frac{k}{n^d+1} - 1 \right) (n^{d}+1) + 1 \]
Comparison to Generalized Hypercubes

• Use of router bandwidth
  - Concentration in a k-ary n-flat produces better link utilization and lower cost
  - Example: Use 1K nodes and radix 32 switches
    * GHC – (8,8,16)
    * FB – one dimension

If BW is matched, serialization latency will dominate

• Load balanced non-minimal routing in FB

Fat Trees


• Simple engineering premise: a network topology that has the advantages of the binary tree without the problems

Comparison

• For 1024 nodes
  - Flattened butterfly
    * Radix 32 switches and two stages
  - Folded Clos
    * Radix 64 switches and two stages
  - Binary hypercube
    * 10 dimensional hypercube

• Trade-off analysis will keep the bisection width constant

Reading Assignment

• Xin Yuan et. al., “Oblivious Routing for Fat-Tree Based System Area Networks with Uncertain Traffic Demands,” SIGMETRICS 2007, Section 2.2 (until property 1).

• Recommended
**Fat Trees: Basic Idea**

- Alleviate the bandwidth bottleneck closer to the root with additional links
- Common topology in many supercomputer installations

**Alternative Construction**

- Nodes at tree leaves
- Switches at tree vertices
- Building crossbars with simpler switches
- Total link bandwidth is constant across all tree levels, with full bisection bandwidth
- This construction is also known as having constant bisection bandwidth

**FT(4,4): Sub-trees**

- Built with constant radix \( m \) switches and \( L \) levels
- Can be viewed as a less costly way of building crossbar switches

---

*X. Lin, Y. Chung and T. Huang, “A Multiple LID Routing Scheme for Fat-tree Based InfiniBand Networks,” IEEE IPDPS 2004*
Properties

- The properties follow from multistage interconnection networks
- Number of inputs as a function of $m$ and $L$

\[
\text{Number of processors connected per switch: } \frac{m}{2}
\]

\[
\text{Number of reachable switches on the spine: } \left(\frac{m}{2}\right)^{L-1}
\]

Generalized Fat Trees

- Networks with variable bisection bandwidth
- Asymmetric use of switch radix between successive levels
- Can construct expanders and concentrators
- \( \text{GFT} (L,m,w) \)
  - \( L \) levels
  - Bandwidth ratio of \( m:w \) between levels

Revisiting the Clos Network

Folded Clos Network

- Network is effectively folded into itself to produce a fat tree
- Consequently often referred to as a folded Clos Network
  - Note this is also equivalent to a bidirectional Benes network
  - Rearrangeable when $m \geq n$

Implications

- The connectivity of this class of fat trees are equivalent to that of a crossbar
- Realizing the performance of a crossbar is another matter
  - Recall it is not strictly non-blocking (crossbar) but rather rearrangeable
  - Packet scheduling and congestion management are key to performance
- Achievable performance a function of the communication pattern

Basic Routing

- Compare addresses of source and destination
  - Digit position of the first difference identifies the stage/level is to turn around.
  - Any path up the tree
  - Deterministic routing down the tree
- Load balanced routing randomizes the up path
Example: Roadrunner

- 122400 nodes
- Infiniband: 2-level fat-tree
- Each leaf switch has 180 down links and 96 up links (18 such CUs), 12 up links from each CU connected each of the 2nd level switches

From http://charm.cs.uiuc.edu/~bmatele/phd/topology.htm

Dragonfly Topology

- How do we build networks that scale to $>10^6$ nodes?
  - Technology to date had prescribed low dimensional networks
  - Target – Exascale computing systems
- Topology challenge
  - Reconciling trade-offs between diameter, switch radix, and cost
  - What is a feasible radix to use vs. need for scalability?
  - Indirect topology

Motivation

- Key question: How do we construct 1M+ node systems?
- Hypothesis: Need high radix switches (to reduce the diameter) and efficient long distance communication (due to size and low hop count)
  - Relationship between hop count, diameter, and wire length (more when we get to optimization)

Reading Assignment


Many of the figures in the following slides are from the preceding paper
**Dragonfly: Drivers**

- Increasing pin bandwidth has moved the design point
  - More channels, smaller diameter rather than less channels higher bandwidth/channel
  - Increases the number and length of cables
- Network cost proportional to number of channels, especially global channels
  - Reduce the number of global channels
  - Use new signaling technology for long channels → optical signaling
- Engineering optimizations to increase the effective switch radix

**Topology Basics**

- One level hierarchy
- Traffic limited to one global hop
- Flexible definition of intra-group interconnection

**Properties**

- Switch radix
  \[ k = p + a + h - 1 \]
- Effective radix (group radix)
  \[ k = a(p + h) \]
- Number of groups
  \[ g = ah + 1 \]
- Number of nodes
  \[ N = ap(ah+1) \]

**Variations**

- To reduce hop count need high radix
  - Note the virtual radix of a group
- Sheer size still implies long global links
  - Need to use these efficiently
Topology Variations

- The inter-group network
  - Downsizing: redistribute the global channels amongst the groups
  - Upsizing: move to multiple global hops
  - Rightsizing: increase virtual radix to maintain single hop property, e.g., radix 64, 2D k-ary n-flat intra-group network and 256K nodes

- The intra-group network
  - Trade-off local hops vs. physical radix

More on Properties

- Bisection bandwidth follows inter-group network
- Pin-out (radix) is a design parameter
  - Defines network scale and diameter
  - Defines local (group) hop count

Balanced Design

- How do we think about values of parameters \( a, p, \) and \( h \)?
  - Balancing traffic on global and local channels
    
    \[ a \geq 2p \geq 2h \]

- Balancing cost
  - Trade-off switch radix with number of global channels (cables)

Routing

- Minimal routing takes at most three steps
  - Route to the router within the source group that connects to the correct destination group (multi-hop?)
  - Route across the global channel to the correct destination group
  - Route to the correct destination router within the destination group that connects to the destination (multi-hop?)

- Does not work well under adversarial patterns
  - Use Valiant’s 2-phase routing algorithm
    - Route to a random intermediate group
    - Route to the destination

From J. Kim, et.al, “Technology-driven highly scalable Dragonfly topology,” ISCA 2008
A Baseline System

- 12 networks
- One radix 64 router/network
- 12 processor channels
- 31 rack groups
- 21 other cabinets

21x32 = 672 global channels out of each rack to unique cabinets

Summary

- Topology designed to scale to large numbers of nodes (>10^6)
  - Sensitive to a hierarchy of interconnect delays and costs
- Exploits increased pin-bandwidth in emerging technologies
- We will return to more sophisticated routing algorithms later

Hybrid Networks

- Topology designed to scale to large numbers of nodes (>10^6)
  - Sensitive to a hierarchy of interconnect delays and costs
- Exploits increased pin-bandwidth in emerging technologies
- We will return to more sophisticated routing algorithms later

Comparison of Direct and Indirect Networks

- Concentration can be used to reduce direct network switch & link costs
  - "C" end nodes connect to each switch, where C is concentration factor
  - Allows larger systems to be built from fewer switches and links
  - Requires larger switch degree
  - For N = 32 and k = 8, fewer switches and links than fat tree
Comparison of Direct and Indirect Networks

Distance scaling problems may be exacerbated in on-chip MINs.

Switches

End Nodes

Commercial Machines

A Unified View of Direct and Indirect Networks

- Switch designs in both cases are coalescing
  - Generic network may have 0, 1, or more compute nodes/switch

- Switches implement programmable routing functions

- Differences are primarily an issue of topology
  - Imagine the use of source routed messages

- Deadlock avoidance

ECE 8813a (61)

ECE 8813a (62)

ECE 8813a (63)

ECE 8813a (64)
Summary and Research Directions

- Use of hybrid interconnection networks
  - Best way to utilize existing pin-out?

- Engineering considerations rapidly prune the space of candidate topologies

- Routing + switching + topology = network