Science of Communication Networks
The University of Kansas EECS 784
Regular Networks

James P.G. Sterbenz
Department of Electrical Engineering & Computer Science
Information Technology & Telecommunications Research Center
The University of Kansas

jpgs@eeecs.ku.edu

http://www.ittc.ku.edu/~jpgs/courses/scinets
Regular Networks

Outline

RN.1  Regular network types and properties
RN.2  Nearly-regular constructions
RN.3  Interconnection networks

Primary references:
[B]=[B1965], [RF]=[RF1987]
[L]=[L2009]
Regular Networks

RN.1 Types and Properties

RN.1 Regular network types and properties
RN.2 Nearly-regular constructions
RN.3 Interconnection networks
Regular Networks

Definition

- **Regular network** has a repeating pattern in structure
  - may or may not be a $k$-regular graph
  - entropy is zero or low
    - *entropy* is amount of randomness in graph structure
- **Examples**
  - ring, torus, hypercube, full mesh ($k$-regular)
  - linear, grid (not $k$-regular)
- **In this class**
  - will say $k$-regular rather than regular for $k$-regular graphs
Regular Networks

Types

- Linear and ring
- Manhattan grid and toroid
- Tree and star
- Hypercube
- Complete mesh
- Motif
Regular Networks

Linear and Ring

- Linear network $L_n$
  - linear sequence of connected vertices $v_0, e_{01}, v_1, e_{12}, \ldots, v_{n-1}$

Properties?
  - scale?
  - degree?
  - diameter?
  - clustering coëfficient?
  - connectedness?
  - adjacency matrix?

Note: [L2009] calls these “line networks” but this could be confused with the line graph $L(G)$
Regular Networks

Linear and Ring

- Linear network $L_n$
  - linear sequence of connected vertices $v_0, e_{01}, v_1, e_{12}, \ldots, v_{n-1}$

- Properties
  - scale: $|L_n| = n \Rightarrow ||L_n|| = n - 1$
  - degree: $1 \leq d(L_n) \leq 2$
  - diameter: $\text{diam}(L_n) = n - 1$
  - clustering coefficient: $\text{cc}(L_n) = 0$
  - connectedness: not biconnected
  - adjacency matrix: diagonals

Use in communication networks?
Regular Networks
Linear and Ring

- Linear network $L_n$
  - linear sequence of connected vertices $v_0, e_{01}, v_1, e_{12}, \ldots, v_{n-1}$

- Use in communication networks
  - cost?
  - resilience?
  - scalability?
  - properties?
Regular Networks
Linear and Ring

- **Linear network** $L_n$
  - linear sequence of connected vertices $v_0, e_{01}, v_1, e_{12}, \ldots, v_{n-1}$

- **Use in communication networks**
  - cost: cheapest interconnection
  - resilience: poor since not biconnected
  - scalability: poor given high betweenness toward center
  - properties
    - rarely appropriate for physical network infrastructure
    - end-to-end path is a linear network graph
Regular Networks

Linear and Ring

• Ring network $R_n$
  – wrapped line network $v_0, e_{01}, v_2, e_{12}, \ldots, v_{n-1}, e_{n-1,0}$

Properties?
- scale?
- degree?
- diameter?
- clustering coefficient?
- connectedness?
- adjacency matrix?
Regular Networks

Linear and Ring

- **Ring network** $R_n$
  - wrapped line network $v_0, e_{01}, v_2, e_{12}, \ldots, v_{n-1}, e_{n-1,0}$

- **Properties**
  - scale: $|R_n| = n \Rightarrow ||R_n|| = n$
  - degree: $d(R_n) = 2$
  - diameter: $\text{diam}(R_n) = \lfloor n/2 \rfloor$
  - clustering coëfficient: $cc(R_n) = 0$
  - connectedness: biconnected
  - adjacency matrix: diagonals + corners

*Use in communication networks?*
Regular Networks

Linear and Ring

- Ring network $R_n$
  - wrapped line network $v_0, e_{01}, v_2, e_{12}, \ldots, v_{n-1}, e_{n-1,0}$
- Use in communication networks
  
  cost?
  resilience?
  scalability?
  properties?
Regular Networks

Linear and Ring

• Ring network $R_n$
  - wrapped line network $v_0, e_{01}, v_2, e_{12}, \ldots v_{n-1}, e_{n-1,0}$

• Use in communication networks
  - cost: relatively cheap interconnection
  - resilience: resilient to single failure
  - scalability: limited but traffic balanced along ring
  - properties:
    • common for access network physical infrastructure
      - SONET (synchronous optical network)/SDH metropolitan ring
    • sometimes used for application overlay
      - when applications need to take turns in sequence
Regular Networks
Manhattan Grid and Torus

- Manhattan grid $M_{n,m}$
  - rectangular array of vertices
  - interconnected set of linear nets
    $$M_{n,m} = \bigcup_m L_n + \{e_{n,m}\}$$

Properties?
- scale?
- degree?
- diameter?
- clustering coefficient?
- connectedness?
- adjacency matrix?
Regular Networks
Manhattan Grid and Torus

- Manhattan grid $M_{n,m}$
  - rectangular array of vertices

- Properties
  - scale: $|M_{n,m}| = n \Rightarrow ||M_{n,m}|| = 2(n-1)(m-1)$
  - degree: $2 \leq d(M_{n,m}) \leq 4$
  - diameter: $\text{diam}(L_n) = n + m - 2$
  - clustering coefficient: $cc(M_{n,m}) = 0$
  - connectedness: well connected
  - adjacency matrix: multiple diag. depending on vertex order

Use in communication networks?
Regular Networks
Manhattan Grid and Torus

- Manhattan grid $M_{n,m}$
  - rectangular array of vertices
- Use in communication nets
  - cost?
  - resilience?
  - scalability?
  - properties?
Regular Networks
Manhattan Grid and Torus

• Manhattan grid $M_{n,m}$
  - rectangular array of vertices

• Use in communication nets
  - cost: moderate
  - resilience: multiply connected
  - scalability: good
  - properties
    • physical node location not generally grid
    • occasionally appropriate for wireless mesh network
    • Gabriel graph better representation for arbitrary node location
Regular Networks
Manhattan Grid and Torus

- Torus $T_{n,m}$
  - wrapped grid

Properties?
- scale?
- degree?
- diameter?
- clustering coefficient?
- connectedness?
- adjacency matrix?
Regular Networks

Manhattan Grid and Torus

• Torus $T_{n,m}$
  - wrapped grid

• Properties
  - scale: $|T_n| = n \Rightarrow ||T_n|| = 2nm$
  - degree: $d(T_{n,m}) = 4$
  - diameter: $\text{diam}(T_{n,m}) = 2[n/2]; \ n > m$
  - clustering coëfficient: $cc(R_n) = 0$
  - connectedness: well connected
  - adjacency matrix: diagonals + corners

Use in communication networks?
Regular Networks
Manhattan Grid and Torus

- Torus $T_{n,m}$
  - wrapped grid
- Use in communication nets
  - cost?
  - resilience?
  - scalability?
  - properties?
Regular Networks

Manhattan Grid and Torus

- Torus $T_{n,m}$
  - wrapped grid
- Use in communication nets
  - cost: moderate
  - resilience: multiply connected
  - scalability: good
  - properties
    - physical node location not generally grid
    - long links not cost-effective for physical deployment
    - sometimes used as multiprocessor interconnection network
Regular Networks

Tree and Star

- **Tree network** $T_{b,l,n}$ $b =$ fanout, $l =$ depth
  - balanced tree: $n = (b^l - 1) / (b - 1)$

**Properties?**
- degree?
- diameter?
- clustering coefficient?
- connectedness?
- adjacency matrix?

<table>
<thead>
<tr>
<th>0 1 1 0 0 0 0 ...</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 0 0 1 1 0 ...</td>
</tr>
<tr>
<td>1 0 0 0 0 1 ...</td>
</tr>
<tr>
<td>0 1 0 0 0 ...</td>
</tr>
<tr>
<td>0 1 0 0 0 ...</td>
</tr>
<tr>
<td>0 0 1 0 0 0 ...</td>
</tr>
<tr>
<td>... ... ... ... ... ... ... ...</td>
</tr>
</tbody>
</table>
Regular Networks

Tree and Star

- Tree network $T_{b,l,n}$ $b=$ fanout, $l=$ depth
  - balanced tree: $n=(b^{l-1})/(b-1)$

- Properties
  - scale: $|T_n|=n \Rightarrow ||T_{b,l,n}||=(b^l-b)/(b-1)$
  - degree: $d(T_{b,l,n})=b+1$ or 1 (leaf)
  - diameter: $\text{diam}(T_{b,l,n})=2(l-1)$
  - clustering coefficient: $cc(T_{b,l,n})=0$
  - connectedness: not biconn.; no loops
  - adjacency matrix: (binary tree shown)

Use in communication networks?
Regular Networks

Tree and Star

- Tree network $T_{b,l,n}$, $b =$ fanout, $l =$ depth
  - balanced tree: $n = \frac{(b^l - 1)}{(b - 1)}$
- Use in communication networks

  - cost?
  - resilience?
  - scalability?
  - properties?
Regular Networks

Tree and Star

• Tree network \( T_{b,l,n} \) \( b=\) fanout, \( l=\) depth
  - balanced tree: \( n=(b^{l-1})/(b-1) \)

• Use in communication networks
  - cost: relatively cheap interconnection
  - resilience: not biconnected
  - scalability: scalable; bottleneck near root
  - properties:
    - useful for passive optical access networks
    - spanning tree overlay for multicast
      - spanning tree can be embedded in any graph
Regular Networks

Tree and Star

- Star network $S_n$
  - tree with only one internal vertex $v_0, e_{01}, v_1, e_{02}, \ldots, v_{n-1}, e_{0,n-1}$

Properties?
  - degree?
  - diameter?
  - clustering coëfficient?
  - connectedness?
  - adjacency matrix?
Regular Networks
Tree and Star

• Star network \( S_n \)
  - tree with only one internal vertex
    \( v_0, e_{01}, v_1, e_{02}, \ldots, v_{n-1}, e_{0,n-1} \)

• Properties
  - scale: \( |S_n| = n \Rightarrow ||S_n|| = n-1 \)
  - degree: \( d(S_n) = 1 \) or \( n-1 \)
  - diameter: \( \text{diam}(S_n) = 2 \)
  - clustering coefficient: \( \text{cc}(S_n) = 0 \)
  - connectedness: internal vertex
  - adjacency matrix: 1st row and column

Use in communication networks?
Regular Networks

Tree and Star

- **Star network** $S_n$
  
  - tree with only one internal vertex $v_0, e_{01}, v_1, e_{02}, \ldots v_{n-1}, e_{0,n-1}$

- **Use in communication networks**
  
  *cost?*
  
  *resilience?*
  
  *scalability?*
  
  *properties?*
Regular Networks

Tree and Star

- Star network $S_n$
  - tree with only one internal vertex (hub)
    $v_0$, $e_{01}$, $v_1$, $e_{02}$, $\ldots$, $v_{n-1}$, $e_{0,n-1}$

- Use in communication networks
  - cost: cheap if leaves close to hub
  - resilience: loss of hub fatal to network
  - scalability: 1 new link for every node
  - properties:
    - useful topology for small LANs
      - e.g. home or small office network
    - not practical for wide-area infrastructure
    - overlay graph client/server applications

<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>1</th>
<th>1</th>
<th>1</th>
<th>$\ldots$</th>
<th>1</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>$\ldots$</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>$\ldots$</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>$\ldots$</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>$\ldots$</td>
<td>$\ldots$</td>
<td>$\ldots$</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>$\ldots$</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>$\ldots$</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Regular Networks

Hypercube Networks

- Hypercube Network $H_D$
  - $D$-dimensional square

Properties?
  - degree?
  - diameter?
  - clustering coëfficient?
  - connectedness?
  - adjacency matrix?

Regular Networks

Hypercube Networks

- **Hypercube Network** \(H_d\)
  - \(D\)-dimensional square
- **Properties**
  - scale: \(|H_d| = 2^d \Rightarrow ||H_d|| = d2^{d-1}\)
  - degree: \(d(H_d) = d\)
  - diameter: \(\text{diam}(H_d) = d\)
  - clustering coefficient: \(cc(H_i) = 0\)
  - connectedness: multiple paths
  - adjacency matrix:

  Use in communication networks?

Regular Networks

Hypercube Networks

- Hypercube Network $H_d$
  - $D$-dimensional square
- Use in communication networks
  - cost?
  - resilience?
  - scalability?
  - properties?
Regular Networks

Hypercube Networks

- Hypercube Network \( H_d \)
  - \( D \)-dimensional square

- Use in communication networks
  - cost: moderate, related to number of nodes
  - resilience: resilient to node and link failures
  - scalability: scalable in links
  - properties:
    - useful topology for HPC (high perf. computer) interconnection
    - not practical for wide-area infrastructure
Regular Networks

Full Mesh Networks

- Full mesh network $C_n$
  - complete graph

**Properties?**

- degree?
- diameter?
- clustering coëfficient?
- connectedness?
- adjacency matrix?
Regular Networks

Full Mesh Networks

- Full mesh network $C_n$
  - complete graph

- Properties
  - scale: $\| C_n \| = n \Rightarrow || C_n || = n(n-1)$
  - degree: $d(T_{n,m}) = n-1$
  - diameter: $\text{diam}(T_{n,m}) = 1$
  - clustering coëfficient: $cc(R_n) = 1$
  - connectedness: totally connected
  - adjacency matrix: all except diagonal

*Use in communication networks?*
Regular Networks

Full Mesh Networks

• Full mesh network $C_n$
  – complete graph
• Use in communication networks
  
  \textit{cost?}
  
  \textit{resilience?}
  
  \textit{scalability?}
  
  \textit{properties?}
Regular Networks

Full Mesh Networks

- Full mesh network $C_n$
  - complete graph

- Use in communication networks
  - cost: maximum with $O(n^2)$ links
  - resilience: as resilient as possible
  - scalability: $n-1$ new links for every node
  - properties:
    - every set of vertices is a clique
    - too expensive for physical infrastructure
    - logical overlays mesh-like
      - but not fully connected
      - interface and processing cost too high
Regular Networks
RN.2  Nearly-Regular Constructions

RN.1  Regular network types and properties
RN.2  Nearly-regular constructions
RN.3  Interconnection networks
Nearly-Regular Constructions

Introduction

• Some network constructions are not regular
  – but share some similarities
  – no standard terminology (?)
  – in this lecture called *nearly-regular constructions*

• Give a set of vertices $V$
  – use a well-defined construction algorithm to place edges $E$

• Examples:
  – geometric graphs
  – Gabriel graphs
  – meshlike graphs
Nearly-Regular Constructions

Geometric Graph

- Geometric graph
  - edges within a given threshold adjacent
Nearly-Regular Constructions
Geometric Graph

- **Geometric graph**
  - edges within a given threshold adjacent
- Given pair of vertices $u, v \in V$
  - place an edge $e \in E$ iff $d(u, v) \leq r$
    - $d(v_1, v_3) \leq r$?
    - $d(v_1, v_3) \leq r$?
Nearly-Regular Constructions
Geometric Graph

- **Geometric graph**
  - edges within a given threshold adjacent

- Given pair of vertices $u, v \in V$
  - place an edge $e \in E$ iff $d(u, v) \leq r$
    - $d(v_1, v_3) < r$
    - $d(v_1, v_3) > r$
Nearly-Regular Constructions

Geometric Graph

- **Geometric graph**
  - edges within a given threshold adjacent

- **Given pair of vertices** $u, v \in V$
  - place an edge $e \in E$ iff $d(u, v) \leq r$
Nearly-Regular Constructions

Geometric Graph

- **Geometric graph**
  - edges within a given threshold adjacent
- Given pair of vertices $u, v \in V$
  - place an edge $e \in E$ iff $d(u, v) \leq r$

*Example in communication networks?*
Nearly-Regular Constructions
Geometric Graph

- Geometric graph
  - edges within a given threshold adjacent
- Example in communication networks:
  - connectivity graph of wireless nodes
    - in range of one another
    - given uniform
      - transmission power
      - attenuation
Nearly-Regular Constructions

Gabriel Graph

- *Gabriel graph*
  - edges that are nearest are adjacent
Nearly-Regular Constructions

Gabriel Graph

- **Gabriel graph**
  - edges that are nearest are adjacent
- Given pair of vertices \( u, v \in V \)
Nearly-Regular Constructions

Gabriel Graph

- **Gabriel graph**
  - edges that are nearest are adjacent
- **Given pair of vertices** \( u, v \in V \)
  - place an edge \( e \in E \) iff
Nearly-Regular Constructions
Gabriel Graph

- **Gabriel graph**
  - edges that are nearest are adjacent
- **Given pair of vertices** \( u, v \in V \)
  - place an edge \( e \in E \) iff
  - no other vertex \( w \) exists in the closed disc
  - of which the line segment \( uv \) is a diameter
Nearly-Regular Constructions

Gabriel Graph

- Gabriel graph
  - edges that are nearest are adjacent
- Given pair of vertices \( u, v \in V \)
  - place an edge \( e \in E \) iff
  - no other vertex \( w \) exists in the closed disc
  - of which the line segment \( uv \) is a diameter
  - else, recurse on \( u, w \) and \( v, w \)
Nearly-Regular Constructions

Gabriel Graph

- **Gabriel graph**
  - edges that are nearest are adjacent
- **Given pair of vertices** \( u, v \in V \)
  - place an edge \( e \in E \) iff
  - no other vertex \( w \) exists in the closed disc
  - of which the line segment \( uv \) is a diameter
  - else, recurse on \( u, w \) and \( v, w \)

*Example in communication networks?*
Nearly-Regular Constructions

Gabriel Graph

- **Gabriel graph**
  - edges that are nearest are adjacent
- Given pair of vertices $u, v \in V$
  - place an edge $e \in E$ iff
  - no other vertex $w$ exists in the closed disc
  - of which the line segment $uv$ is a diameter

*Example in communication networks?*

Nearly-Regular Constructions

Gabriel Graph

- Gabriel graph
  - edges that are nearest are adjacent
- Example in communication networks:
  - resemble physical infrastructure graphs
    - e.g. fiber-optic physical network
    - grid-like topology (as opposed to mesh-like topology)
Grid-Like Graph

Example: Sprint L1 Physical Fiber Topology

L3
L2.5
L1
Nearly-Regular Constructions

Mesh-Like Graph

• Mesh-like graph
  – edges unrelated to distance
  – engineered to give the overall graph desired properties
    • degree distribution vs. diameter to balance costs

Example in communication networks?
Nearly-Regular Constructions
Mesh-Like Graph

- Mesh-like graph
  - edges unrelated to distance
  - engineered to give the overall graph desired properties
    - degree distribution vs. diameter to balance cost

- Examples in communication networks
  - logical overlay networks
    - IP-level router connectivity
    - application overlays
Mesh-Like Graph
Example: Sprint L3 IP PoP Topology
Multilevel Network Graph
Example: Sprint L1–3 Topology
Regular Networks
RN.3  Interconnection Networks

RN.1  Regular network types and properties
RN.2  Nearly-regular constructions
RN.3  Interconnection networks
Interconnection Network

Introduction

- **Interconnection network**
  - set of vertices (switch elements)
  - set of edges (links)

- To interconnect set of terminal vertices
  - switch or router interfaces or linecards
  - multiprocessor or HPC cluster processors, memory, and I/O
    - HPC = high performance computing
Interconnection Network
Types

- **Stages**
  - single stage: single switch element
  - multistage (MIN): multiple stages of interconnected elements

- **Topology**
  - crossbar
  - delta, etc.
  - Clos

- **Blocking characteristics**
  - strictly nonblocking
  - wide-sense nonblocking
  - blocking
Interconnection Network Blocking

- Blocking when one path prevents another crossing
  - when connecting a different input/output pair
- Blocking characteristics
  - strictly nonblocking: never blocks
  - wide-sense nonblocking: if proper routing algorithm used
  - re-arrangably nonblocking: existing paths may be moved
  - blocking
Switch Fabric Architecture

Single Stage: Basic $2 \times 2$ Switch Element

- States
  - point-to-point
    - straight
    - cross
  - multicast
Switch Fabric Architecture

Single Stage: Basic 2×2 Switch Element

- States
  - point-to-point
    - straight
    - cross
  - multicast
Switch Fabric Architecture

Single Stage: Basic 2×2 Switch Element

- **States**
  - point-to-point
    - straight
    - cross
  - multicast

![Diagram of switch fabric architecture](image)
Switch Fabric Architecture

Single Stage: Basic 2×2 Switch Element

- States
  - point-to-point
    - straight
    - cross
  - multicast

![Diagram of Switch Fabric Architecture]

- States
  - point-to-point
    - straight
    - cross
  - multicast

![Diagram of Switch Fabric Architecture]
Switch Fabric Architecture
Single Stage: Basic 2×2 Switch Element

- States
  - point-to-point
    - straight
    - cross
  - multicast
Switch Fabric Architecture

Single Stage: Basic 2×2 Switch Element

- **States**
  - point-to-point
    - straight
    - cross
  - multicast
Switch Fabric Architecture
Single Stage: Basic 2×2 Switch Element

- **States**
  - point-to-point
    - straight
    - cross
  - multicast

![Switch Fabric Architecture Diagram](image-url)
Switch Fabric Architecture

Single Stage: Basic $2 \times 2$ Switch Element

- **States**
  - point-to-point
    - straight
    - cross
  - multicast

- **Types**
  - internally buffered or unbuffered
  - self routing or externally controlled
Switch Fabric Architecture

Single Stage: Crossbar Switch

- Crosspoint switch element
  - electronic
  - optical MEMS
    - rotating mirror
Switch Fabric Architecture

Single Stage: Crossbar Switch

- **Crossbar** fabric
  - square array of crosspoint elements
  - $O(n^2)$ growth complexity
  - reasonable for moderate $n$
Crossbar Switch
Path Selection

- Crossbar fabric
  - simple path routing
  - element \((o,i)\) turns
Crossbar Switch

Path Selection

- Crossbar fabric
  - simple path routing
    - element \((o,i)\) turns
    - \(i_3 \rightarrow o_4\)
Crossbar Switch
Path Selection

• Crossbar fabric
  – simple path routing
  • element \((o,i)\) turns
  • \(i_3 \rightarrow o_4\)
Crossbar Switch

Path Selection

• Crossbar fabric
  – simple path routing
  • element \((o, i)\) turns
  • \(i_3 \rightarrow o_4\)
Crossbar Switch
Strictly Nonblocking

- Crossbar fabric
  - simple path routing
    - element \((o, i)\) turns
    - \(i_3 \rightarrow o_4\)
  - strictly nonblocking
    - \(i_j \rightarrow o_n\) noblock \(i_k \rightarrow o_m\)
    - \(\forall j, k, n, m: i \neq j, n \neq m\)
    - \(i_1 \rightarrow o_1\)
Crossbar Switch
Strictly Nonblocking

- Crossbar fabric
  - simple path routing
    - element \((o, i)\) turns
    - \(i_3 \rightarrow o_4\)
  - strictly nonblocking
    - \(i_j \rightarrow o_n\) noblock \(i_k \rightarrow o_m\)
    - \(\forall j, k, n, m: i \neq j, n \neq m\)
    - \(i_1 \rightarrow o_1\)
Crossbar Switch
Strictly Nonblocking

• Crossbar fabric
  – simple path routing
    • element \((o,i)\) turns
    • \(i_3 \rightarrow o_4\)
  – strictly nonblocking
    • \(i_j \rightarrow o_n\) noblock \(i_k \rightarrow o_m\)
    \(\forall j,k,n,m: i \neq j, n \neq m\)
    • \(i_1 \rightarrow o_1\)
Switch Fabrics
Multistage Switches

- Large switches built from single stage elements
  - $2 \times 2$ elements or $n \times n$ crossbars
  - $O(n \log n)$ growth complexity
Multistage Switch Fabrics
Delta Fabric Construction and Scalability

- Delta fabric
  - $O(n \log n)$
    - $n/2$ rows
    - $\log_2 n$ stages
    - $n = 2$
    - $2/2 \log_2 2 = 1$
Multistage Switch Fabrics
Delta Fabric Construction and Scalability

- Delta fabric
  - $O(n \log n)$
    - $n/2$ rows
    - $\log_2 n$ stages
    - $n = 4$
    - $4/2 \log_2 4 = 4$
Multistage Switch Fabrics
Delta Fabric Construction and Scalability

- Delta fabric
  - $O(n \log n)$
  - $n/2$ rows
  - $\log_2 n$ stages
  - $n = 8$
  - $8/2 \log_2 8 = 12$
Multistage Switch Fabrics
Delta Fabric Construction and Scalability

- Delta fabric
  - $O(n \log n)$
    - $n/2$ rows
    - $\log_2 n$ stages
    - $n = 16$
    - $16/2 \log_2 16 = 32$
Multistage Switch Fabrics
Delta Fabric Construction Self-Routing

- Delta fabric
  - self-routing
  - $i^{th}$ bit of $p_{out}$ used to make routing decision in $i^{th}$ stage
**Multistage Switch Fabrics**

**Delta Fabric Construction Self-Routing**

- **Delta fabric**
  - self-routing
  - $i^{th}$ bit of $p_{out}$ used to make routing decision in $i^{th}$ stage
  - $i_2 \rightarrow o_{10}$
Multi-stage Switch Fabrics

Delta Fabric Construction Self-Routing

- Delta fabric
  - self-routing
  - \( i^{th} \) bit of \( p_{out} \) used to make routing decision in \( i^{th} \) stage
  - \( i_2 \rightarrow o_{10} \)
Multistage Switch Fabrics
Delta Fabric Construction Self-Routing

- Delta fabric
  - self-routing
  - i\textsuperscript{th} bit of $p_{\text{out}}$ used to make routing decision in i\textsuperscript{th} stage
  - $i_2 \rightarrow o_{10}$
Multistage Switch Fabrics
Delta Fabric Construction Self-Routing

- Delta fabric
  - self-routing
  - $i^{th}$ bit of $p_{out}$ used to make routing decision in $i^{th}$ stage
  - $i_2 \rightarrow o_{10}$
Multistage Switch Fabrics
Delta Fabric Construction Self-Routing

- Delta fabric
  - self-routing
  - $i^{th}$ bit of $p_{out}$ used to make routing decision in $i^{th}$ stage
  - $i_2 \rightarrow o_{10}$
Multistage Switch Fabrics
Delta Fabric Construction Self-Routing

- Delta fabric
  - self-routing
  - $i^{th}$ bit of $p_{out}$ used to make routing decision in $i^{th}$ stage
  - $i_2 \rightarrow o_{10}$
Multistage Switch Fabrics

Delta Fabric Construction Self-Routing

- **Delta fabric**
  - self-routing
  - $i^{th}$ bit of $p_{out}$ used to make routing decision in $i^{th}$ stage
  - $i_2 \rightarrow o_{10}$
Multistage Switch Fabrics

Delta Fabric Construction Self-Routing

- Delta fabric
  - self-routing
  - $i^{th}$ bit of $p_{out}$ used to make routing decision in $i^{th}$ stage
  - $i_2 \rightarrow o_{10}$
Multistage Switch Fabrics
Delta Fabric Construction Self-Routing

- **Delta fabric**
  - self-routing
  - $i^\text{th}$ bit of $p_{\text{out}}$ used to make routing decision in $i^\text{th}$ stage
  - $i_2 \rightarrow o_{10}$
Multistage Switch Fabrics
Delta Fabric Construction Self-Routing

- Delta fabric
  - self-routing
  - $i^\text{th}$ bit of $p_{\text{out}}$ used to make routing decision in $i^\text{th}$ stage
  - $i_2 \rightarrow o_{10}$
Multistage Switch Fabrics
Delta Fabric Construction Self-Routing

- Delta fabric
  - self-routing
  - $i^{th}$ bit of $p_{out}$ used to make routing decision in $i^{th}$ stage
    - $i_2 \rightarrow o_{10}$
    - $i_{13} \rightarrow o_{10}$
Multistage Switch Fabrics
Delta Fabric Construction Self-Routing

- Delta fabric
  - self-routing
  - $i^{th}$ bit of $p_{out}$ used to make routing decision in $i^{th}$ stage
  - $i_2 \rightarrow o_{10}$
  - $i_{13} \rightarrow o_{10}$
Multistage Switch Fabrics
Delta Fabric Construction Self-Routing

• Delta fabric
  – self-routing
  – $i^{th}$ bit of $p_{out}$ used to make routing decision in $i^{th}$ stage
  – $i_2 \rightarrow o_{10}$
  – $i_{13} \rightarrow o_{10}$

![Diagram of multistage switch fabrics with delta fabric construction self-routing](image-url)
Multistage Switch Fabrics
Delta Fabric Construction Self-Routing

- Delta fabric
  - self-routing
  - $i^{th}$ bit of $p_{out}$ used to make routing decision in $i^{th}$ stage
    - $i_2 \rightarrow o_{10}$
    - $i_{13} \rightarrow o_{10}$
Multistage Switch Fabrics
Delta Fabric Construction Self-Routing

- Delta fabric
  - self-routing
  - $i^{th}$ bit of $p_{out}$ used to make routing decision in $i^{th}$ stage
  - $i_2 \rightarrow o_{10}$
  - $i_{13} \rightarrow o_{10}$
**Multistage Switch Fabrics**

**Delta Fabric Construction Self-Routing**

- Delta fabric
  - Self-routing
  - $i^{th}$ bit of $p_{out}$ used to make routing decision in $i^{th}$ stage
  - $i_2 \Rightarrow o_{10}$
  - $i_{13} \Rightarrow o_{10}$
Multistage Switch Fabrics
Delta Fabric Construction Self-Routing

- Delta fabric
  - self-routing
  - $i^{th}$ bit of $p_{out}$ used to make routing decision in $i^{th}$ stage
    - $i_2 \rightarrow o_{10}$
    - $i_{13} \rightarrow o_{10}$
Multistage Switch Fabrics
Delta Fabric Construction Self-Routing

- Delta fabric
  - self-routing
  - $i^{th}$ bit of $p_{out}$ used
    to make routing decision in $i^{th}$ stage
    - $i_2 \rightarrow o_{10}$
    - $i_{13} \rightarrow o_{10}$
Multistage Switch Fabrics
Delta Fabric Construction Self-Routing

- Delta fabric
  - self-routing
  - $i^\text{th}$ bit of $p_{out}$ used to make routing decision in $i^\text{th}$ stage
    - $i_2 \rightarrow o_{10}$
    - $i_{13} \rightarrow o_{10}$
Multistage Switch Fabrics
Clos Fabric Overview

- **Clos fabric**
  - $n \times n$ fabric is interconnection of 3 crossbars stages
  - input stage of $n/d$ elements:
    - $d \times r$ crossbars each with $d : r$ expansion
  - middle stage of $r$ elements: $n/d \times n/d$ square crossbars
  - output stage of $n/d$ elements:
    - $r \times d$ crossbars each with $r : d$ concentration
  - perfect shuffle interconnection between stages:
    - $i^{th}$ output of $j^{th}$ crossbar $\rightarrow$ $j^{th}$ input of $i^{th}$ crossbar
Multistage Switch Fabrics
Clos Fabric Architecture and Construction

\[
\begin{align*}
\begin{array}{c}
\text{0} \\
\text{d} \\
\text{d \times r} \\
\text{d < r} \\
\text{r} \\
\text{r \times n/d} \\
\end{array}
\end{align*}
\]

\[
\begin{align*}
\begin{array}{c}
\text{0} \\
\text{n/d \times n/d} \\
\text{1} \\
\text{r - 1} \\
\end{array}
\end{align*}
\]

\[
\begin{align*}
\begin{array}{c}
\text{0} \\
\text{r \times d} \\
\text{r > d} \\
\text{n/d - 1} \\
\end{array}
\end{align*}
\]
Multistage Switch Fabrics
Clos Fabric Architecture and Construction
Multistage Switch Fabrics
Clos Fabric Architecture and Construction
Multistage Switch Fabrics
Clos Fabric Architecture and Construction
Multistage Switch Fabrics
Clos Fabric Architecture and Construction
Multistage Switch Fabrics
Clos Fabric Architecture and Construction
Multistage Switch Fabrics
Clos Fabric Architecture and Construction
Multistage Switch Fabrics
Clos Fabric Architecture and Construction
Multistage Switch Fabrics
Clos Fabric Architecture and Construction
Multistage Switch Fabrics
Clos Fabric Architecture and Construction
Multistage Switch Fabrics
Clos Fabric Architecture and Construction
Multistage Switch Fabrics
Clos Fabric Architecture and Construction
Multistage Switch Fabrics
Clos Fabric Architecture and Construction
Multistage Switch Fabrics
Clos Fabric Architecture and Construction
Multistage Switch Fabrics
Clos Fabric Architecture and Construction
Multistage Switch Fabrics
Clos Fabric Architecture and Construction
Multistage Switch Fabrics
Clos Fabric Architecture and Construction
Multistage Switch Fabrics
Clos Fabric Architecture and Construction
Multistage Switch Fabrics
Clos Fabric Architecture and Construction
Multistage Switch Fabrics
Clos Fabric Routing and Blocking

• Clos fabric path routing
  – for input from \( i \)th crossbar in first stage \( \rightarrow \)
    \( j \)th crossbar in last stage
  – choose any middle stage crossbar with \((i, j)\) point available

• Blocking characteristics
  – blocking depends on expansion in middle stages \( r \)
  – strictly nonblocking for unicast iff \( r \geq 2d-1 \)
  – engineered to balance cost against blocking probability
Multistage Switch Fabrics

Clos Fabric Routing
Multistage Switch Fabrics

Clos Fabric Routing
Multistage Switch Fabrics
Clos Fabric Routing
Multistage Switch Fabrics

Clos Fabric Routing
Multistage Switch Fabrics

Clos Fabric Routing
Multistage Switch Fabrics

Clos Fabric Application

- **Clos fabric**
  - used in modern switch fabrics
  - e.g. Ciena optical switches, Juniper routers

- **Engineering tradeoff optimisations**
  - crossbar elements to VLSI switch complexity
  - partitioning: 1st and last stages can be on line-group cards
    - permits switching among different rate interfaces
Regular Networks

References and Further Reading


End of Foils