

# Arquiteturas Paralelas: redes de conexão

Paulo Sérgio Lopes de Souza  
*pssouza@icmc.usp.br*

Universidade de São Paulo / ICMC / SSC – São Carlos  
Laboratório de Sistemas Distribuídos e Programação Concorrente

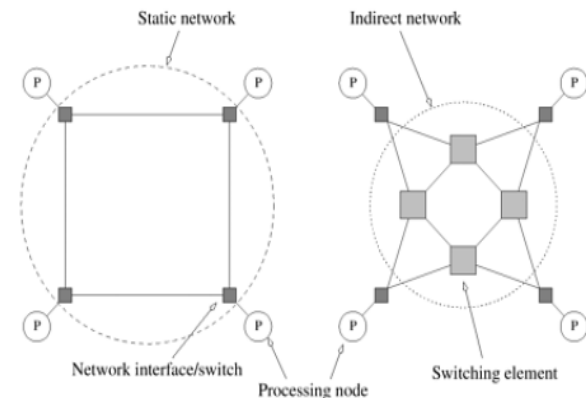
# Redes de Conexão: conceitos básicos

- Redes de conexão permitem transferências de dados e sinais de controle entre:
  - Computadores, processadores, memórias, E/S e outros dispositivos
- São vitais para desempenho de aplicações concorrentes
  - Conceito de programa armazenado força vários acessos à memória
    - Afeta granulação das porções paralelas
  - Espera-se que transmitam mensagens
    - Corretamente e tão rápido quanto possível

# Redes de Conexão: conceitos básicos

- Podem ser representadas por grafos
  - Nós são os dispositivos conectados
    - Computadores, CPUs, Memórias, E/S, switches, ...
    - Possuem N entradas e M saídas
  - Arestas são os links de comunicação
    - Meio físico (fios/fibras) capazes de transportar informações
- Switches
  - Permitem a conexão dinâmica de dispositivos
    - Mapeamento dinâmico de mensagens das portas de E para S
  - Podem usar crossbars físicos, memórias (buffers temporários), multiplexadores, barramentos,...

Figure 2.6. Classification of interconnection networks: (a) a static network; and (b) a dynamic network.



- Interfaces
  - Conectividade entre nós e a rede
  - Resp por (des)empacotamento  
Inf para roteamento,  
Detecção/correção de erros  
...

# Redes de Conexão: principais propriedades

- Topologia: como dispositivos e links são organizados
  - Dinâmica & Estática
- Diâmetro da rede: distância máxima entre quaisquer dois nós da rede
  - Opta-se pelo menor caminho, caso haja redundância
  - Determina o maior atraso entre dois nós
- Grau do nó: número de links, ou de entrada ou de saída
- Latência: atraso na transferência entre dois nós da rede
  - Latência de software e de hardware
- Conectividade de nó/aresta:
  - Nr de elementos que precisam falhar para que a rede fique desconexa
  - Quanto maior, maior é a tolerância à desconexão na rede
- Largura do canal: determina quantos bits são enviados simultaneamente
- Custo do hardware de rede: há várias métricas como nr de links, nr de fios, nr de switches, interfaces, ...

# Redes de Conexão: principais propriedades

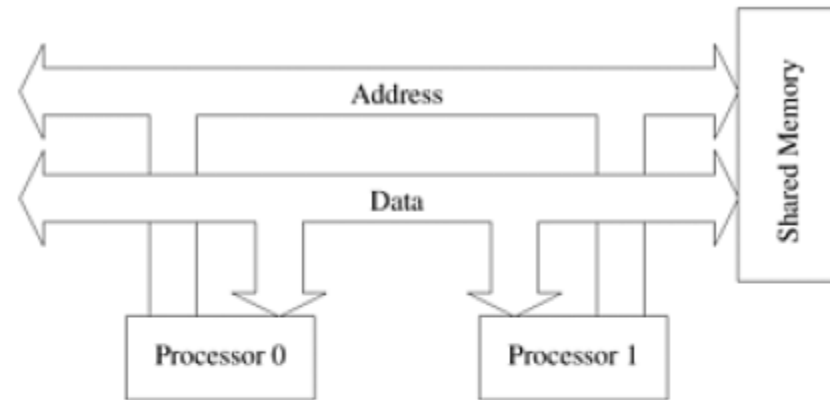
- *Throughput* da rede: capacidade total de transmissão por unidade de tempo
- Roteamento: estabelece um caminho entre nós origem e destino
  - Determina qual canal de saída deve ser usado
  - Balanceamento da carga sobre os links e nós
  - Evitam *deadlocks* quando mensagens são bloqueantes
- Chaveamento: determina como uma msg é transmitida pelo caminho já determinado
  - Por circuito: reserva o caminho todo previamente
  - Por pacotes: msg dividida em pacotes, transmitidos independentemente
    - Pctes podem usar caminhos diferentes e chegar fora da ordem
    - *Store-and-forward*: recebe pacotes, armazena e os envia
    - *Cut-Through*: envia pacotes assim que os recebe, formando um pipeline
- Bloqueantes & Não bloqueantes
  - Na não bloqueante nós livres podem transmitir em paralelo a outras msgs
- Dimensão da Rede: arranjo do nós na rede: linear (1D), *mesh*/malha (2D), ...
- *Broadcast & Multicast*: envio a todos & um grupo de nós na rede

# Topologias em Redes de Conexão

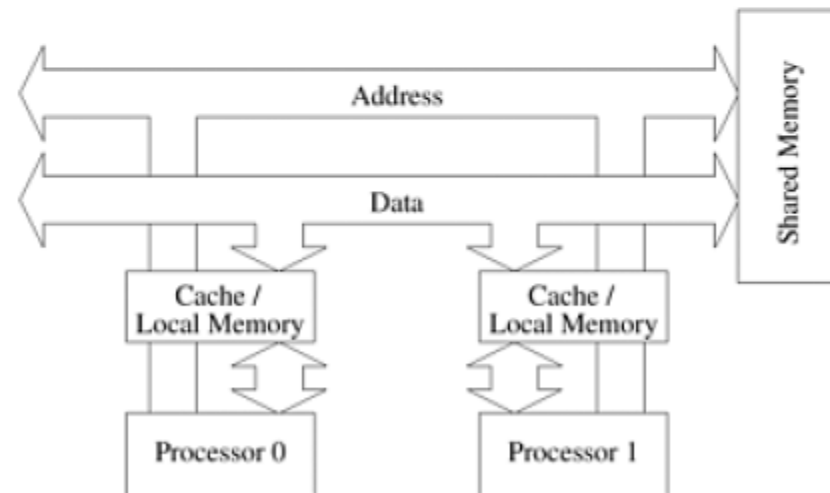
- Diferentes topologias visam:
  - Reduzir custo da rede
  - Fornecer escalabilidade
  - Aumentar desempenho
- Desejam-se alguns aspectos como:
  - Diâmetro pequeno para ter pequenas distâncias nas transmissões
  - Nós com graus pequenos reduzem sobrecarga de hardware nos nós
  - Latências menores
  - *Throughputs* maiores
  - Alta conectividade
  - Facilidade de extensão
- Objetivos podem ser conflitantes
  - Latência vs throughput, graus menores de nós vs alta conectividade

# Topologias Dinâmicas: redes em barramento

**Figure 2.7. Bus-based interconnects (a) with no local caches; (b) with local memory/caches.**



(a)



(b)

# Topologias Dinâmicas: redes *crossbar*

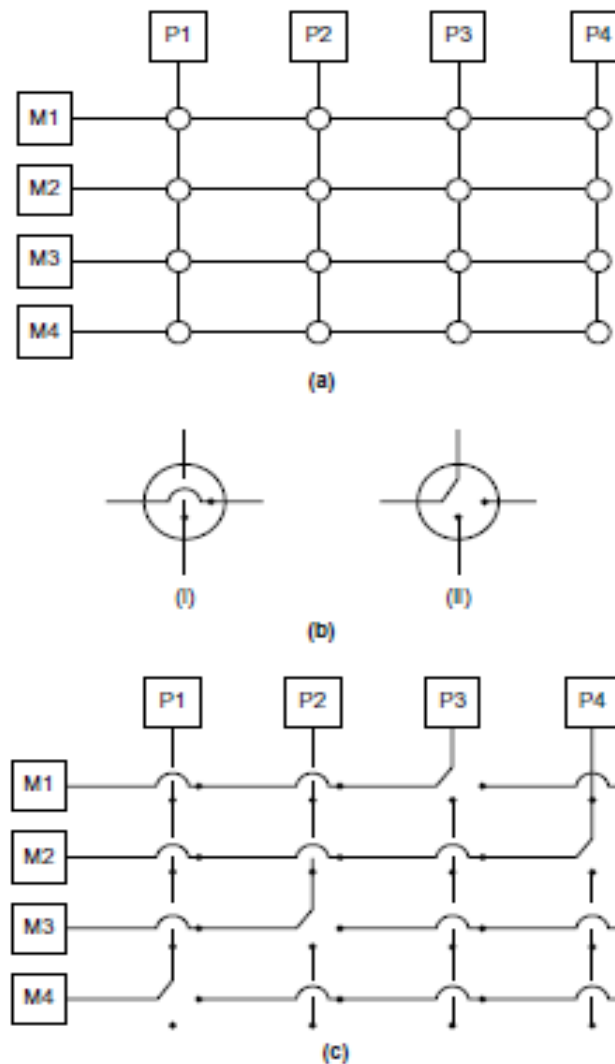


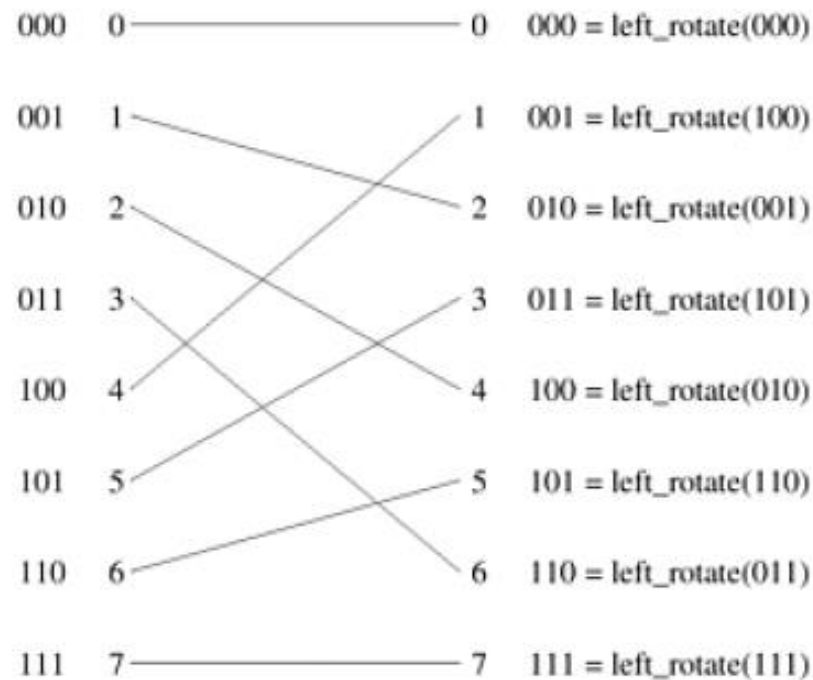
FIGURE 2.7

(a) A crossbar switch connecting four processors ( $P_i$ ) and four memory modules ( $M_j$ ); (b) configuration of internal switches in a crossbar; (c) simultaneous memory accesses by the processors



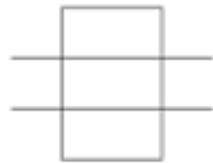
# Topologias Dinâmicas: redes multi-estágio (ômega)

**Figure 2.10. A perfect shuffle interconnection for eight inputs and outputs.**

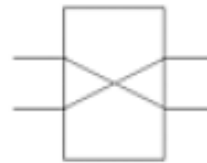


# Topologias Dinâmicas: redes multi-estágio (ômega)

**Figure 2.11. Two switching configurations of the 2 x 2 switch: (a) Pass-through; (b) Cross-over.**



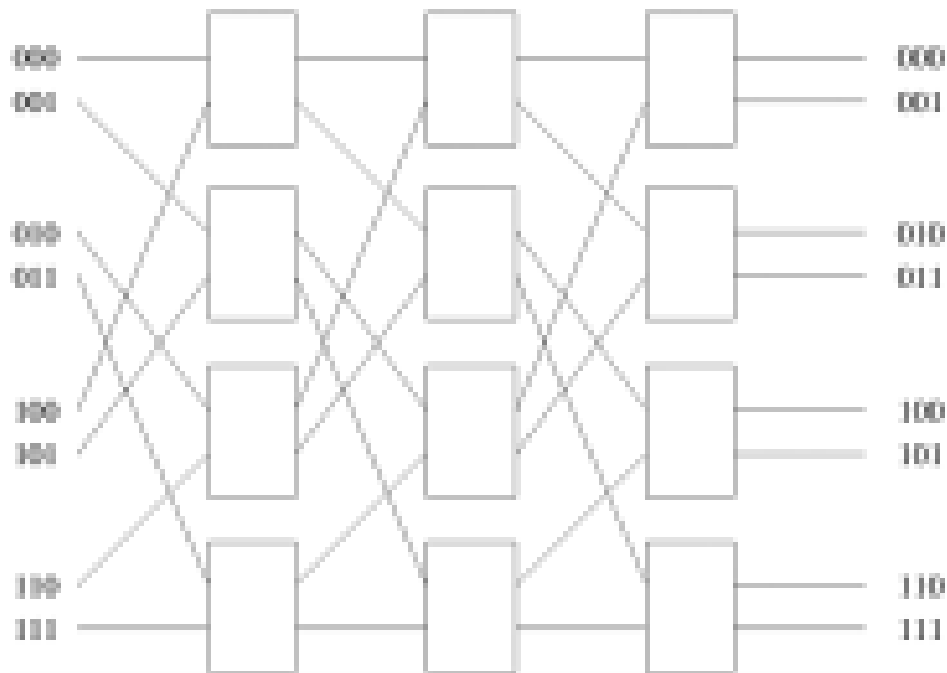
(a)



(b)

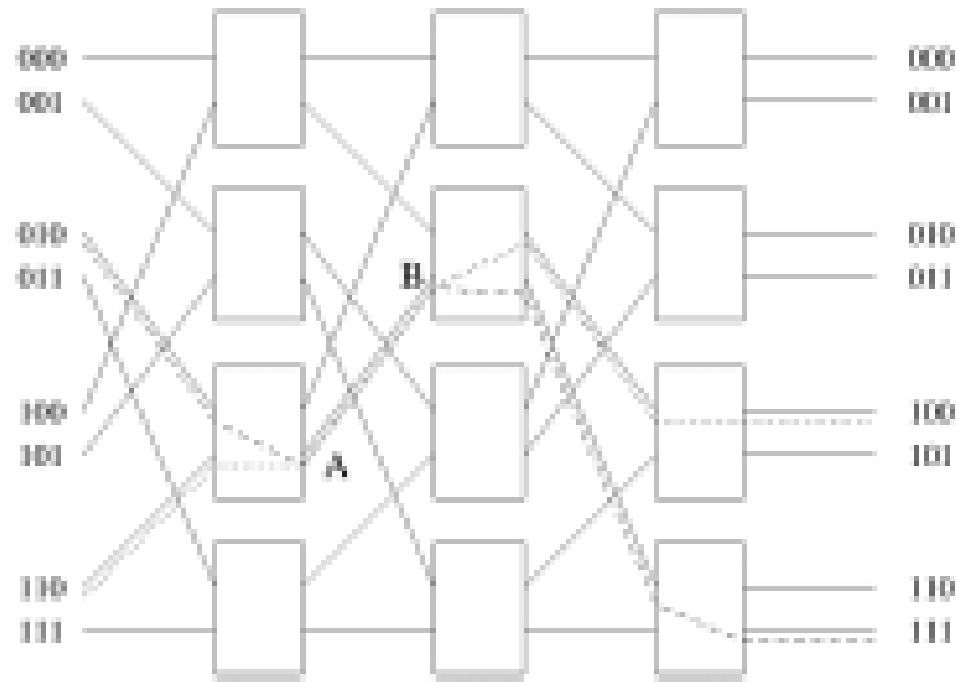
## Topologias Dinâmicas: redes multi-estágio (ômega)

**Figure 2.12. A complete omega network connecting eight inputs and eight outputs.**



## Topologias Dinâmicas: redes ômega são bloqueantes

**Figure 2.13. An example of blocking in omega network: one of the messages (010 to 111 or 110 to 100) is blocked at link AB.**



## Topologias Dinâmicas: redes *butterfly*

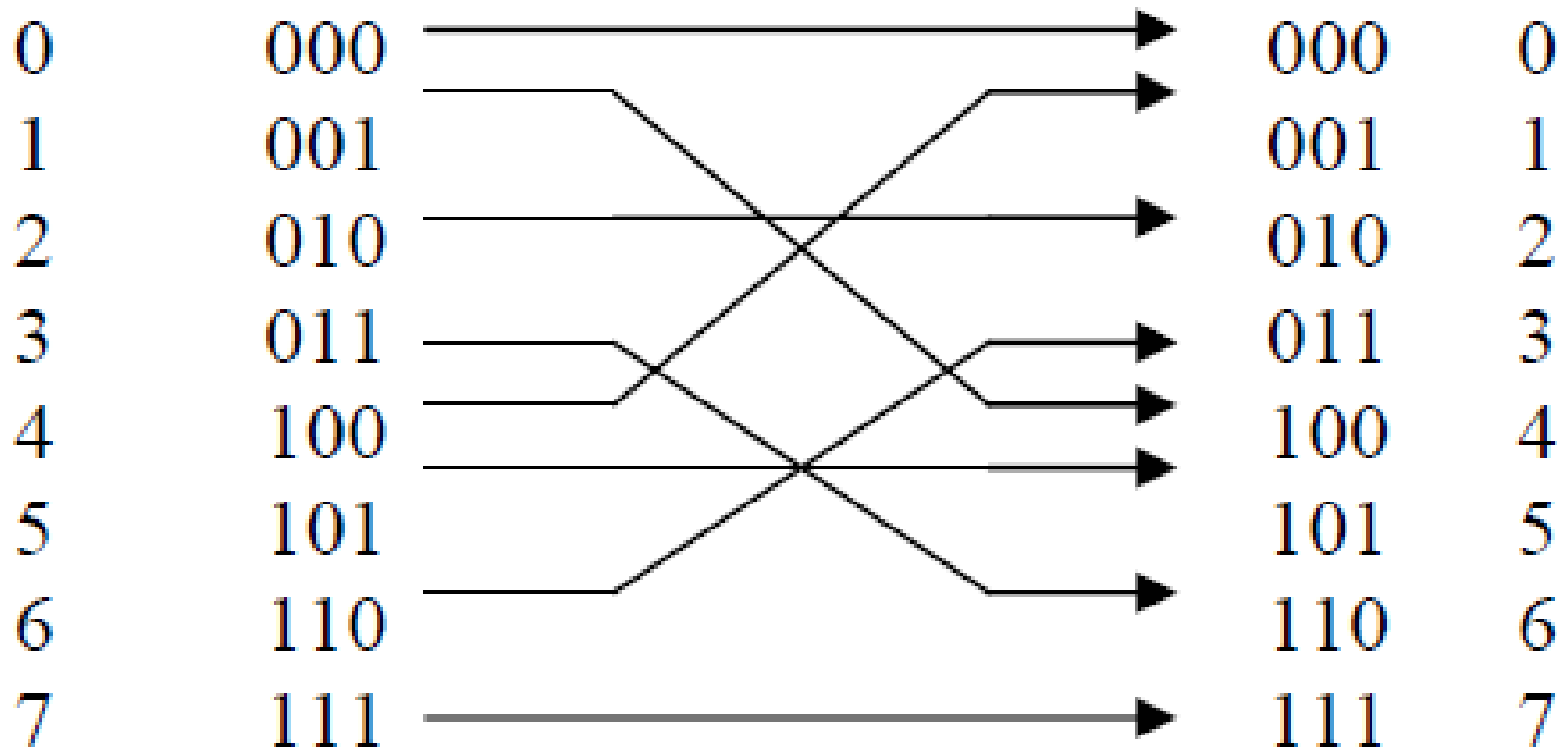
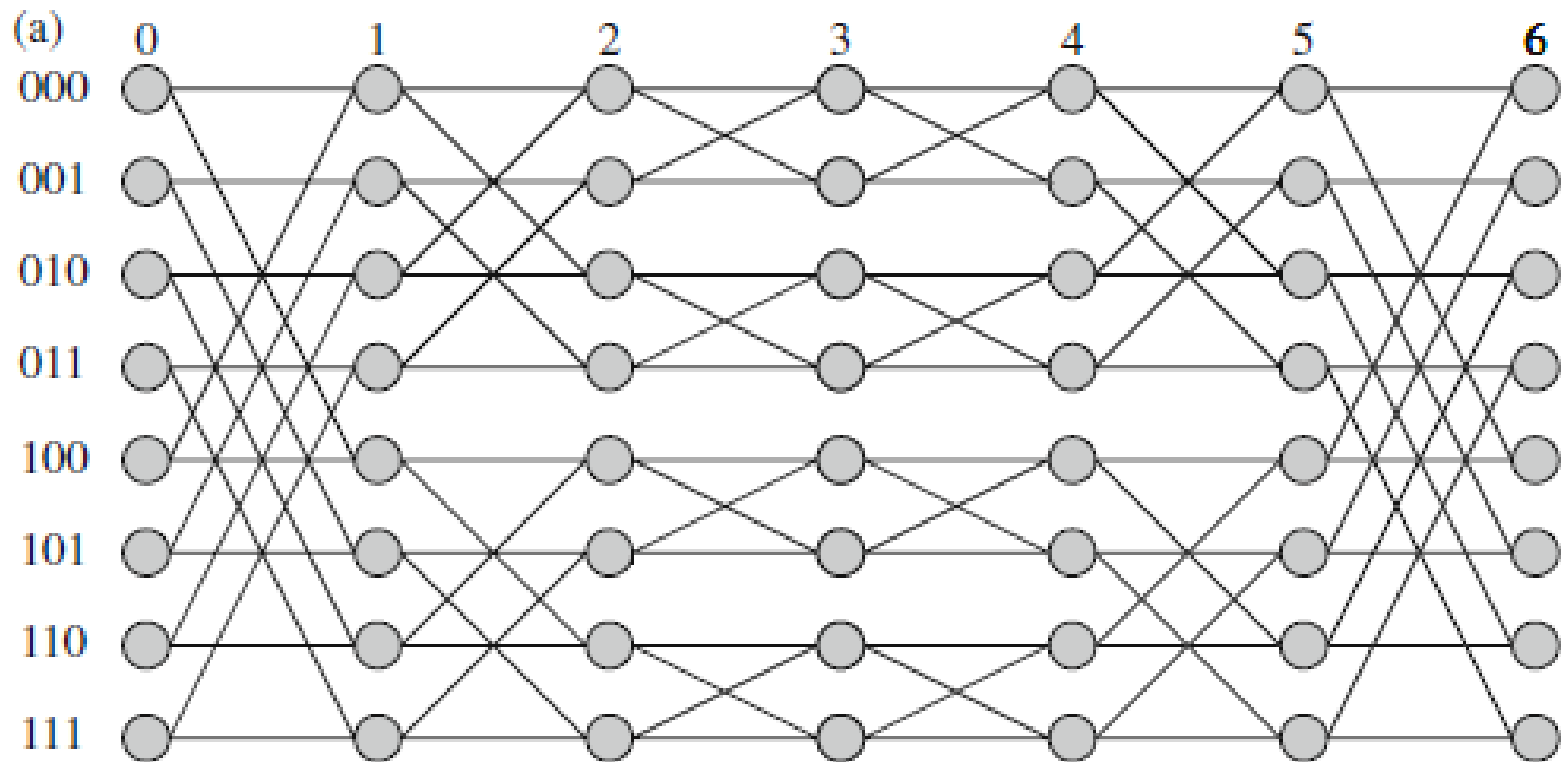


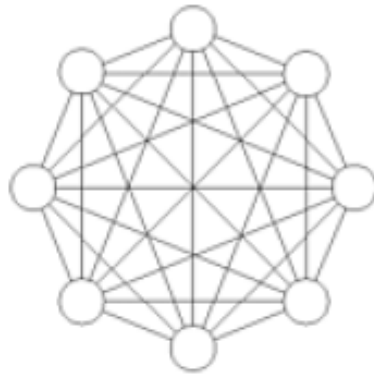
Figure 18: Butterfly permutation

## Topologias Dinâmicas: *butterfly* - *benes*



## Topologias Estáticas: totalmente conectada e estrela

**Figure 2.14. (a) A completely-connected network of eight nodes;  
(b) a star connected network of nine nodes.**



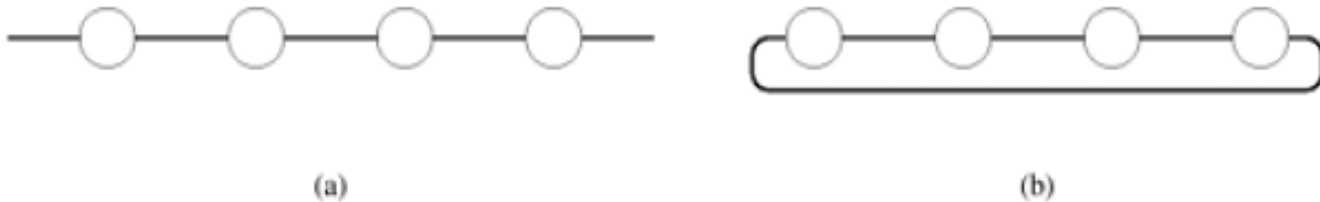
(a)



(b)

## Topologias Estáticas: linha e anel

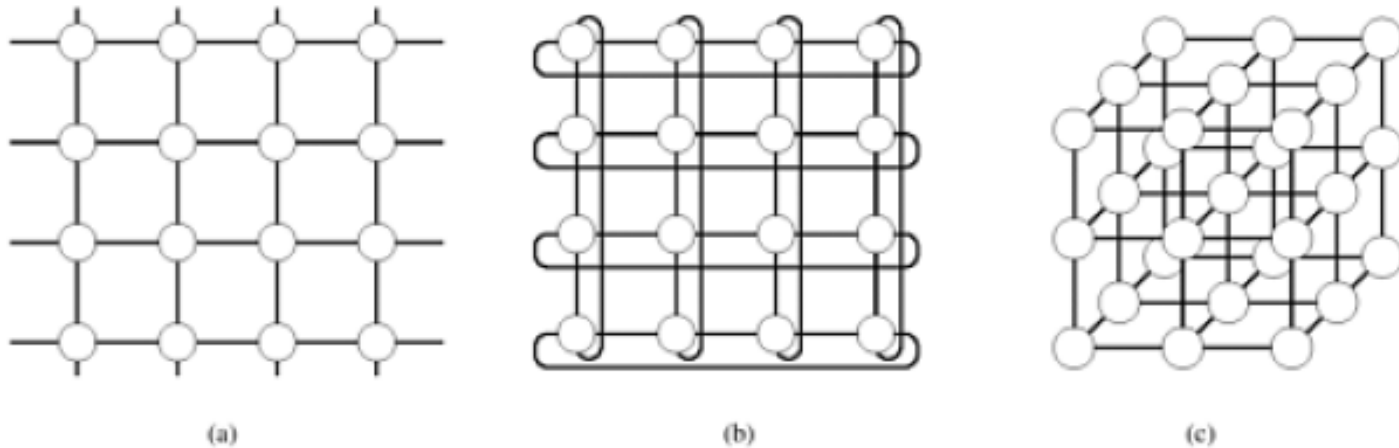
**Figure 2.15. Linear arrays: (a) with no wraparound links; (b) with wraparound link.**





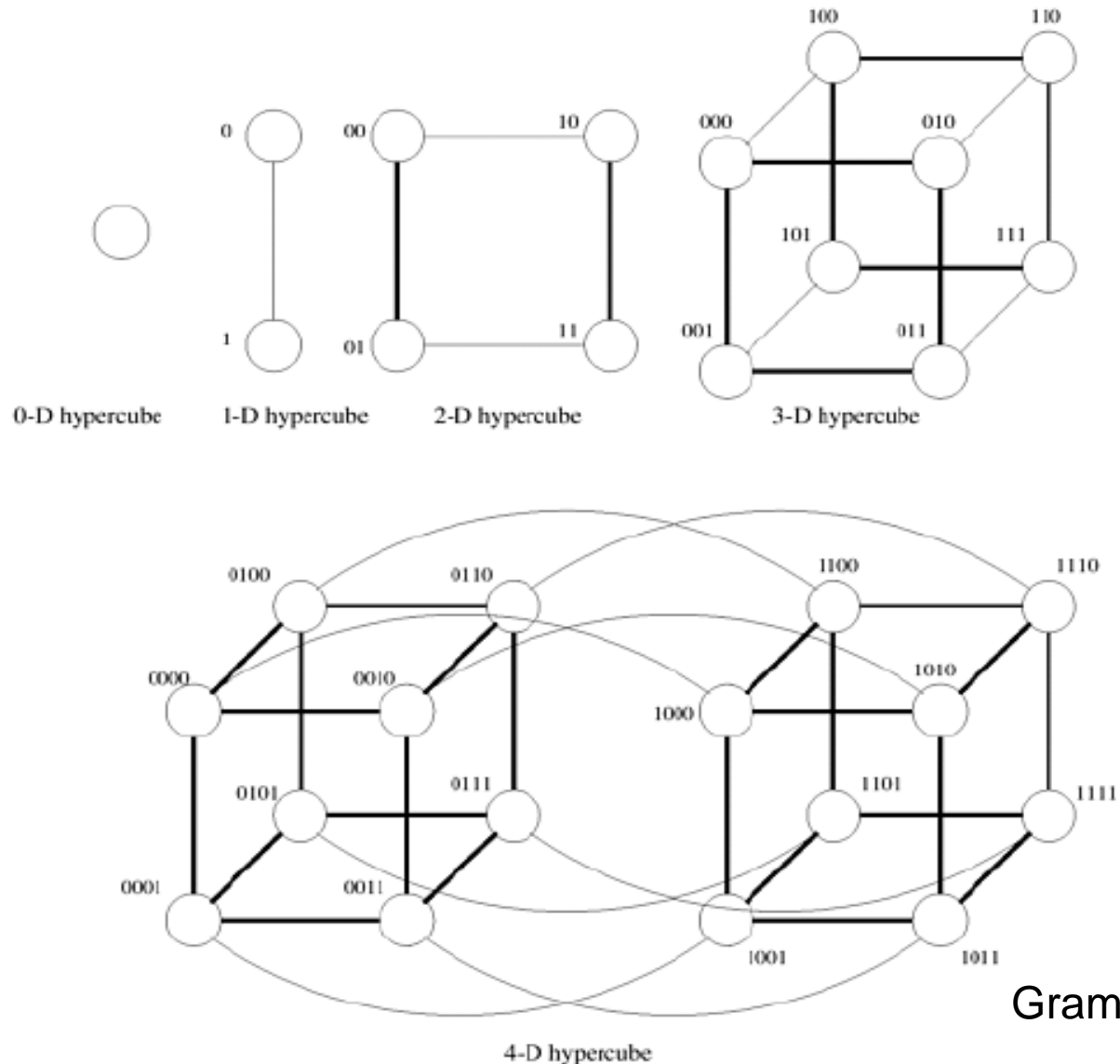
## Topologias Estáticas: malha 2D e torus 2D/3D

**Figure 2.16. Two and three dimensional meshes: (a) 2-D mesh with no wraparound; (b) 2-D mesh with wraparound link (2-D torus); and (c) a 3-D mesh with no wraparound.**



# Topologias Estáticas: cubo e hipercubo

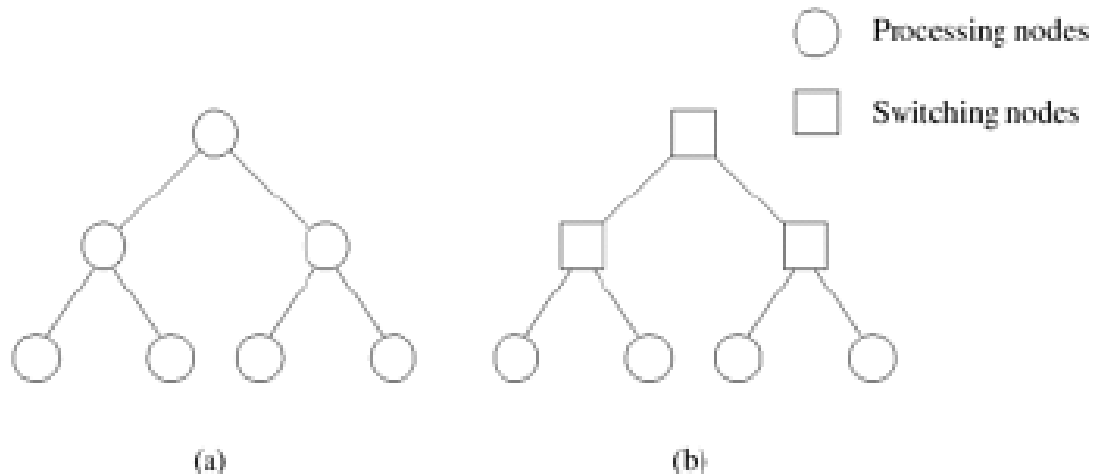
**Figure 2.17. Construction of hypercubes from hypercubes of lower dimension.**



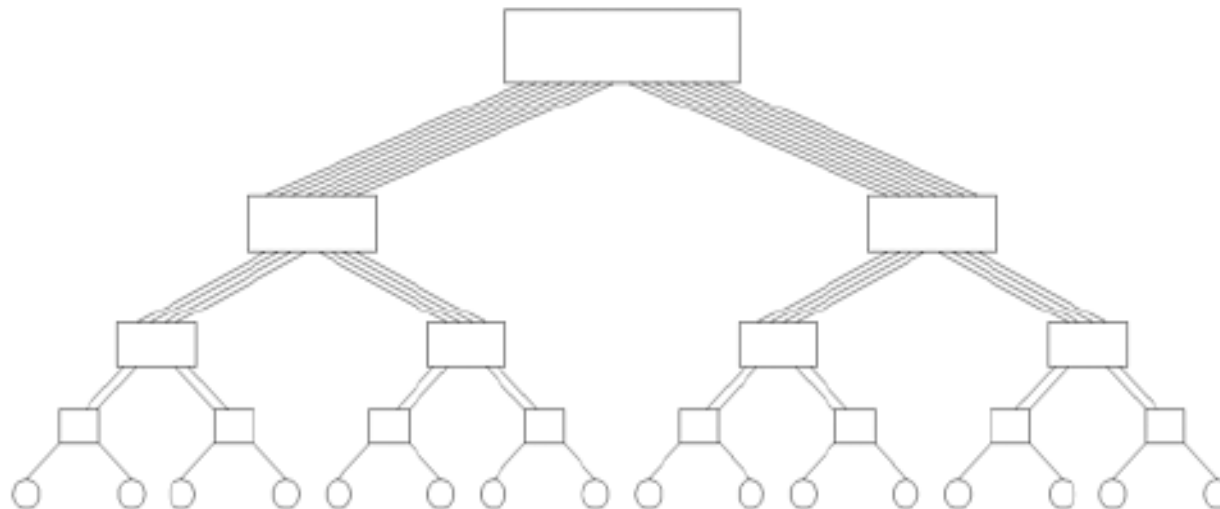
Gramma et al. (2003)

# Topologias Din/Est: redes baseadas em árvores

**Figure 2.18. Complete binary tree networks: (a) a static tree network; and (b) a dynamic tree network.**



**Figure 2.19. A fat tree network of 16 processing nodes.**



Grama et al. (2003)

# Referências

GRAMA,A.; KUMAR, U.; GUPTA,A.; KARYPIS, G. Introduction to Parallel Computing, 2nd Edition, 2003.

Stallings, W.; Computer Organization and Architecture: Designing for Performance. Ninth Edition. Pearson. 2013.

Tanenbaum, A. S.; Austin, T.; Structured Computer Organization. Sixth Edition. Pearson. 2013.

Patterson, D. A.; Hennessy, J. L.; Computer Organization and Design: the hardware / software interface. Fifth Edition. Elsevier, 2014.

Rauber, T.; Rünger, G.; Parallel Programming for Multicore and Cluster Systems. Second Edition. Springer. 2013.



# Arquiteturas Paralelas: redes de conexão

Paulo Sérgio Lopes de Souza  
*pssouza@icmc.usp.br*

Universidade de São Paulo / ICMC / SSC – São Carlos  
Laboratório de Sistemas Distribuídos e Programação Concorrente

THAT'S THE END OF  
PRESENTATION



meksememe.org