Cost Performance Efficient Interconnection Networks



Yamin Li, Graduate School of CIS, Hosei University, yamin@hosei.ac.jp

Outline

- Top 10 supercomputers
- Multiprocessor shared-memory parallel systems
- Multicomputer distributed systems
- Problems of interconnection network (IN)
- Low-degree and short-diameter static INs
 - Dual-Cube
 - Metacube and KMS-Cube
 - RDN Recursive Dual-Net
- Dynamic IN <u>MiKANT</u>
 Dynamic IN <u>Peer Fat-Tree</u>
- Summary and exercise

Top 10 Supercomputers

TOP500 Supercomputer Sites:

https://www.top500.org/

TOP 10 Supercomputers in June 2025

Rank	System	Country	Maker	Cores	PFlop/s
1	El Capitan	USA	HPE	11,039,616	1,742.00
2	Frontier	USA	HPE	9,066,176	1,353.00
3	Aurora	USA	Intel	9,264,128	1,012.00
4	Jupiter Booster	Germany	EVIDEN	4,801,344	793.40
5	Eagle	USA	Microsoft	2,073,600	561.20
6	HPC6	Italy	HPE	3,143,520	477.90
7	Fugaku	Japan	Fujitsu	7,630,848	442.01
8	Alps	Switzerland	HPE	2,121,600	434.90
9	LUMI	Finland	HPE	2,752,704	379.70
10	Leonardo	Italy	EVIDEN	1,824,768	241.20

Parallel and Distributed Systems

- In general, parallel systems are referred to as shared-memory multiprocessors.
 - SMP Symmetric multiprocessors
 - Uniform memory access (UMA)
 - Example: Servers
 - DSM Distributed shared memory
 - Non-uniform memory access (NUMA)
 - Example: Supercomputers
- In contrast, distributed systems are referred to as message-passing multicomputers.
 - Cannot access other computer's memory directly
 Example: Infrastructure of Cloud Computing

Symmetric Multiprocessors (Servers)



Distributed Shared Memory (Supercomputers)



Interconnection Networks



An Implementation of Switch



A Switch with Self-Routing Function



It can support wormhole routing.

Properties of Interconnection Networks

Degree

- Two nodes are neighbors if there is a link connecting them.
- The degree of a node is defined to be the number of its neighbors.
- Affect hardware cost.

Diameter

- The diameter of a network is defined as the maximum of the shortest distances between any two nodes.
- Affect communication time.

Fat-tree Almost all systems Torus Fujitsu IBM HPE/Cray Dragonfly HPE/Cray Hypercube SG Intel

Ring and Completely Connected Networks



(a) Ring

(b) Completely connected

Symmetric

Mesh Interconnection Networks



Not symmetric

Torus Interconnection Networks



Symmetric

Hypercube Interconnection Networks



Symmetric

Tree Interconnection Networks



(c) Fat-tree

K-ary N-tree Interconnection Networks



Dragonfly Interconnection Networks



$$CR(G) = \frac{d(G) \times w_1 + D(G) \times w_2}{\log_2|(G)|}$$

d(G): the node degree of G (the number of links of a node) D(G): the diameter of G |(G)|: the total number of nodes in G w_1 and w_2 : weights, $w_1 + w_2 = 100\%$

INs	Nodes	Degree	Diameter	CR
3D-Torus (10)	1,000	6	15	1.05
10-cube	1,024	10	10	1.00
3D-Torus (128)	2,097,152	6	192	4.72
21-cube	2,097,152	21	21	1.00

 $w_1 = w_2 = 0.5$

Problems of Hypercube, 3D Torus, and Tree

- Hypercube problem
 - The number of links increases logarithmically as the number of nodes increases: n = log₂ N
- 3D Torus problem
 - Large (long) diameter: D = [X/2] + [Y/2] + [Z/2]
- Tree problem
 - Not symmetric

Problems of Interconnection Networks

- Suppose a supercomputer has 2²¹ = 2,097,152 nodes
- By Hypercube
 - Diameter = 21
 - Node-degree = 21 (high cost)
- By 128 × 128 × 128 Torus
 - Diameter = [128/2] × 3 = 192 (too large)
 - Node-degree = 6

SGI Origin2000's Solution

A router has <mark>six</mark> links Two links connect to CPU boards

Offen 20

Origin2000 of 3D and 4D



An Origin2000 router has six links
Two links connect to two CPU boards
A board contains two CPUs

Cray Router in 5D Origin2000



WWW — What We Want



Low node degree

and



Short diameter

These are conflicting!

Cube-Connected Cycles



Node degree === 3

Hierarchical Cubic Network



Each node (X, Y) is adjacent to

- 1. $(X, Y^{(k)})$ for all $1 \le k \le n$, where $Y^{(k)}$ differs from Y at the kth bit position,
- 2. (Y, X) if $X \neq Y$, and
- 3. $(\overline{X}, \overline{Y})$ if X = Y, where \overline{X} and \overline{Y} are the bitwise complements of X and Y, respectively.

Dual-Cube

Dual-Cube DC(m) Interconnection Network

- Node degree: m + 1
- Can connect N = 2^{2m+1} nodes
- Keeps the main properties of hypercube
- Simple routing algorithm
- Is Hamiltonian
- Performs collective communications efficiently
- Low communication cost for matrix multiplication
- Easy to build disjoint paths
- Maximum length of fault-free cycle embedding
- Efficient fault-tolerant routing

Node Address Format of Dual-Cube DC(m)



Each node has (m + 1) links

- The m links in Node ID builds a cluster (m-cube)
- One link in Class ID connects to a node in a cluster of the other class
- No links in Cluster ID
- A DC(m) can connect 2^{2m+1} nodes

A Dual-Cube DC(2)



A Dual-Cube DC(3)



Building Origin2000 with Dual-Cube



It can connect $16 \times 8 \times 2 \times 2 = 512$ CPUs



Node Address Format of Metacube MC(k, m)



class_id : c

node_id : m_c

cluster_id : m_{2k-1}, ..., m_{c+1}, m_{c-1}, ..., m₀

- An MC(k, m) can connect 2^{m2^k+k} nodes
- Each node has (m + k) links
- Links in c field form high-level k-cubes
- Links in m_c field form low-level m-cubes

A Metacube MC(2,2)


An Alternative Address Format



Address of k-Cube Oriented Metacube

A k-Cube Oriented Metacube MC(1,2)



A k-Cube Oriented Metacube MC(2,1)



Number of Nodes in Metacubes

An MC(k, m) can connect $2^{m2^{k}+k}$ nodes. Degree: m + k

Links/node	3	4	5	6	7	8
Hypercube	8	16	32	64	128	265
MC(1, m)	32	128	512	2,048	8,192	32,768
MC(2, m)	64	1,024	16,384	2 ¹⁸	2 ²²	2 ²⁶
MC(3, m)		2,048	2 ¹⁹	2 ²⁷	2 ³⁵	243
MC(4, m)			2 ²⁰	2 ³⁶	2 ⁵²	2 ⁶⁸

2²⁷ = 134,217,728

16,384 Nodes: 14-Cube vs MC(2,3)

Hypercube

Links: 14 links per node 2¹⁴ × 14/2 = 114,688 links in total Diameters: 14

Metacube

- Links: 5 links per node 2¹⁴ × (3 + 2)/2 = 40,960 links in total
 Diameters: 16
- The reduction in the total number of links for this example is 73,728 links or about 64%
- Diameters: only 2 more than that of hypercube

MC(2,4) vs 18-Cube vs 3D Torus

Metacube(2,4) k = 2, m = 4 The number of nodes: $2^{18} = 262.144$ Node degree: 6 Diameter: 20 Hypercube (18-Cube) The number of nodes: $2^{18} = 262.144$ Node degree: 18 Diameter: 18 ■ 3D Torus (64 × 64 × 64) The number of nodes: $2^{18} = 262.144$ Node degree: 6 Diameter: 32 + 32 + 32 = 96

Building Origin2000 with MC(2,2)

An Origin2000 router has six links Two links connect to two CPU boards A board contains two CPUs Four links are used to build an MC(2,2) k = 2 and m = 2 (k + m = 4) There are $2^{m2^{k}+k} = 2^{10} = 1024$ nodes It can connect 1024 x 2 x 2 = 4096 CPUs Does not use the Cray Router anymore In contrast, the Origin2000 using Cray Router can only connect $32 \times 2 \times 2 = 128$ CPUs



Node Address of KMS-Cube



Links of KMS-Cube



KMS-Cube with K = 2, M = 1, and S = 1



KMS-Cube with K = 2, M = 1, and S = 1



KMS-Cube with K = 2, M = 2, and S = 1





Diameter Comparison



Cost Ratio Comparison



RDN — Recursive Dual-Net



Recursive Construction of RDN



$$n_{i} = n_{i-1} \times n_{i-1} \times 2 = 2n_{i-1}^{2}$$

N = (2m)^{2^k}/2 where m is the number of nodes in a base network; k: level

A Recursive Dual-Net RDN(4,1)



RDN(m,k)

m: the number of nodes in the base network; k: level The base network can be an any symmetric network.

A Recursive Dual-Net RDN(4,2)



Cluster 0 of Type 1

Cluster 31 of Type 1

RDN(27,2)

Base network: 3D Torus $(3 \times 3 \times 3)$, k = 2

m = 27 (nodes)

d₀ = 6 (degree)

D₀ = 1 + 1 + 1 = 3 (diameter)

The number of nodes

k = 2: N₂ = 1,458 × 1,458 × 2 = 4,251,528

 $d = d_0 + k = 6 + 2 = 8$ (degree) (k = 2)

 $D = 2(2D_0 + 2) + 2 = 18$ (diameter) (k = 2)

3D Torus (162 × 162 × 162): 4,251,528 nodes; d = 6 (degree)

Base network: 5-node-Ring, k = 3m = 5 (nodes) d₀ = 2 (degree) \square D₀ = 2 (diameter) The number of nodes $k = 1: N_1 = 5 \times 5 \times 2 = 50$ $k = 2: N_2 = 50 \times 50 \times 2 = 5,000$ $k = 3: N_3 = 5,000 \times 5,000 \times 2 = 50,000,000$ d = d_0 + k = 2 + 3 = 5 (degree) (k = 3) $D = 2[2(2D_0 + 2) + 2] + 2 = 30$ (diameter) (k = 3) 3D Torus ($500 \times 500 \times 200$): 50,000,000 nodes; d = 6 (degree) 3D Torus ($500 \times 500 \times 200$): D = 250 + 250 + 100 = 600 (diameter)

Cost Ratio (CR)

$CR(G) = \frac{d(G) \times w_1 + D(G) \times w_2}{\log_2|(G)|}$

INs	Nodes	Degree	Diameter	CR
3D-Torus (10)	1,000	6	15	1.05
10-cube	1,024	10	10	1.00
RDN(5 ² ,1)	1,250	5	10	0.73
RDN(3 ³ , 1)	1,458	7	8	0.71
3D-Torus (128)	2,097,152	6	192	4.72
21-cube	2,097,152	21	21	1.00
DC(10)	2,097,152	11	22	0.79
RDN(5 ² , 2)	3,125,000	6	22	0.65
RDN(3 ³ ,2)	4,251,528	8	18	0.59
RDN(5,3)	50,000,000	5	30	0.68

Topological Properties of INs

Network	Nodes	Degree	Diameter	
3D Torus	n ³	6	3[n/2]	
n-cube	2 ⁿ	n	n	
CCC(n)	n2 ⁿ	3	2n + [n/2] – 2	
HCN(n)	2 ²ⁿ	n + 1	n + [(n + 1)/3] + 1	
Dual-Cube(m)	2 ^{2m+1}	m + 1	2(m + 1)	
Metacube(k, m)	2 ^{m2k+k}	m+k	(m + 1)2 ^k	
RDN(m,k)	(2m) ^{2k} /2	d ₀ + k	$2^{k}D_{0} + 2^{k+1} - 2$	

Degree Comparison



Diameter Comparison



Cost Ratio Comparison



MiKANT — Mirrored K-Ary N-Tree



Dynamic IN: Fat Tree (3-ary 3-tree)



Root switches have fewer ports than others

Bidirectional Clos Network



High switch cost High link cost Long distance

Mirrored K-Ary N-Tree MiKANT(3,3)





Switch Address Format of MiKANT(n, k)



G: Group ID; L: Level (Stage) ID; D: Switch ID

Node Address Format of MiKANT(n, k)



G: Group ID; C: Node ID; C_{n-2} . . . C₀: Switch ID

A switch

 $(G, L, D_{n-2}, \ldots, D_{l+1}, D_{l}, D_{l-1}, \ldots, D_{n})$

will connect to switches

$$\begin{split} \langle G,L+1,D_{n-2},\ldots,D_{L+1},*,D_{L-1},\ldots,D_0\rangle \\ & \text{if } 0 \leq L \leq n-3; \, \text{otherwise} \, (L=n-2) \text{ to switches} \\ & \quad \langle \overline{G},L,*,D_{n-3},\ldots,D_1,D_0\rangle \end{split}$$



There is a link between a switch

 $\langle G, O, D_{n-2}, \ldots, D_1, D_0 \rangle$

and a compute node

$$\langle G, C_{n-1}, C_{n-2}, \ldots, C_1, C_0 \rangle$$

if $D_i = C_i$ for all $i \in \{n - 2, ..., 1, 0\}$.

Mirrored K-Ary N-Tree MiKANT(3,4)


Comparison of Topological Properties

	Classical k-ary n-tree	Clos k-ary n-tree	Mirrored k-ary n-tree	
Nodes	k ⁿ	2k ⁿ	2k ⁿ	
Switches	nk ⁿ⁻¹	(2n-1)k ⁿ⁻¹	(2n-2)k ⁿ⁻¹	
Links	nk ⁿ	2nk ⁿ	(2n-1)k ⁿ	
Degree	2k	2k	2k	
Diameter	2n	2n	2n	
Bisection	k ⁿ /2	k ⁿ /2	k ⁿ /2	
Ave. dist	$2n - \frac{2}{k-1} + \frac{2}{(k-1)k^n}$	$2n - \frac{1}{k-1} + \frac{1}{(k-1)k^n}$	$2n - \frac{1}{k-1} + \frac{1}{(k-1)k^n} - \frac{1}{2}$	

Cost Ratios of Links and Switches



Performance Improvement of MiKANT



Interconnection Networks – 75 / 91

Relative Cost Performance to Hypercube



Average Packet Latencies (Clock Cycles)



Fault Tolerance — Performance



Fault Tolerance — Path Length



Fault Tolerance — Performance



Fault Tolerance — Path Length



K-ary N-tree Peer Network

Peer Fat-Tree

3-ary 3-tree Peer Network



3-ary 3-tree Peer Network



3-ary 4-tree Peer Network



3-ary 4-tree Peer Network



Comparison of Topological Properties

	Classical	Bidir. Clos	Mirrored	Peer
Radix	2k	2k	2k	2k
Diameter	2n	2n	2n	2n
Number of nodes	k ⁿ	2k ⁿ	2k ⁿ	2k ⁿ
Number of switches	nk ⁿ⁻¹	(2n-1)k ⁿ⁻¹	(2n-2)k ⁿ⁻¹	nk ⁿ⁻¹
Number of links	nk ⁿ	2nk ⁿ	(2n-1)k ⁿ	(n + 1)k ⁿ
Bisection width	k ⁿ /2	k ⁿ /2	k ⁿ /2	k ⁿ /2

Summary

- The IN is at the center of supercomputers.
 It is built with switches (routers) and cables that connect ports of switches by following some topologies.
 - Switches: Infiniband, Gigabit Ethernet, or custom
- A good interconnection network should use a small number of links (low cost of switch) and have a short diameter (fast communication).
- IN topologies for building supercomputers
 - Dual-Cube, Metacube, and KMS-Cube
 - RDN Recursive Dual-Net
 - MiKANT and Peer Fat-Tree

Research Topics on Parallel Systems

- Shortest-path routing algorithm
- Multicast and Broadcast algorithms
- Collective communication
- Disjoint paths and Hamiltonian
- Fault-tolerant routing
- Algorithmic design
 - Parallel prefix computation
 - Parallel sorting
- Matrix multiplication and Linpack (Linear algebra)
- Parallel (programming languages and) compilers

Define Cost Ratio CR(G) = $\frac{d(G) \times w_1 + D(G) \times w_2}{\log_2|(G)|}$

d(G): the node degree of G (the number of links of a node) D(G): the diameter of G

|(G)|: the total number of nodes in G

Question: Let $w_1 = w_2 = 50\%$. Calculate CR(G) for

- (a) A 512-node Ring
- (b) A 512-node Completely Connected Network
- (c) A 9-Cube
- (d) A 16 × 32 Two-Dimensional Torus
- (e) An $8 \times 8 \times 8$ Three-Dimensional Torus
- (f) A Dual-Cube DC(4)
- (g) A Mirrored 4-Ary 4-Tree MiKANT(4, 4)
- (h) A Dragonfly DF(g,t,c) with g = 7, t = 2, and c = 4

Report submission: PDF to Moodle by June 27, 23:59, 2025.

Preparing conference/journal papers
Use <u>LATEX</u> to prepare the manuscript
Draw block diagrams with <u>tgif</u>
Generate graphs with <u>gnuplot</u>

CANDAR 2025 CFP (call for papers)

CANDAR 2025, Nov. 25 - Nov. 28, Yamagata

The Thirteenth International Symposium on Computing and Networking

Abstract submission due: July 25, 2025 Paper (PDF) submission due: July 30, 2025