ONE → ALL PERSONALIZED communication (single node SCATTER)

- single processor sends a unique msg to every other processor
- Dual: single node GATHER
  - single proc. collects a unique msg from each other proc.
  - Note: GATHER differs from Accumulation !
- Complexity (1 → all personalized) = complexity (all → all BC)
  source proc sends $m(p-1)$ each proc receives $m(p-1)$

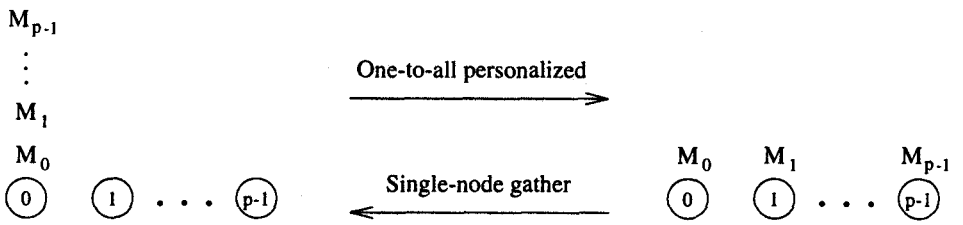
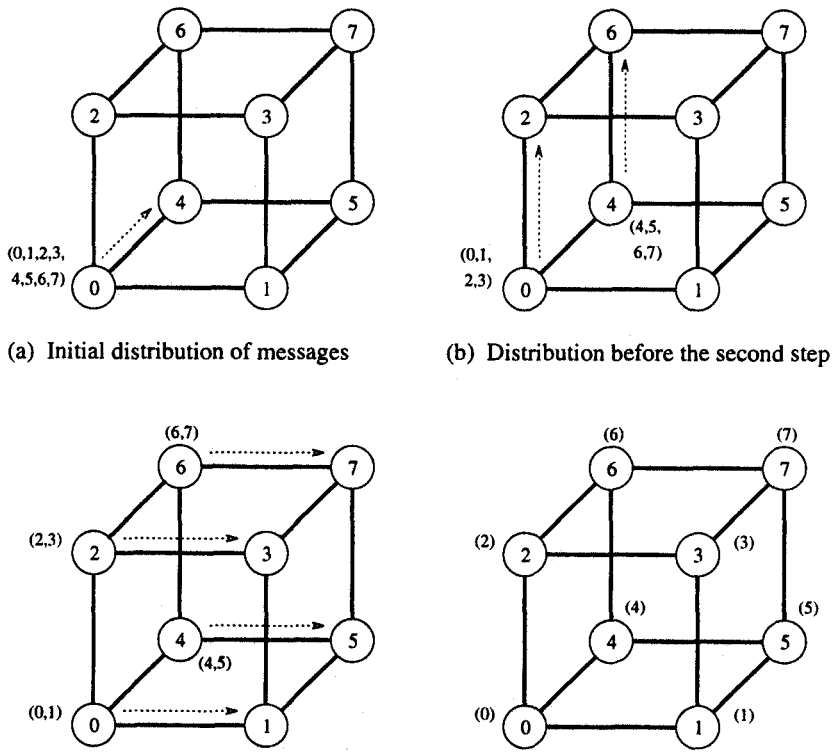


Figure 3.15 One-to-all personalized communication and its dual—single-node gather.

- log p steps
- same communication pattern as for 1→all BC
  but msg size & contents are different
- $T_{all\to all(pers)} = \sum_{i=1}^{\log p} (t_s + 2^{i-1} t_w m) = t_s \log p + t_w m (p-1)$



(a) Initial distribution of messages

(b) Distribution before the second step

(c) Distribution before the third step

(d) Final distribution of messages

Figure 3.16 One-to-all personalized communication on an eight-processor hypercube. Msgs are labeled by the labels of their dests

- Ring (CT & SF)

$$T_{1\to all(pers)} = (t_s + t_w m)(p-1)$$

- 2D square mesh (CT & SF)

$$T_{1\to all(pers)} = 2t_s(\sqrt{p}-1) + t_w m (p-1)$$

ALL → ALL PERSONALIZED COMMUNICATION

- each proc sends distinct msg to each other proc
- eg uses: parallel FFT, matrix transpose, parallel database join op
- Communication pattern same as for all → all BC
  → msg size & contents differ

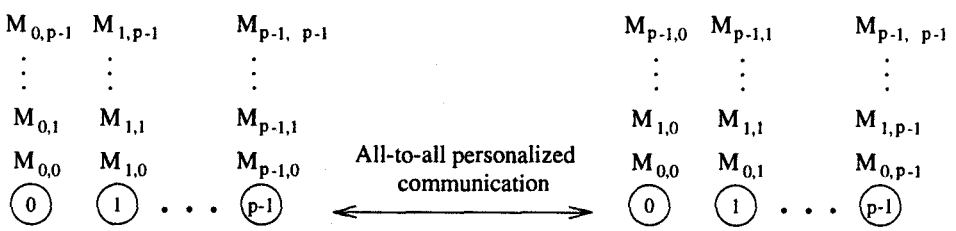


Figure 3.17 All-to-all personalized communication.

- Ring (SF)

→ $(p-1)$ steps

→ i^{th} step msg size $= m(p-i)$

$$T_{all\to all\ (pers)} = \sum_{i=1}^{p-1} \left(t_s + t_w m(p-i)\right) = \left(t_s + \frac{1}{2} t_w m p\right)(p-1)$$

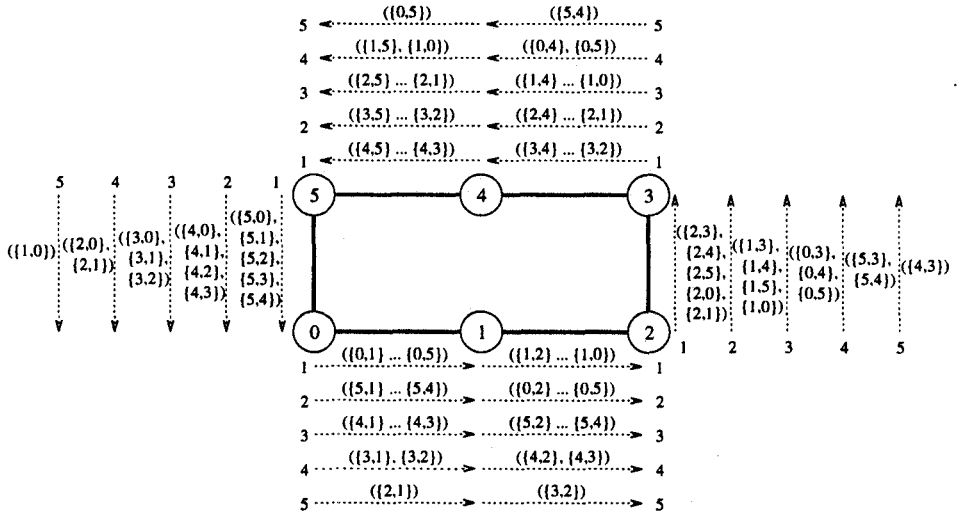


Figure 3.18 All-to-all personalized communication on a six-processor ring. The label of each message is of the form $\{x, y\}$, where x is the label of the processor that originally stored the message, and y is the label of the processor that is the final destination of the message. The label $(\{x_1, y_1\}, \{x_2, y_2\}, \ldots, \{x_n, y_n\})$ indicates a message that is formed by concatenating n individual messages.

→ assumption: msgs sent in 1 direction only

- Total traffic $= m(p-1) \times \frac{p}{2} \times p$, shared across p channels
- Communic time $\geq \frac{t_w m p(p-1)}{2}$ (same as above for SF)

∴ cannot be improved with CT

MESH (SF)

- Phase 1 (row): $\left(t_s + \frac{t_w (m\sqrt{p})\sqrt{p}}{2}\right)(\sqrt{p}-1)$
- Phase 2 (col): same

$$T_{all \to all\ (pers)} = (2t_s + t_w mp)(\sqrt{p}-1)$$

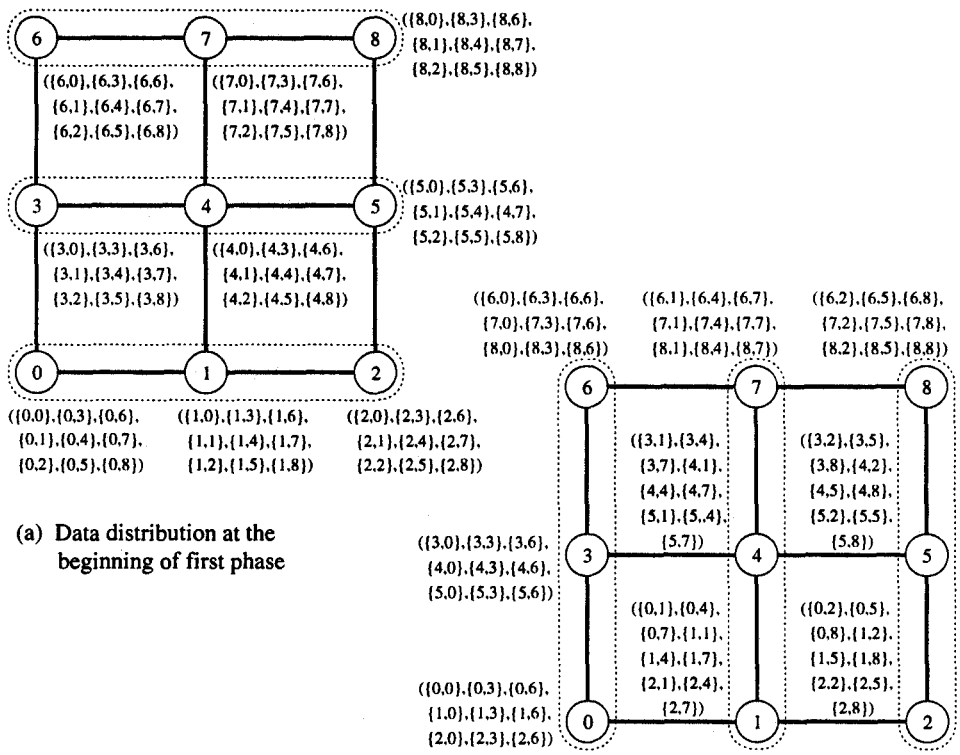
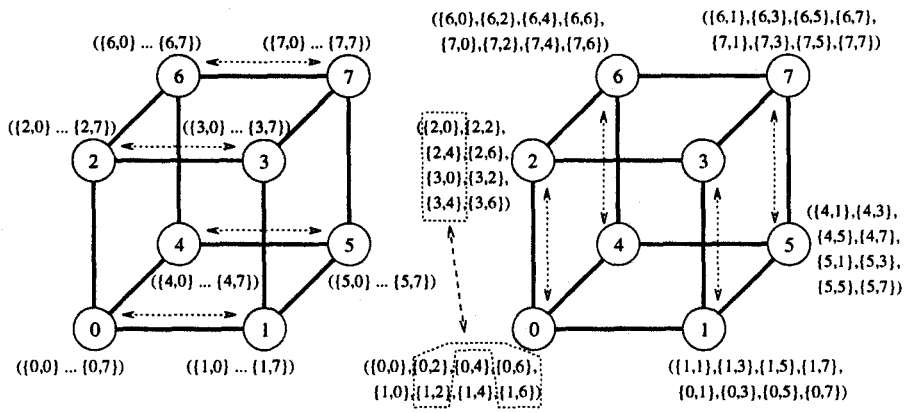


Figure 3.19 The distribution of messages at the beginning of each phase of all-to-all personalized communication on a 3 × 3 mesh. At the end of the second phase, processor i has messages $(\{0,i\}, \ldots, \{8,i\})$, where $0 \leq i \leq 8$. The groups of processors communicating together in each phase are enclosed in dotted boundaries.

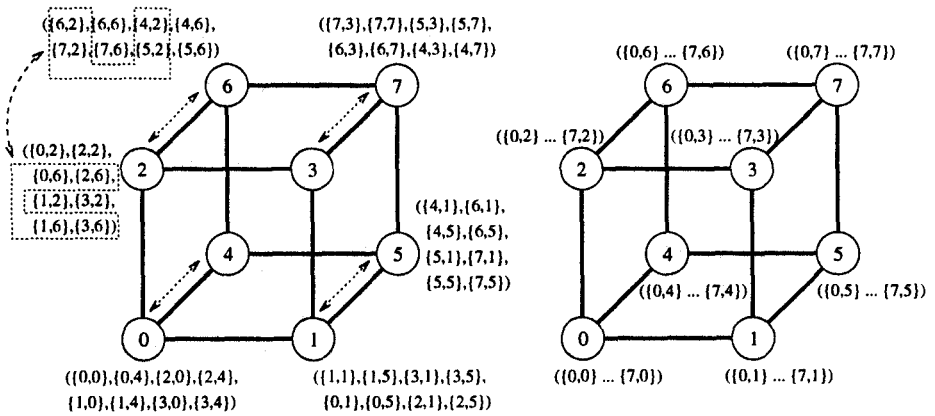
- Extra ovhd for local rearrangement of data
  → $t_r mp$, where t_r = time to do a read & write on single word
- Cannot be improved by CT (same reasoning as for ring)

- Extend 2D to log p dimensions
- Send data for the "other" subcube, in each step
- $\frac{mp}{2}$ words exchanged along bi-directional links in each of log p steps
- $T_{all \to all (pers)} = (t_s + \frac{1}{2} t_w mp) \log p$



(a) Initial distribution of messages

(b) Distribution before the second step



(c) Distribution before the third step

(d) Final distribution of messages

Figure 3.20 All-to-all personalized communication on a three-dimensional hypercube with SF routing.

- Local rearrangements ovhd = $t_r mp \log p$

(t_r = time to do a single read and write on a word)

- $T_{all \to all\ (pers)}^{\text{lower bound}} = \dfrac{t_w m (p-1)(p \log p)/2}{(p \log p)/2} = t_w m (p-1)$

CT

- $(p-1)$ min. communication steps; j^{th} step: i exchanges data w/ $(i \text{ XOR } j)$
- no congestion
- Ecube routing time/step $= t_s + t_w m + l t_h \Rightarrow$ Total $T_{all \to all\ (pers)} =$

$$(t_s + t_w m)(p-1) + \frac{t_h p \log p}{2}$$

(t_h term is greater than for SF)

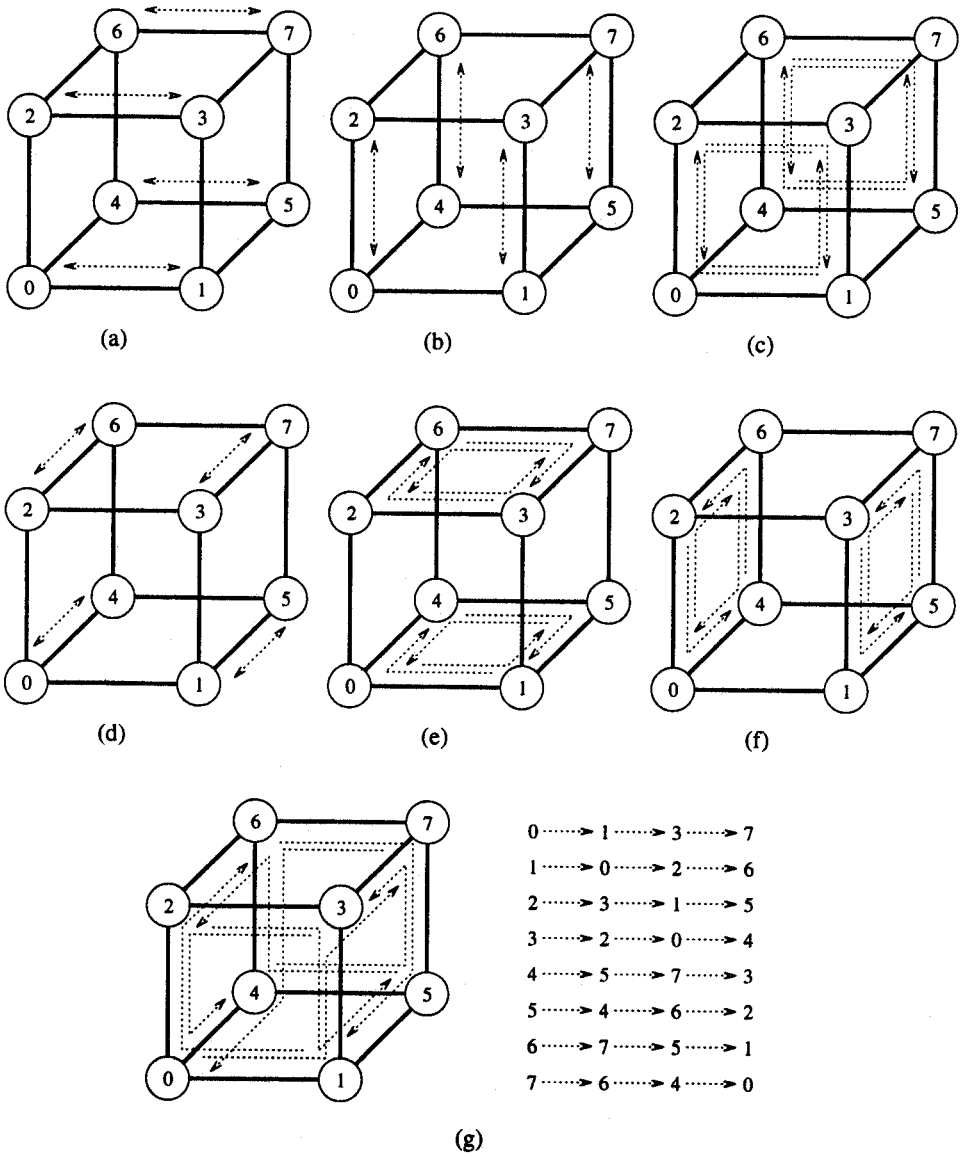


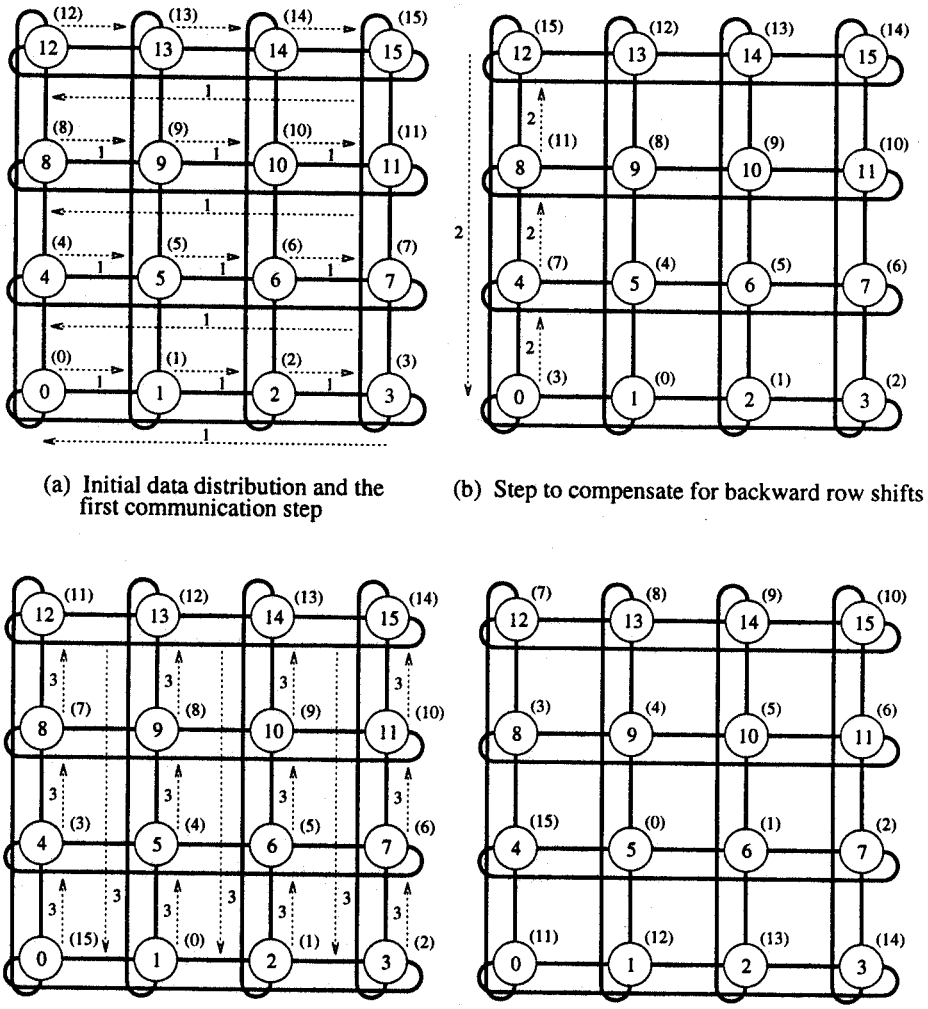
Figure 3.21 Seven steps in all-to-all personalized communication on an eight-processor hypercube with CT routing.

```
1.  procedure ALL_TO_ALL_PERSONAL(d, my_id)
2.  begin
3.      for i := 1 to 2^d − 1 do
4.      begin
5.          partner := my_id XOR i;
6.          send M_{my_id, partner} to partner;
7.          receive M_{partner, my_id} from partner;
8.      endfor;
9.  end ALL_TO_ALL_PERSONAL
```

Program 3.8 A procedure to perform all-to-all personalized communication on a d-dimensional hypercube with CT routing. The message $M_{i,j}$ initially resides on processor

CIRCULAR SHIFT (used in some matrix computations, string & image pattern matching)

- Ring: straightfwd
- MESH: stage ①: shift by $(q \bmod \sqrt{p})$ steps along rows
  ①.5: data that wrapped around must shift 1 step along cols
  stage ②: shift by $\lfloor q/\sqrt{p} \rfloor$ steps along cols.



(a) Initial data distribution and the first communication step

(b) Step to compensate for backward row shifts

(c) Column shifts in the third communication step

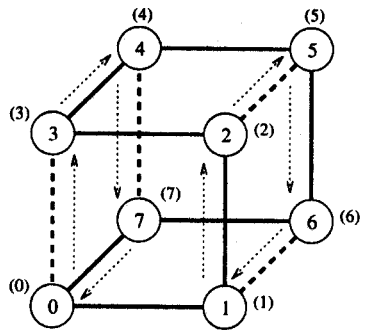
(d) Final distribution of the data

Figure 3.22 The communication steps in a circular 5-shift on a 4 × 4 mesh.

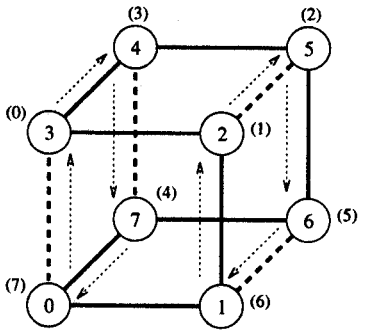
$$T_{shift} = (t_s + t_w m)\left(2\left\lfloor \frac{\sqrt{p}}{2} \right\rfloor + 1\right)$$

[assuming both fwd & backward shifts]

- Map ring to HC using RGC
- Property: 2 processors at a distance of 2^i on ring are separated by exactly 2 links on HC (exception $i = 0$)
- # communic phases = # 1's in the binary representation of q in a q-shift
- Total # steps in q-shift = $2 \log p - 1$

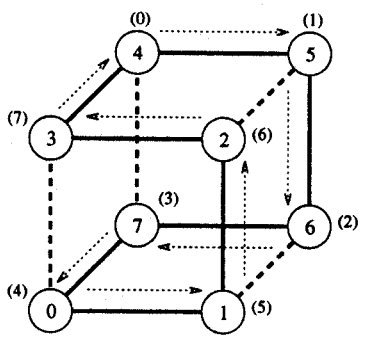


First communication step of the 4-shift

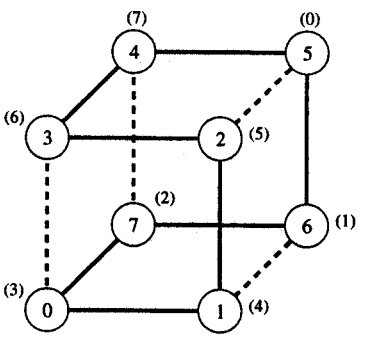


Second communication step of the 4-shift

(a) The first phase (a 4-shift)



(b) The second phase (a 1-shift)



(c) Final data distribution after the 5-shift

Figure 3.23 The mapping of an eight-processor ring onto a three-dimensional hypercube to perform a circular 5-shift as a combination of a 4-shift and a 1-shift.

SF

- all communication in a step is congestion-free

[property of ring mapping:] procs whose distance on ring is power of 2 are in disjoint subrings on HC

$$T_{circ\_shift} = (t_s + t_w m)(2 \log p - 1)$$

CT

- Use std labeling of processors, not RGC
- Use std E-cube routing to ensure congestion-free paths
- For q-shift, longest path has $(\log p - \gamma(q))$ links,
  where $\gamma(q)$ = highest int j such that q is divisible by 2^j

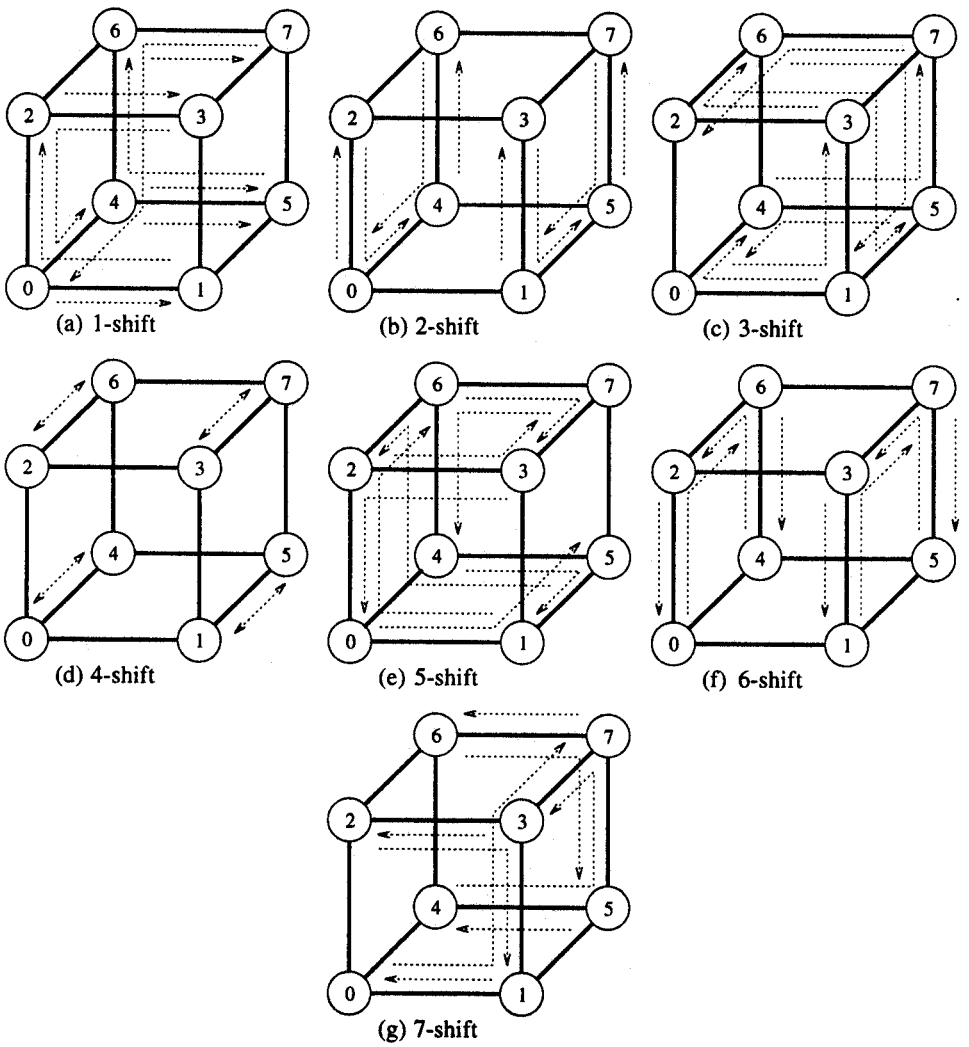


Figure 3.24 Circular q-shifts on an 8-processor hypercube for $1 \leq q < 8$.

CT

$$T_{circular\_shift} = t_s + t_w m + t_h (\log p - \gamma(q))$$

- Routing Msg in Parts is faster?
- HC properties

1) $\log p$ distinct paths between any pair of procs.
If labels differ in ℓ bits, ℓ paths have ℓ links each
$(\log p - \ell)$ paths have $(\ell + 2)$ links each

Msg split into $\log p$ parts, each to dest along separate path, (longest 1st)
then dest can receive all data in max. $(2 \log p)$ steps
$\Rightarrow 2(t_s \log p + t_w m)$ time

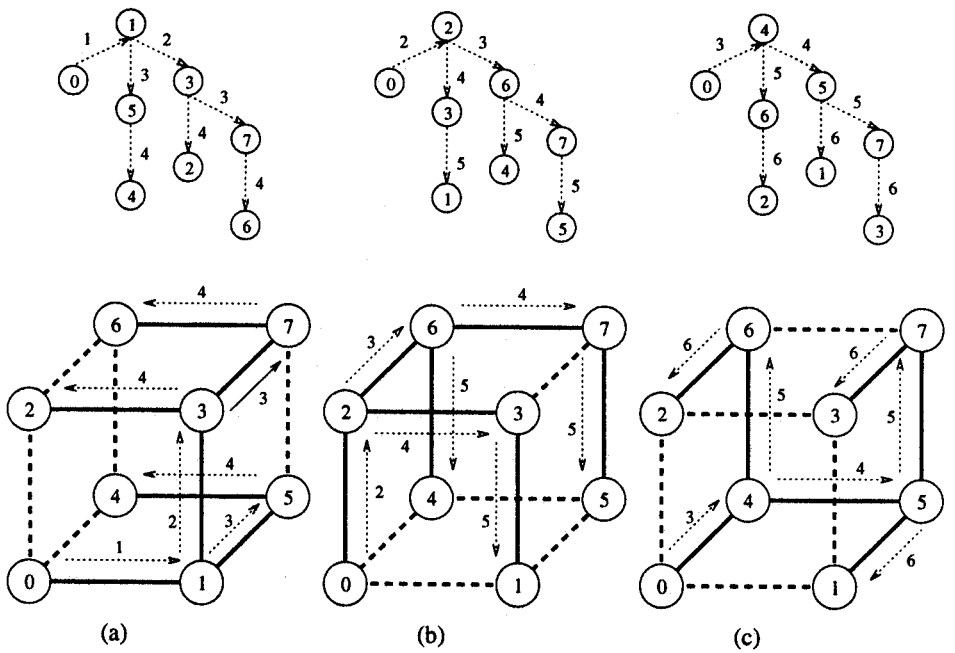


Figure 3.25 The six time-steps in one-to-all broadcast on an eight-processor hypercube with SF routing when the message is split into three parts that are routed separately on three different spanning binomial trees.

1→all BC

2) p-node binomial tree can be embedded into p-node HC w/ 1-1 node mapping

$\log p$ binomial trees rooted at 3 neighbors of source proc ϕ.
Each processor (incl. root) sends out received msg to subtrees in the order of decreasing sizes of the subtrees

$\Rightarrow$ conflict-free msg passing

Table 3.1 Summary of communication times of various operations discussed in Sections 3.2–3.5 on different architectures with one-port communication and CT routing. The message size for each operation is m and the number of processors is p. The time for one-to-all broadcast on the hypercube is not optimal, and, as shown in Section 3.7.1 and Problem 3.24, can be improved to $2(t_s \log p + t_w m)$. In the hypercube expression for circular q-shift, $\gamma(q)$ is the highest integer j such that q is divisible by 2^j.

Operation	Ring	2-D Mesh (wraparound, square)	Hypercube
One-to-all broadcast	$(t_s + t_w m) \log p + t_h(p-1)$	$(t_s + t_w m) \log p + 2t_h(\sqrt{p}-1)$	$(t_s + t_w m) \log p$
All-to-all broadcast	$(t_s + t_w m)(p-1)$	$2t_s(\sqrt{p}-1) + t_w m(p-1)$	$t_s \log p + t_w m(p-1)$
One-to-all personalized	$(t_s + t_w m)(p-1)$	$2t_s(\sqrt{p}-1) + t_w m(p-1)$	$t_s \log p + t_w m(p-1)$
All-to-all personalized	$(t_s + t_w mp/2)(p-1)$	$(2t_s + t_w mp)(\sqrt{p}-1)$	$(t_s + t_w m)(p-1) + (t_h/2)p \log p$
Circular q-shift	$(t_s + t_w m)\lfloor p/2 \rfloor$	$(t_s + t_w m)(2\lfloor \sqrt{p}/2 \rfloor + 1)$	$t_s + t_w m + t_h(\log p - \gamma(q))$

For SF, above results are valid, except

1) 1 → all BC : ring $(t_s + t_w m)\lceil p/2 \rceil$

mesh $2(t_s + t_w m)\lceil \sqrt{p}/2 \rceil$

2) all → all personalized communication :

HC : $\left(t_s + t_w \frac{mp}{2}\right) \log p$