CS 442 Introduction to Parallel Processing
Homework 3

Erik Erhardt

February 20, 2006

GGKK 4.2 *One-to-all broadcast, along lowest dimension first: Alogrithm 4.1.*

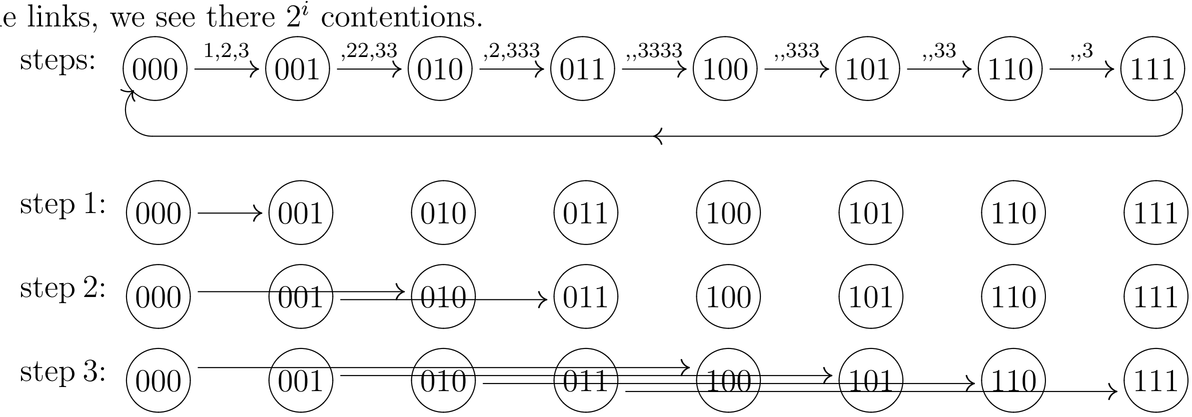
1. *Hypercube cost.*
Same as in highest dimension first; no contention.

$$\begin{aligned} T &= \sum_{i=0}^{\log_2 p-1} (t_s + t_w m) \\ &= (t_s + t_w m)\log_2 p \end{aligned}$$

2. *Ring cost.*
Assuming t_s is small enough and $t_w m$ is large enough so that $k t_w m$ causes contention at every stage.

$$T = \sum_{i=0}^{\log_2 p-1} (t_s + t_w m 2^i)$$

For example, on a ring with $p = 2^3$, with each communication noted on the links, we see there 2^i contentions.



GGKK 4.6 *All-reduce, sum, on ring, assuming $p = 2^d$.*

i. *All-to-all broadcast, local sum.*

ii. *All-to-one reduce, one-to-all broadcast.*

iii. *All-to-all, but sends cumulative sum only.*

1. See table.

2. (*iii.*) is clearly better than (*i.*), and (*ii.*) is better than (*i.*) when $p \geq 2^2$, but (*ii.*) should not beat (*iii.*) for large p if m is sufficiently large.

3. (*i.*) and (*iii.*) are equal, and (*ii.*) is better when $p \geq 2^2$.

case	*i.*	*ii.*	*iii.*
1.	$T = (p-1)(t_s + t_w m)$	$T = 2(t_s + t_w m)\log_2 p$	$T = (p-1)(t_s + t_w 1)$
2.	$T = (p-1)(100 + m)$	$T = 2(100 + m)\log_2 p$	$T = (p-1)(100 + 1)$
3.	$T = (p-1)(100 + 1)$	$T = 2(100 + 1)\log_2 p$	$T = (p-1)(100 + 1)$

GGKK 4.9 *Hypercube shortest distance.*

The number of nodes n whose shortest distance is length l from any given node in a p-node hypercube follows the binomial coefficient, so that $n = \binom{\log_2 p}{l} = \frac{(\log_2 p)!}{l!(\log_2 p - l)!}$.