| CS 537 | Randomized Routing in a Hypercube | Fall 2013 |
|---|---|---|
| | Prepared by: Aanchal Malhotra | |

In this lecture we will discuss a very interesting application of Chernoff Bounds, that is, packet routing in sparse networks. We first discuss the deterministic oblivious permutation routing algorithm in a hypercube network, and analyze the worst case. Then we discuss the randomized algorithm for the same problem.

## 1. Basic Terminology:

a. A *telecommunications network* is a collection of terminal nodes, links and any intermediate nodes which are connected so as to enable telecommunication between the terminals.

b. A *computer network* or *data network* is a telecommunications network that allows computers to exchange data. Links and nodes are the elements of a computer network. The best-known computer network is the Internet.

c. *Network topology* is the arrangement of the various elements (links, nodes, etc.) of a computer network. The study of network topology recognizes eight basic topologies: point-to-point, bus, star, ring, mesh, tree, hybrid, daisy chain etc.

d. A *fully connected network* is a communication network in which each node is connected to every other node. In graph-theoretic terms, it is a complete graph. The number of connections c grows quadratically with the number of nodes *n* as per the formula below and so it is extremely impractical for large networks.

$$C = n(n-1)/2$$

e. *Routing* is the process of finding a path in a network from source to destination. Given a network topology, a *routing algorithm* specifies, for each pair of nodes, a route or a sequence of edges connecting the pair in the network.

## 2. Permutation Routing on the Hypercube

### 2.1 Hypercube

A hypercube can be viewed as an $n$-dimensional version of a square ($n$=2), or a cube ($n$=3). A hypercube with dimension $n$ (some integer) has $N=2^n$ nodes. Let $x' = (x_1, x_2, \ldots, x_n)$ denote the binary representation of the number $0 \leq x \leq N-1$.

Figure 1 shows three hypercubes with dimensions one, two and three.

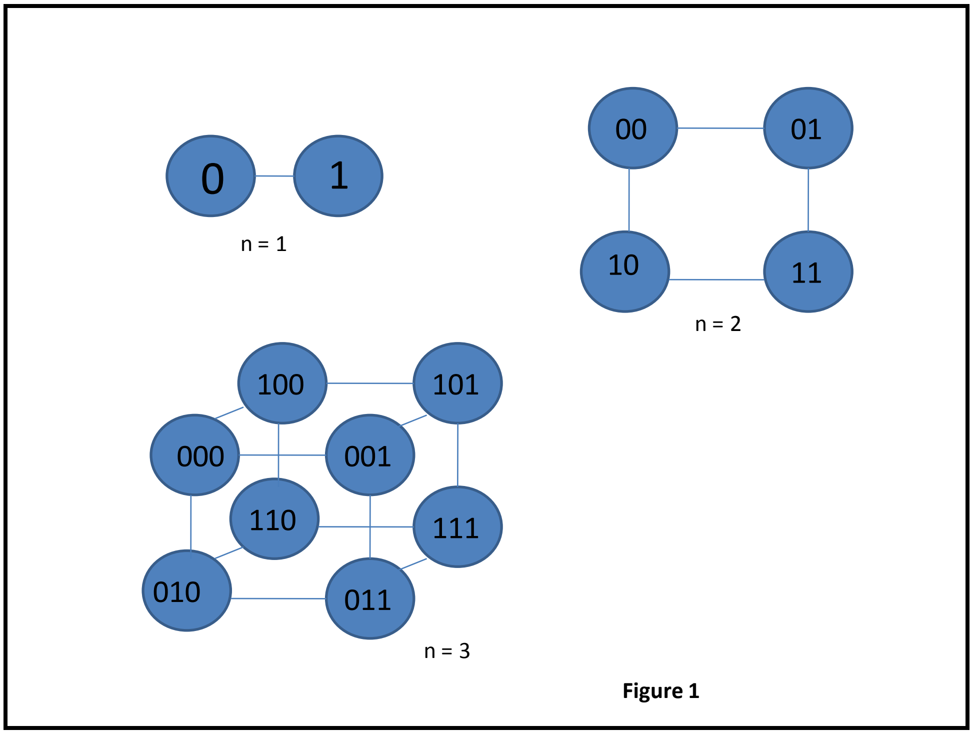


Figure 1

**Definition 1.** *An n-dimensional hypercube is a network with $N = 2^n$ nodes such that node x has a direct connection to the node y if and only if x´ and y´ differ in exactly one bit.*

So according to figure 1, the total number of directed edges in the n-cube is *2nN* by virtue of the fact that each node has exactly *n* outgoing and *n* incoming edges.

Let us now analyze the problem of permutation routing in a hypercube network. Here we make two assumptions: (a) Each node sends exactly one packet to its destination, (b) Each node receives exactly one packet. We can now observe that the problem of routing in a hypercube corresponds to some permutation of the address space of the computers or nodes in the network. This topology of the hypercube allows for a natural *bit-fixing* routing algorithm as shown below. As the name suggests, at each step one bit of the address is fixed, starting from the least significant bit.

**n-cube Bit-Fixing Routing Algorithm**

1. Let x´ and y´ be the binary representations of the source and the destination nodes of the packet.
2. For i = 1 to n, do

   (a) If $x_i \neq y_i$, then traverse the edge $(y_1, y_2, \ldots, y_{i-1}, x_i, \ldots, x_n) \rightarrow (y_1, y_2, \ldots, y_{i-1}, y_i, x_{i+1}, \ldots, x_n)$

n-cube Bit-Fixing Routing Algorithm

Let me illustrate with a simple example. Let x´ = 1011 and y´ = 0000, then a packet from node representing x´ destined to node representing y´ would first pass through node 0011, and then through node 0001, and will finally reach its destination node 0000.

2.2 Worst-Case Analysis

**Theorem 1.** *Given the worst case permutation π, the bit-fixing routing algorithm requires at least $\Omega(\sqrt{N}/n)$ rounds*.

**Proof 1:** Without loss of generality, let us assume that n is even. Further we split each of the source and the destination address into two binary vectors each of length n/2. Now we define the permutation π as π(ab) = ba (where my source vector is ab and the destination vector is ba). For instance, while sending a packet from source 10001111 to destination 00000000, we perform the following steps (Figure 2):

| **a** | **b** |
|---|---|
| 1000 | 1111 |
| 1100 | 1111 |
| . | . |
| . | . |
| **1111** | **1111** |
| . | . |
| . | . |
| 1111 | 1000 |
| **b** | **a** |

Figure 2

On careful observation, we see that while performing bit-fixing algorithm on permutation of the type shown in Figure 2, there is a node of the form *aa* (shown in bold in the figure), which means that left part equals right part of the permutation in this node. And for any source-destination pair $ab \rightarrow ba$, bit-fixing algorithm always uses such a node *aa* to route the packet towards its destination. There are at most $2^{n/2} = \sqrt{N}$ number of nodes of the form *aa* in the n-dimensional hypercube and we have a total of *N* packets to be routed (recall that each node has exactly one packet to be sent) through these nodes. Also, each of these nodes has n direct links and according to our assumption in the beginning, at most one packet can pass through an edge in one step. Therefore, routing *N* packets through $\sqrt{N}$ nodes sending at most n packets in one step, we require at least $(\sqrt{N}/n)$ rounds. Thus we obtain a lower bound of $\Omega(\sqrt{N}/n)$ on the number of rounds in the worst case.

3. Analysis of the randomized permutation Algorithm

Please refer to pages 75-28 from [2].

References:


1. Computer Network [Internet]. Wikipedia. Available from: en.wikipedia.org/wiki/Computer_network (Last available from: December 12, 2013).
2. Mitzenmacher M, Upfal E. 2005. Probability and Computing: Randomized Algorithms and Probabilistic Analysis. Oxford University Press.
3. Sommer C. 2010. Advanced Algorithms (lecture) [Internet]. University of Tokyo. Available from: ww.sommer,jp/aa10/aa9.pdf (Last accessed on December 12, 2013).