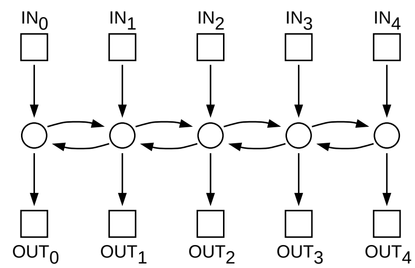
Massachusetts Institute of Technology
6.042J/18.062J, Spring '07: Mathematics for Computer Science — March 16
Prof. Albert R. Meyer — revised March 16, 2007, 740 minutes

In-Class Problems Week 6, Fri.

Problem 1. A *5-path* communication network is shown below. From this, it's easy to see what an n-path network would be. Fill in the table of properties below, and be prepared to justify your answers.



5-Path

network	# switches	switch size	diameter	max congestion
5-path				
n-path				

Problem 2. A *binary-tree network* has n inputs and n outputs, where n is a power of 2. Each input is connected to the root of a binary tree with $n/2$ leaves and with edges pointing away from the root. Likewise, each output is connected to the root of a binary tree with $n/2$ leaves and with edges pointing toward the root.

Two edges point from each leaf of an input tree, and each of these edges points to a leaf of an output tree. The matching of leaf edges is arranged so that for every input and output tree, there is an edge from a leaf of the input tree to a leaf of the output tree, and every output tree leaf has exactly two edges pointing to it.

(a) Draw such a binary-tree net for $n = 4$.

(b) Fill in the table, and explain your entries.

# switches	switch size	diameter	max congestion

Problem 3. The Beneš network has a max congestion of 1; that is, every permutation can be routed in such a way that a single packet passes through each switch. Let's work through an example. A Beneš network of size $N = 8$ is attached.

(a) Within the Beneš network of size $N = 8$, there are two subnetworks of size $N = 4$. Put boxes around these. Hereafter, we'll refer to these as the *upper* and *lower* subnetworks.

(b) Now consider the following permutation routing problem:

$$\begin{aligned} \pi(0) &= 3 & \pi(4) &= 2 \\ \pi(1) &= 1 & \pi(5) &= 0 \\ \pi(2) &= 6 & \pi(6) &= 7 \\ \pi(3) &= 5 & \pi(7) &= 4 \end{aligned}$$

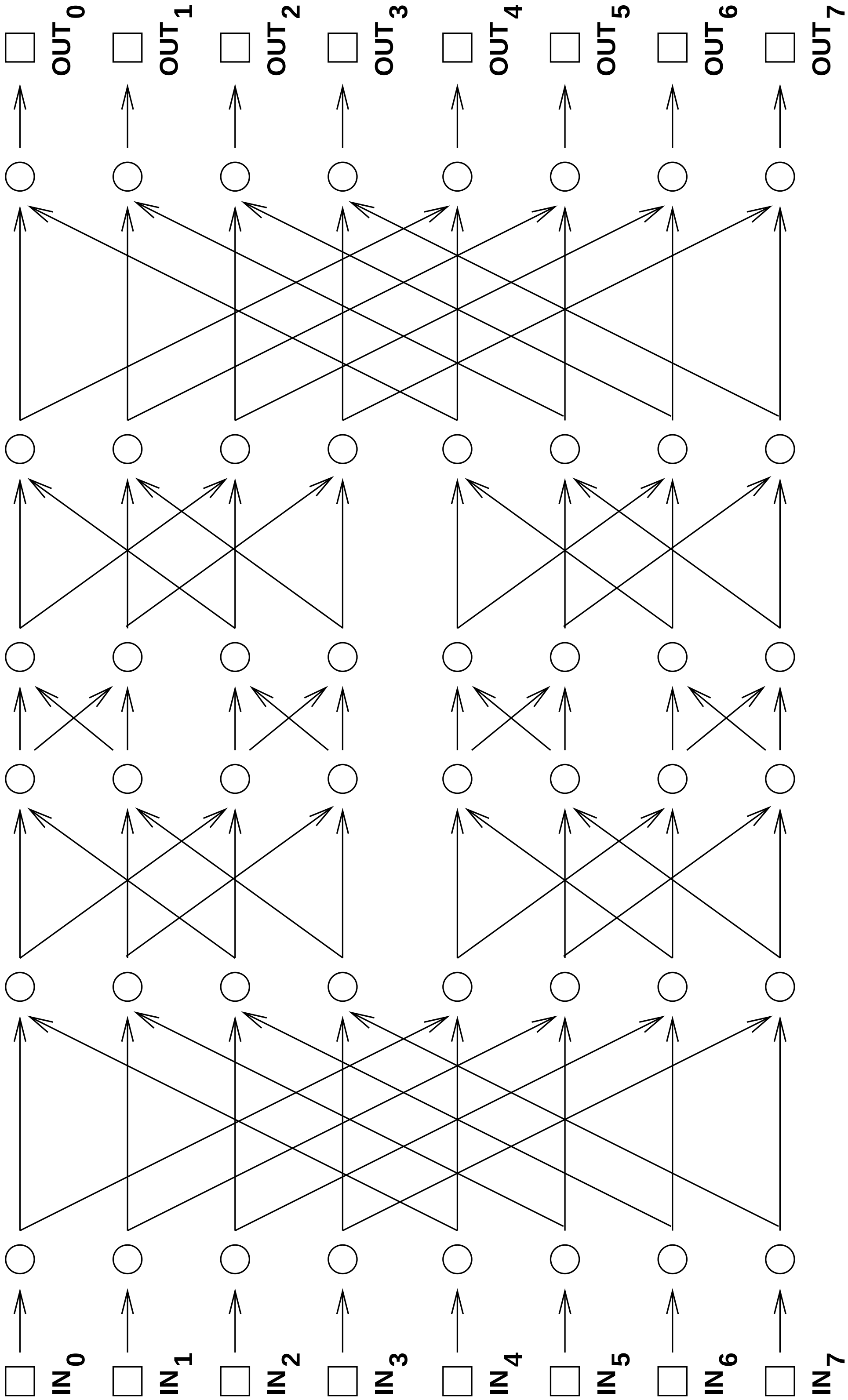
Each packet must be routed through either the upper subnetwork or the lower subnetwork. Construct a graph with vertices 0, 1, ..., 7 and draw a *dashed* edge between each pair of packets that can not go through the same subnetwork because a collision would occur in the second column of switches.

(c) Add a *solid* edge in your graph between each pair of packets that can not go through the same subnetwork because a collision would occur in the next-to-last column of switches.

(d) Color the vertices of your graph red and blue so that adjacent vertices get different colors. Why must this be possible, regardless of the permutation π?

(e) Suppose that red vertices correspond to packets routed through the upper subnetwork and blue vertices correspond to packets routed through the lower subnetwork. On the attached copy of the Beneš network, highlight the first and last edge traversed by each packet.

(f) All that remains is to route packets through the upper and lower subnetworks. One way to do this is by applying the procedure described above recursively on each subnetwork. However, since the remaining problems are small, see if you can complete all the paths on your own.


OUT$_0$
OUT$_1$
OUT$_2$
OUT$_3$
OUT$_4$
OUT$_5$
OUT$_6$
OUT$_7$
IN$_0$
IN$_1$
IN$_2$
IN$_3$
IN$_4$
IN$_5$
IN$_6$
IN$_7$