

4A. Theoretical analysis

The question stated, the worse-case should be the switch is always fully loaded. That is, for each of the four in ports, there are always packets going to each of the out ports. For the sake of clarity, name the input ports 1,2,3,4 and output ports a,b,c,d. Packet 1-a would be the packet going from in port 1 to out port a. In the worst case scenario, there would be 16 packets: 1-a to 1-d, 2-a to 2-d, 3-a to 3-d and 4-a to 4-d all together waiting to be forwarded. Without even going into details, it can be concluded that it will take at least 4 iterations to transfer all the 16 packets to the 4 out ports.

It is impossible for any packets to wait for more than 3 iterations before the packet gets a chance to be forwarded. To prove this, consider that each time if the out port does not receive acknowledgement for its offer to an in port, it will move on to the next highest priority in port, until the out port is filled or all the packets will be routed. Therefore if an in port sends a request to an out port, the out port will take the request and send the packet right away, or the out port will take the request from some other port and increment its priority pointer. In this case it will take maximum 4 iterations for the out port's priority pointer to the in port that has a packet to be transferred.

For the worst case scenario example, consider the situation such as initially all input ports are filled with packets going to each of the out ports. There are 16 packets to be transferred in total. Without losing generality, assume at the beginning all ports' priority pointers are at 1 or a.

Initial priority pointer list:

In port	Out port
a	1
a	1
a	1
a	1

After the 1st iteration, 1-a, 2-b, 3-c, and 4-d are transferred. The new priority list is as following:

In port	Out port
b	2
c	3
d	4
a	1

After 2nd iteration, 1-d, 2-a, 3-b and 4-c are transferred. The new priority list is below:

In port	Out port
a	3
b	4
c	1
d	2

After 3rd iteration, 1-c, 2-d, 3-a, and 4-b are transferred. The new priority list is below:

In port	Out port
d	4
a	1
b	2
c	3

Among the initial 16 packets, the only 4 packets left waiting to be transferred are: 1-b, 2-c, 3-d and 4-a. Since in ports 1,2,3,4 will send out requests to the out ports to which they have packets to send, and the out ports respond to the in ports with highest priorities, in port 1 will receive response from port b, as in port 1 is of the highest priority on port b's list. Same idea applies to all other three out ports. So the 4 packets left will for sure be transmitted during this iteration.

In conclusion, the maximum blocking time for a packet at head of VOQ is 3 iterations.