

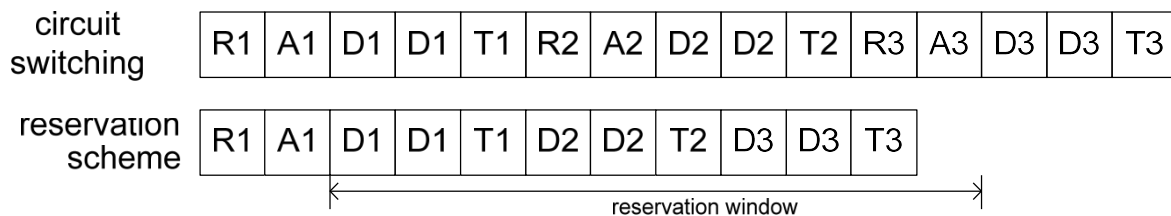
Problem M7.1: Networks-on-Chip

Problem M7.1.A

Consider a flow control method similar to circuit switching but where the request message 'reserves' each channel for a fixed period of time in the future (for example, for 10 cycles since a reservation is made). At each router along the path, a reservation is made if a request from a neighbor can be accommodated. If the request cannot be accommodated a NACK is sent that cancels all previous recommendations for the connection, and the request is retired. If a request reaches the destination, an acknowledgement is sent back to the source, confirming all reservations.

Draw a time-space diagram of a situation that demonstrates the advantage of reservation circuit switching over conventional circuit switching.

Clearly, this scheme eliminates the overhead of establishing connections for every packet. For example, if a source is sending out short packets (two data flits per packet) and the reservation window is 10 cycles, the time-space diagram looks like the following:



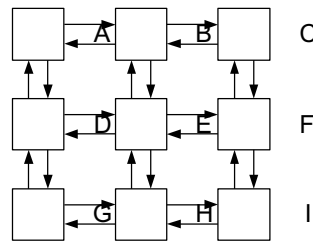
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14

Note that tail flits may be able to get eliminated in this new scheme if they are used only to indicate when channels can be deallocated.

Problem M7.1.B

(a) Randomized dimension-order: All packets are routed minimally. Half of the packets are routed completely in the X dimension before the Y dimension and the other packets are routed Y before X.

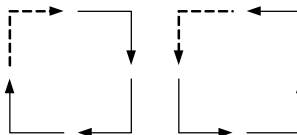
No, this generates a cycle in CDG.



In the CDG corresponding to the mesh network above, for example, $EF \rightarrow FC \rightarrow CB \rightarrow BE \rightarrow EF$ generates a cycle (Flow E-F-C, flow F-C-B, flow C-B-E, and flow B-E-F will generate a deadlock).

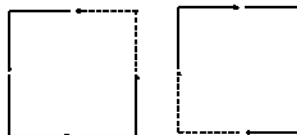
(b) Less randomized dimension-order: All packets are routed minimally. Packets whose minimal direction is increasing in both X and Y, always route X before Y. Packets whose minimal direction is decreasing in both X and Y, always route Y before X. All other packets randomly choose between X before Y and vice versa.

Yes. This effectively eliminates the following two turns.



This corresponds to the third turn model, '*negative-first*' model, which is

deadlock-free. (c) All packets are prohibited to take the two turns in dash:



No. In the 3-by-3 mesh network in part (a), $EB \rightarrow BC \rightarrow CF \rightarrow FE \rightarrow ED \rightarrow DG \rightarrow GH \rightarrow GE \rightarrow EB$ generates a cycle.

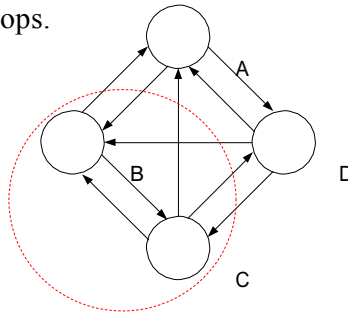
Problem M7.2: Non-mesh Networks

Problem M7.2.A

Fill in the following table of the properties of this network.

Diameter	2
Average Distance	7/8
Bisection Bandwidth	6 flit/cycle

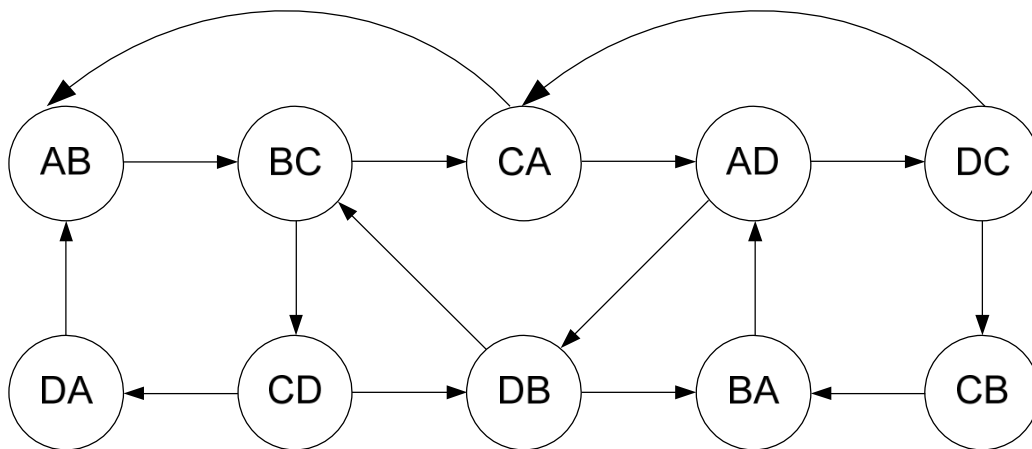
Including self-loops, there are 16 unique (source, destination) pairs. Among them, only the routing distance of B-to-D and A-to-C is 2 hops (which is the diameter of this network), four of them are 0 hops, and all others are 1. Therefore, the average distance is $(0*4 + 1*10 + 2*2)/16 = 7/8$ hops.



And the bisection bandwidth is 6 flit/cycle.

Problem M7.2.B

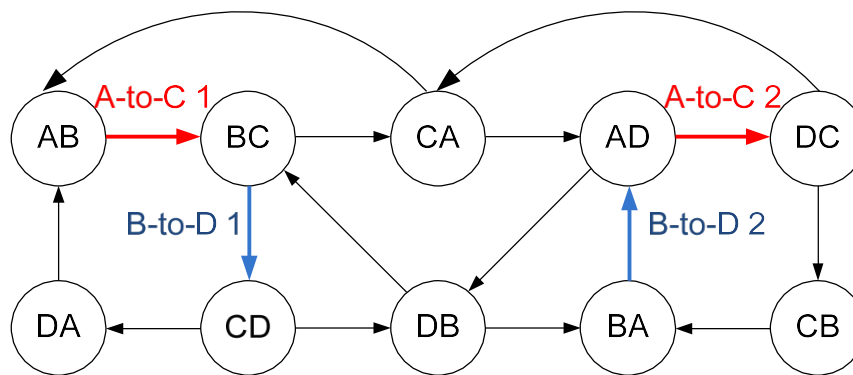
Draw the channel dependency graph of this network.



Problem M7.2.C

Is a **minimal** routing on this network deadlock-free? Show your reasoning and give a deadlock scenario if it is not deadlock-free.

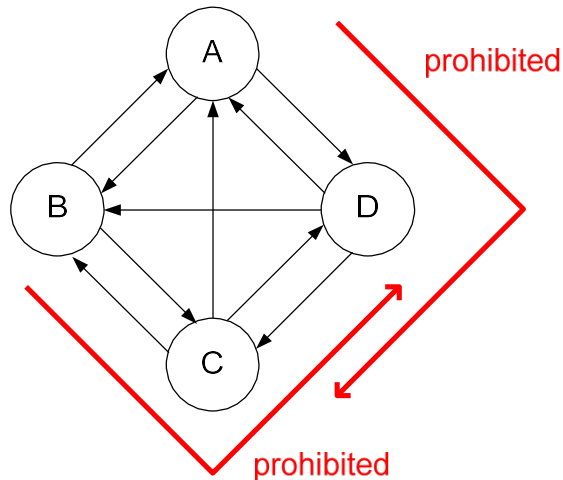
Yes. In minimal routing, all flows except for the ones from B to D and from A to C have 1-hop distance, which is represented by a single node in CDG; they are not holding resources while waiting for another because they need only one resource. The dependencies of flow from B to D and from A to C are represented in the CFG as following (note that each flow can take two possible minimal routes):



There are no cycles in the CDG, thus the routing is deadlock-free.

Problem M7.2.D

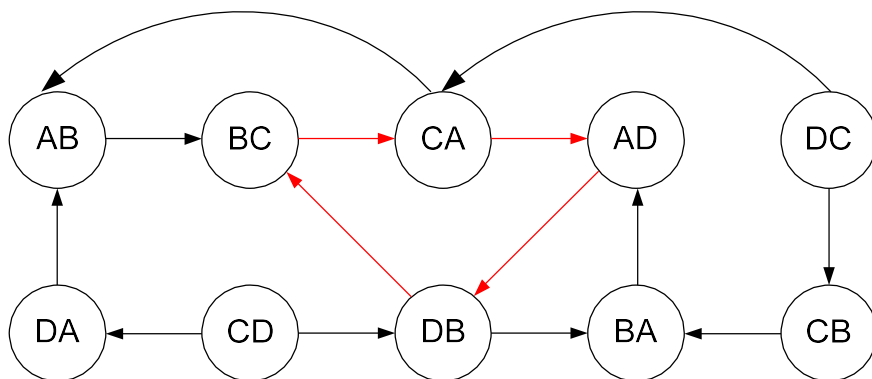
Now, we use a possibly **non-minimal** routing on this network. Plus, we prohibited the following two movements on the non-minimal routing: 1) A to D then D to C and 2) B to C then C to D.



Is this routing deadlock-free? Show your reasoning and give a deadlock scenario if it is not deadlock-free.

No.

Prohibiting those movements, the CDG becomes:



However, there are still cycles in this CDG. For example, if flow 1 from B to D is routed through B-C-A-D, flow 2 from C to B is routed through C-A-D-B, and flow 3 from A to C is routed through A-D-B-C, there can be deadlock by these three flows.

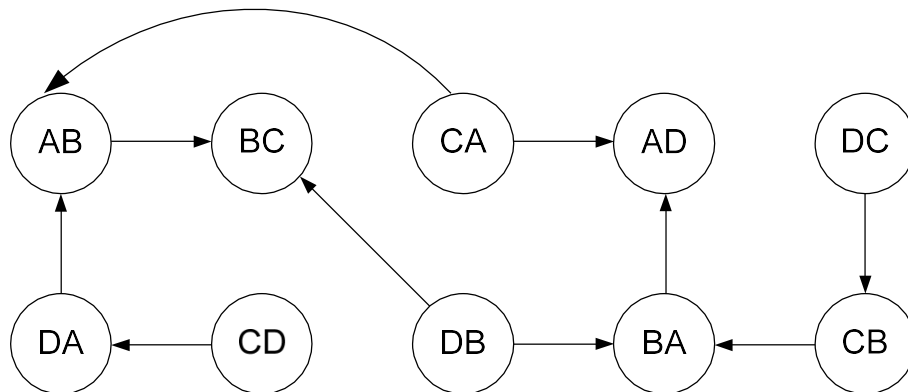
Problem M7.2.E

Still having the two paths in M7.2.D prohibited, we added another restriction in routing: the link from C to A can be used only by packets generated at C, before the packets are transferred to any other nodes (it should be the first link those packets ever take). Also, the link from D to B can be used only by packets generated at D with the same condition (however, routes may be non-minimal).

Is this routing deadlock-free? Show your reasoning and give a deadlock scenario if it is not deadlock-free.

Yes.

These conditions effectively eliminate any turns cornered at C and moving out to A, and any turns cornered at D and moving out to B. Then the CDG becomes:



This CDG is acyclic, thus this routing is deadlock-free.