Shivam Patel

COMP 343

DORDAL

Assignment 2: Chapter 2 Exercises 1,2,3,4,6,8

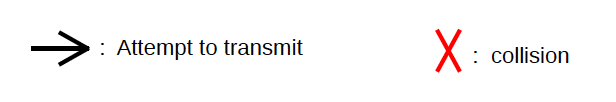
1. Simulate the contention period of five Ethernet stations that all attempt to transmit at T=0 (presumably when some sixth station has finished transmitting), in the style of the diagram in 2.1.6 Exponential Backoff Algorithm. Assume that time is measured in slot times, and that exactly one slot time is needed to detect a collision (so that if two stations transmit at T=1 and collide, and one of them chooses a backoff time k=0, then that station will transmit again at T=2). Use coin flips or some other source of randomness.

T=0 T=1 T=3 T=4 T=5 T=6 T=7

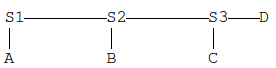
Slot 1 Slot 2 Slot 3 Slot 4 Slot 5 Slot 6 Slot 7

X

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| A1 | k=0 | X  k=3 | ------------- | ------------- | ------------- |  |  |
| A2 | k=1  X | ------------- | k=7  X | ------------- | ------------- | ------------- | ------------- |
| A3 | k=1  X | ------------- | X  k=2 | ------------- | ------------- |  |  |
| A4 | k=0  X | X  k=0 | X  k=1 | ------------- |  |  |  |
| A5 | k=1  X | ------------- | k=3 | X  ------------ | ------------- | ------------- |  |



1. Suppose we have Ethernet switches S1 through S3 arranged as below. All forwarding tables are initially empty.



(a). If A sends to B, which switches see this packet?

All the switches learn where A is

(b). If B then replies to A, which switches see this packet?

This packet goes directly to A, only S2 and S1 learn where B is.

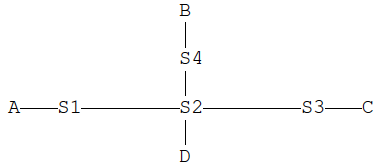
(c). If C then sends to B, which switches see this packet?

S3 and S2 see the packet and learn where C is.

(d). If C then sends to D, which switches see this packet?

All the switches see the packet since they don’t know the location of D, and S1 learns the location of C

1. Repeat the previous exercise, with the same network layout, except that instead the following packet transmissions take place:



* A sends to B

All the switches learn where A is, none know where B is

* B sends to A

This packet goes directly to A, so S4, S2, and S1 learn where B is.

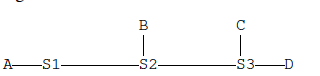
* C sends to B

Since S2 knows where B is, S3, S2, and S4 learn where C is.

* D sends to A

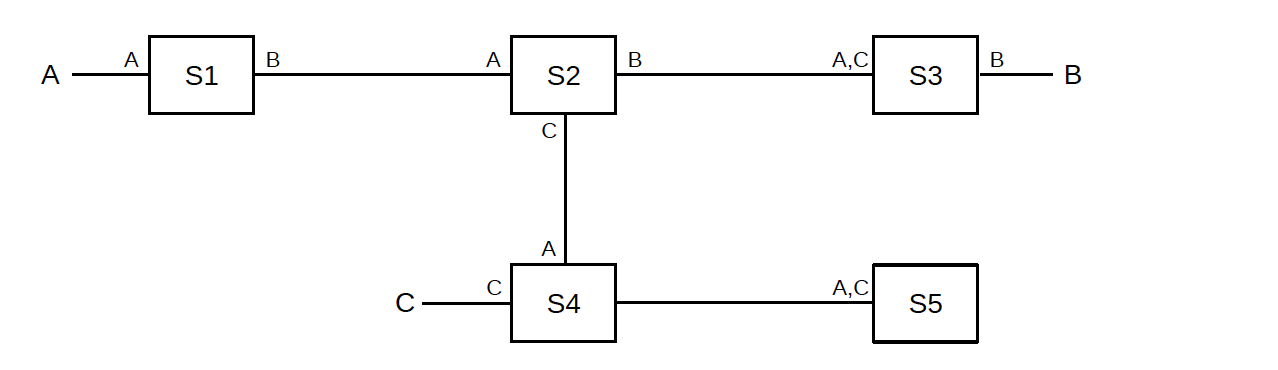
This packet will go directly to A since S2 knows where A is. S2 learns where D is.

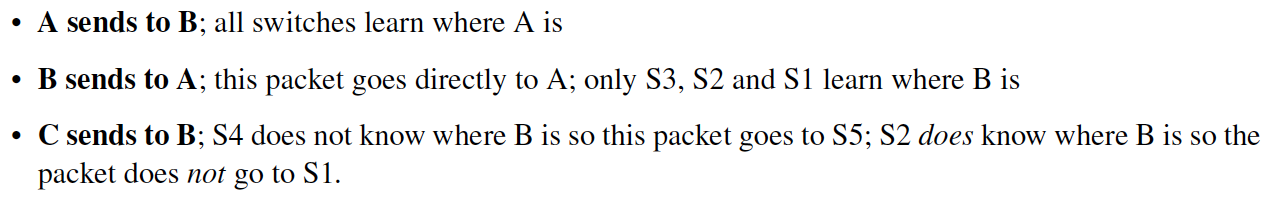
1. In the switched-Ethernet network below, find two packet transmissions so that, when a third transmission A🡪D occurs, the packet *is* delivered to B (that is, it is forwarded out all ports of S2), but is *not* similarly delivered to C. All forwarding tables are initially empty, and each switch uses the learning algorithm of 2.4 Ethernet Switches.



D 🡪 C and C 🡪 D

1. In the diagram of 2.4.1 Ethernet Learning Algorithm, suppose node D is connected to S5, and, with the tables as shown below the diagram, D sends to B.



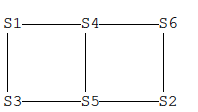


1. Which switches will see this packet, and thus learn about D?

S5, S4, S2, and S3

1. Which of the switches in part (a) do not already know where B is and will use fallback-to-flooding (ie, will forward the packet out all non-arrival interfaces)?

S4 and S5

1. The following network is like that of 2.5.1 Example 1: Switches Only, except that the switches are numbered differently. Again, the ID of switch Sn is n, so S1 will be the root. Which links end up “pruned” by the spanning-tree algorithm, and why?

S1 has the lowest ID and so it becomes the root.

S3 and S4 are directly connected, so they will enable the interfaces by which they reach S1

S6 has a unique lowest-cost route to S1, and so again by Rule 1 it will enable its interface to S4, while by Rule 2 S4 will enable its interface to S6

S5 has two choices; it hears equal-cost paths to the root from both S3 and S4. It picks the lower-numbered neighbor S3. The interface to S4 will never be enabled. Similarly, S4 will never enable its interface to S5.

S2 has two choices, S5 and S6, it chooses S5, and so the interfaces between S6 and S2 will never be enabled.

The network in effect becomes:

S1 -------------- S4 -------------- S6

|

|

|

S3--------------- S5 -------------- S2