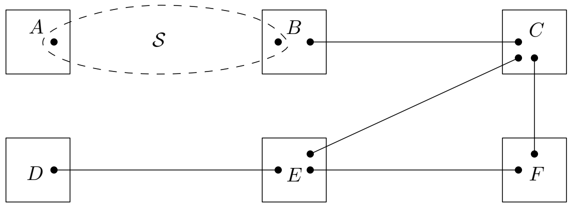
Computer Networks

Exercises List 4

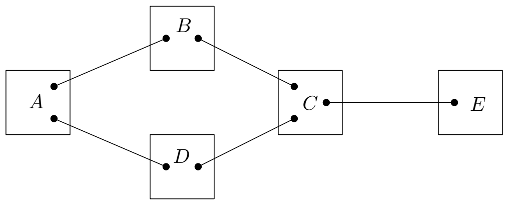
In exercise 1–4 we consider distance vectors algorithm.

1. In the given network, show (step by step) how the routing tables are constructed. How many steps does it take to reach a steady state?

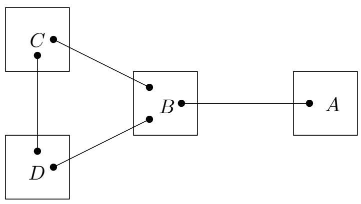


2. Assume, that the routing tables for the network above have been already constructed. What will happen if we add a connection between routers *A* and *D*?

3. In the network below, the link between routers *C* and *E* fails. Show — describing subsequent messages being sent between routers — that the routing loop can occur, even if we use split horizon techniques (with or without path poisoning).



4. Consider the network given below, in which — after steady state has been reached — the link between *A* and *B* fails.



Show, that split horizon technique (without path poisoning) allows to reach a state, in which *C* and *D* routers, as a next hop in the path to *A* router, point to each other.

Hint: Sending routing tables between *C* and *D* routers takes some time, during which their routing tables can possibly be modified. Note, that even if we use pure split horizon technique, such a state can be considered steady[1]. How the situation changes if we employ path poisoning?

5. In this exercise, we will prove that link state algorithm allows a loop to emerge in the network. In order to show that, construct a network with topology in which two routers (*A* and *B*) are not directly connected. Let's assume that all the routers know the topology of the network. At some arbitrary moment link between *A* and *B* fails, and both routers learn about that immediately. They broadcast an LSA update. Show, that during the propagation of broadcast message (the update was received by some routers, but it hasn't reached another routers) a loop may emerge.

[1] In reality, information about paths will be discarded from the routing tables, if they haven't been refreshed for a longer period of time.

6. This exercise does not refer to routing tables related problems.

Let's assume that a network consists of unidirectional links (i.e. the topology of such network is a directed graph). Consider following naïve algorithm of network flooding with a packet: the message originates from a single router; each router that receives the message, sends it over all outgoing links.

Show that:

(a) in general case, the information may circulate in the network indefinitely;

(b) cycle-free graphs of n vertices may exist, and in those graphs flooding will end after $2^{\Omega(n)}$ rounds.

Let's assume, that only one messages can be sent over a link at a time, and the time it takes for a message to travel between two adjacent nodes takes one round.

7. Draw a dozen or so of routers. Assign them with subsequent letter of Latin alphabet. Connect them arbitrarily and haphazardly group into regions. Show full and hierarchical routing tables for router A.

8. Construct a network of n routers grouped into regions. The network must satisfy following property: *for any optimal choice of routing paths – „flat" and hierarchical – exists a pair of routers, between which the distance for hierarchical routing is* $\Omega(n)$ *times bigger than for „flat" routing.*

Marcin Bieńkowski
Translation: *Krystian Bacławski*