**Distance-vector routing protocol**

*A distance-vector routing protocol is one of the two major classes of routing protocols whereas the other class is the link-state protocol. A distance-vector routing protocol notifies its neighbours about changes in the topology of the network. The other class, link-state protocol, notifies the entire network about changes in the topology of the network. In this paper we focus on the first class, distance-vector routing protocol. A lot of research has been done on this topic and many detailed papers have been written about this class of routing protocols. This paper however, will cover the fundamentals of distance-vector routing where details are left out.*

To study distance-vector routing, we have implemented a very basic distance-vector routing protocol. We started by making the protocol identifying its neighbours and accompanying link costs. This information is then put into a distance vector and the vector is stored in the routing table. After this initializing step, the protocol starts exchanging the distance-vector information. When a router receives this information from a neighbour, the protocol updates the distance vector in the routing table and recalculates the shortest cost link to its neighbors. The protocol then resends this information to all its neighbours if a shorter path has been found. After a while this algorithm converges which means that all shortest paths have been found and no more updates are exchanged between routers.

We found by testing the protocol with test cases of static networks that the protocol worked incredibly well for such a simple implementation. However when we tested the protocol with dynamic networks, i.e. some links between routers may change in link cost or even disappear, our protocol could not handle the test cases. Because if a link between two routers disappears, a router can get unreachable, wrong updates are then send back and forth into the networks where to link costs to the unreachable link are going to infinity. This is known as the counting-to-infinity problem[[1]](#footnote-1). There are workarounds for this problem, however, we did not implement these in our protocol An example of a possibly workaround is split horizon with poisoned reverse. If a router detects that one of its neighbours has become unreachable, it sets the cost to that link to infinity. Other routers in the network will known that a particular router is unreachable because the cost to that router is set to infinity.

In order to study a distance-vector routing protocol, it is very effective to implement one. It is possible to make a successful implementation that works in a lot of test cases with only a few lines of code. But this simple implementation will fail certainly for more advanced test cases involving changes in the topology of the network. If you want to get more into distance-vector routing, I suggest starting with implementing the split horizon workaround.

*Æde Symen Hoekstra*

1. A Path-Finding Algorithm for Loop-Free Routing*, J.J. Garcia-Luna-Aceves and S. Murthy, IEEE/ACM Transactions on Networking, February 1997* [↑](#footnote-ref-1)