Greetings to all,

I wanted to give you, in a nutshell, the proof as to why distance vector routing, regardless of the initial state of its routing tables and its cost tables, will converge to a state in which all nodes have the least-cost path to the destination.

For simplicity, we will consider only a single destination d. The argument can be repeated for any other destination d'.

Let the "true cost" of a node be its shortest distance to d. We will assume (for simplicity) that the cost of each link is just 1 (hence shortest distance = minimum hop)

The proof is by induction. We want to show that eventually, for any i, where i is a nonnegative integer, all nodes whose true cost is i (or less) have the correct values in its cost and routing tables, and all other nodes whose true cost is greater than i have in their cost table a cost greater than i.

Base case i = 0. I.e., d must end up with a cost of 0 to reach itself and all other nodes must have a cost of at least 1.

Proof of base case:

Notice than in the action where you receive an upd message you always set the cost to reach yourself to be zero. Hence, d will set its cost to reach itself to be zero.

Consider any node, say p, other than d. We must show its cost to d becomes at least 1 and remains at least 1.

1. Assume p updates its cost by receiving an upd(c) message from some neighbor. P will then set cost[d] := c[d] + 1. Note that c[d] >= 0, and hence, cost[d] >= 1 as desired.

2. What if p does not change its cost and its cost is zero? This would be a problem. However, let rtb[d] = q at p (q is the next hop). Eventually q sends an upd(c) to p, and from the code, since q is the next hop, p believes what q says, and so it sets cost[d] := c[d] + 1. Again, c[d] >= 0, and hence, cost[d] >= 1 as desired.

Proof of Induction Step:

Assuming that for any i, where i is a nonnegative integer, all nodes whose true cost is i (or less) have the correct values in its cost and routing tables, and all other nodes whose true cost is greater than i have in their cost table a cost greater than i,

we must prove that,

all nodes whose true cost is i+1 have the correct values in its cost and routing tables, and all other nodes whose true cost is greater than i+1 have in their cost table a cost greater than i+1.

From the assumption, we don’t have to worry about nodes whose true cost is i or less.

Consider any node, say p, whose true distance is i+1. We show it obtains (and retains) the correct cost and routing table entry.

Since p has true distance i+1, it has at least one neighbor, q, whose true distance is i. (We will assume it only has one such neighbor, it could have more but it does not change the proof below much). From the induction hypothesis, q has a cost of i (and retains this cost).

We have three cases to consider.

1. At p, rtb[d] = q and cost[d] = i+1. This is stable, because the cost sent by p will always be i, and any other neighbor will send a cost of at least i+1 (by the induction hypothesis) hence p will keep pointing to q and remain with cost of i+1 and we are done.

2. At p, rtb[d] = q and cost[d] > i+1. If p receives an upd message from q, then p will choose q as its next hop and we will have case 1 above (note that this message will eventually be received since q sends messages periodically). However, p could receive an upd message from another neighbor g, and the cost of g plus 1 is perhaps better than p’s current cost. From the induction hypothesis, the cost of g is at least i+1, hence the new cost of p is at least i+1, and we have the following case.

3. At p, rtb[d] != q and cost[d] > i+1. In this case, Any neighbor of p, other than q, will offer a cost of at least i+1 (by hypothesis) and hence, as long as rtb[d] != q, p’s cost will remain greater than i+1. Eventually, q will send an upd message to p with a cost of i, and hence, p will choose p as its new next hop, which from case 1 we know is stable.

We are done with p.

Now consider any node x whose true distance is greater than i+1. We must show its distance becomes (and remains) greater than i+1. Note that all neighbors of x have a true distance at least i+1.

We have two cases.

1. Assume x updates its cost by receiving an upd(c) message from some neighbor. x will then set cost[d] := c[d] + 1. Note that c[d] >= i+1 (from hypothesis), and hence, cost[d] > i+1 as desired.

2. What if x does not change its cost and its cost is i+1? This would be a problem. However, let rtb[d] = y at x (y is the next hop). Eventually y sends an upd(c) to x, and from the code, since y is the next hop, x believes what y says, and so it sets cost[d] := c[d] + 1. Again, c[d] >= i+1 (from hypothesis), and hence, cost[d] > i+1 as desired.

End of Proof.