

CSc72010

Cisco Spanning Tree

Review

BFS Complexity:

Time: $diam$

Messages: $|E|$

BFS Applications

Broadcast: Piggyback the actual message on the search messages

Child pointers: When receiving a search message, respond with either a parent or non-parent message.

This can be used for return messages.

If communication is not always bi-directional, may need to run another instance of SynchBFS to reply.

Broadcast/convergecast: When leaves receive a message, send to parents – eventually, start node gets all replies.

Leader election: Use broadcast/convergecast to determine max (or min) UID in network.

Computing the diameter: All processes do BFS, determine $max-dist_i$ from i to any other process by piggybacking distance information. Then broadcast/convergecast to get the largest distance in the graph.

Spanning Tree

Note that BFS computes a spanning tree (the parent pointers identify the edges). How many edges are there in a spanning tree?

The Cisco STP uses this.

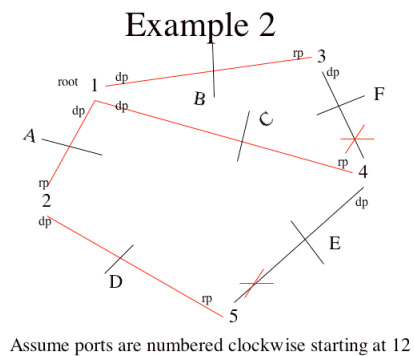
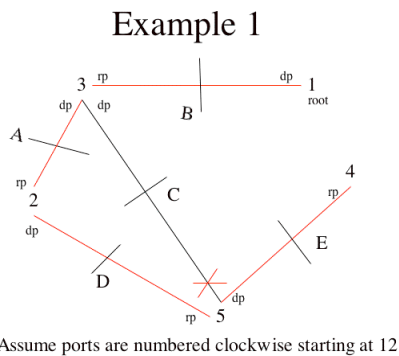
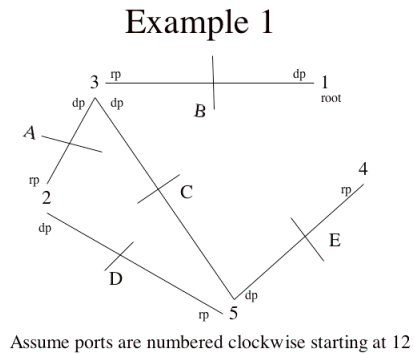
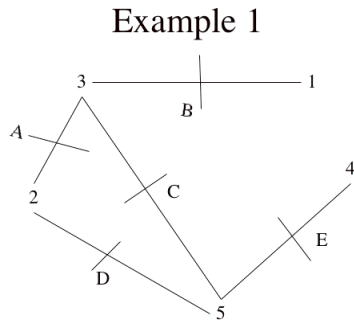
What does the Cisco STP compute

When the protocol stabilizes, the state should be as follows:

- 1) **Root bridge:** The process (switch) with the lowest MAC address (or lowest combined priority+MAC address) is the root. This uses a leader election algorithm.
- 2) **Root ports:** Each bridge has one root port. The root port on each bridge is the port of the bridge with the smallest distance from the root. If two ports are equidistant from the root, then the one going to the bridge with the lower MAC address is the root port. This uses a breadth-first search, if we assume rounds; however, if the network is asynchronous, it's more complicated.
- 3) **Designated ports:** Each network segment (connecting bridges) has a designated port. Messages put on that network segment are forwarded to the rest of the network through the designated port. The designated port is on the bridge closest

to the root. If there is a tie, it is on the bridge with the lowest MAC address. If the bridge selected by this rule has multiple ports on a network, it is the port with the lowest id.

Examples



Cisco Version

All processes send a BPDU (Basic Protocol Data Unit) at each round (actually, default is every 2 seconds – but we will describe this as a synchronous algorithm, running in rounds). The BPDU contains the id (MAC address) of the sending process, the id of the process it thinks is the root, and the distance from the sending process to its presumed root.

id:Int
 id:Int
 cost:Int

When a process receives a BPDU, it compares the ID of the root (designated by the neighbor that sent the BPDU) to the local value for the ID of the root. If the new root ID is lower, it replaces its local root ID with the new one and adds one to the cost in the incoming BPDU and makes that its cost to the root. If it receives different BPDU's

having different roots, it uses the “best,” i.e., the one with the lowest root ID and the lowest distance (if root IDs are the same).
Finally it designates the port to which the sending neighbor is connected as the root port. (Effectively choosing its parent)

Note that this is essentially BFS.

Designated ports are then chosen: for each network it is on, the bridge checks incoming BPDUs. If its own cost is lowest, or if it is tied with another bridge on cost and its id is smaller, it is the designated bridge, and its port on that network segment is the designated port. If it has multiple ports on the network segment, it chooses the one with the lowest id.

Shortest Paths

Assume now that we have directed weighted links. Each node knows the weights of all incident edges and the number of nodes in the graph.

Motivation is min-cost communication

Problem: Determine the shortest path from i_0 to each other node.

This algorithm is the basis of Bellman-Ford, aka RIP (Route Information Protocol). In the presence of failures this is an unstable routing protocol and isn't much used.

Each node keeps track of the shortest known distance from i_0 and updates it as new information arrives; also communicates changed distances.

states_{*i*}
dist: Int := infinity for $i_0 \neq i$, 0 for i_0
parent: Int := undefined

messages_{*i*}
send dist_{*i*} to all out-nbrs

transitions_{*i*}
if $\text{weight}_{ji} + \text{dist}_j < \text{dist}_i$ for some in-nbr j of I , then update dist_{*i*} to $\text{weight}_{ji} + \text{dist}_j$ and update parent to j

Terminates after $n-1$ rounds

Invariant Assertion:

At round r , $0 \leq r \leq n-1$, every process i has its dist and parent components set to the shortest path values among those paths consisting of at most r edges.

Complexity:

Time = $n-1$

Messages: $(n-1) \cdot |E|$

Minimum Weight Spanning Tree

Next time