An Efficient Multicast Tree Computation Algorithm for 802.1aq

Ali Sajassi

March 16, 2008

IEEE Plenary Meeting

IEEE Congruency Requirements

1. Forward & Reverse paths must be congruent. If not, it can
   - impact CFM & CM procedures
2. Multicast & unicast paths in the same direction must be congruent. If not congruent, then it can cause
   - Out-of-order delivery
   - Issues with Ethernet OAM (802.1ag)
   - Black holing in customer's network
   - Loop creation in customer's network

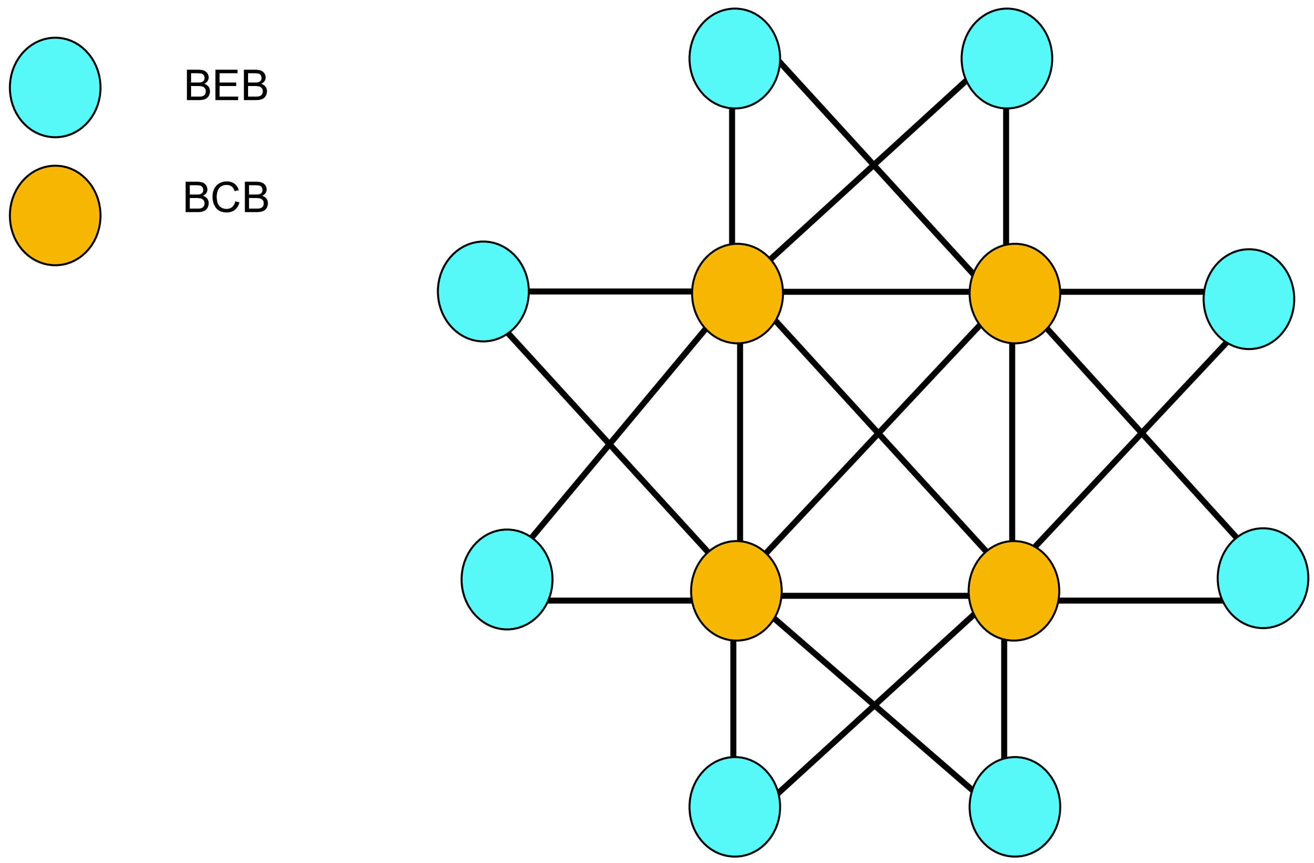
Assumptions

- Forward & Reverse Metrics are the same
- Bridges are connected via P2P links (no hubs in between)
- In a multicast group,
  - Every group member is potentially a source and
  - Every source is a group member

Agenda

- Brute-force method for multicast trees computation w/ congruency
- An efficient method for multicast trees computation w/ congruency

Congruency between Forward & Reverse Paths

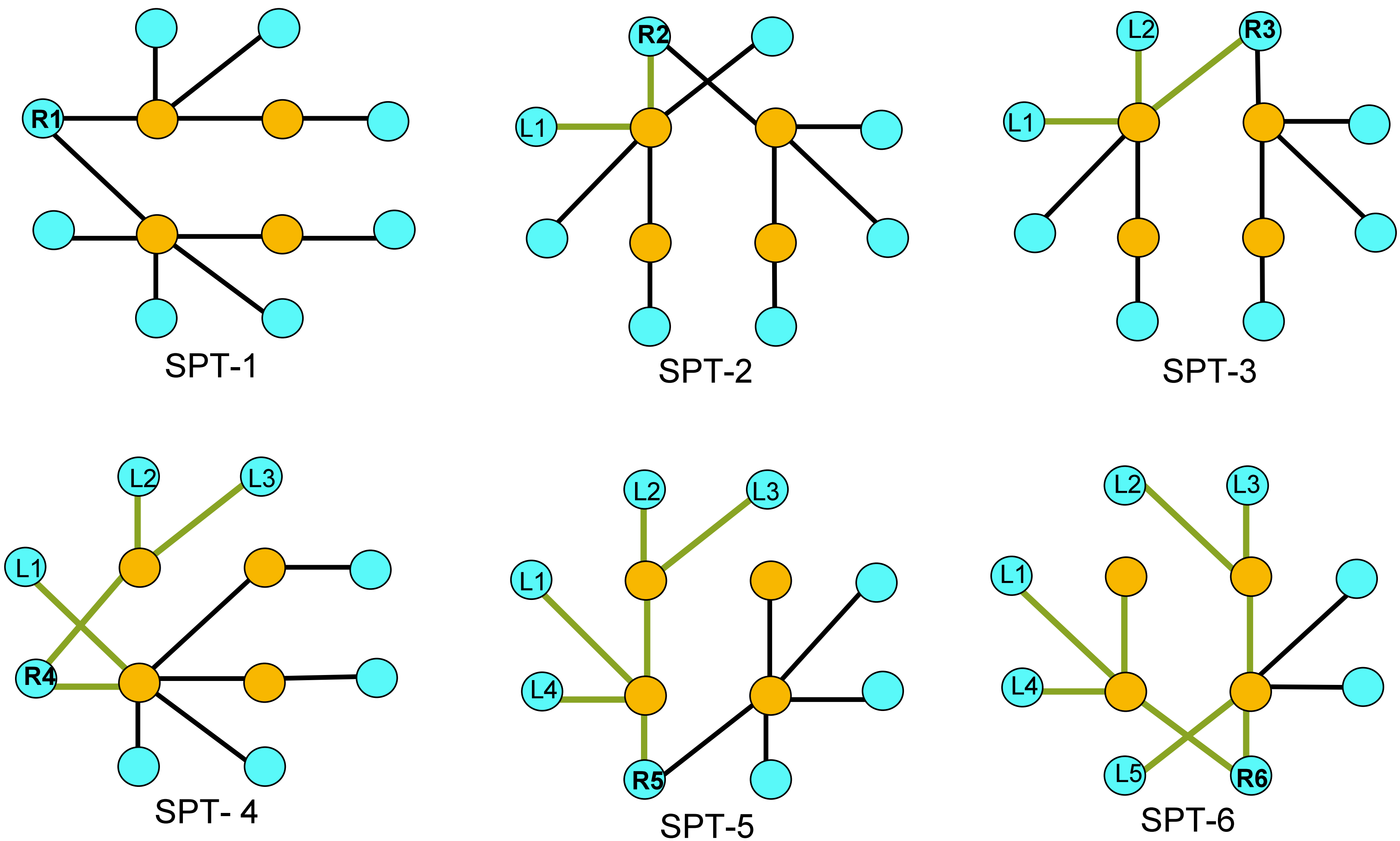


Each Bridge in PBBN runs link-state protocol (IS-IS) and thus has a full picture of the network topology

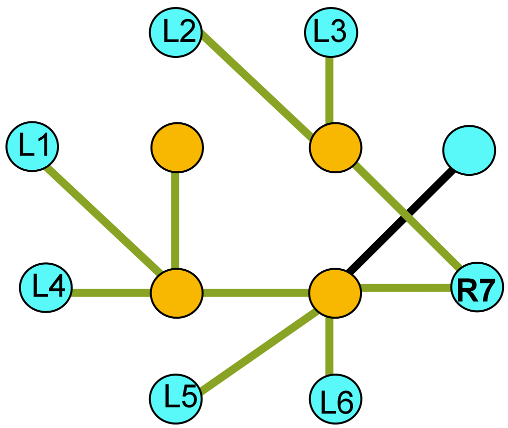
Example Only – Building Congruent Trees

- Each bridge builds its SPT (one per BEB) in an ordered way – e.g., starting with higher BEB IDs (or lower BEB IDs) which translates into higher B-VIDs (or lower B-VIDs) since B-VIDs are used as SPT identifier
- After building the 1st SPT tree for a BEB, then the 2nd SPT tree (next higher or lower B-VID) is build such that it uses the same branch between the two roots as in the 1st tree
- Next the third tree is built such that the first two branches are the same as the ones in 1st and 2nd tree
- And so on till all SPT trees are built on a given node
- Since all nodes follow the same algorithm, all the SPTs trees built in this way on each node are exactly the same as any other nodes – e.g., the same set of SPT trees are computed on every node.

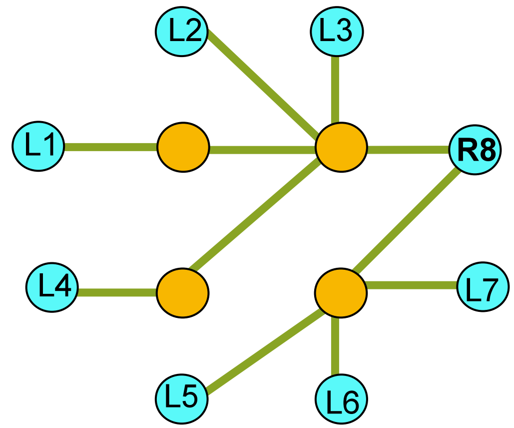
Example of Building Congruent SPT Trees



Example of Building Congruent SPT Trees - II



SPT-7



SPT-8

As it can be see in these figures, the path from root Rx to leave Ly on SPT-x is the same as the path from root Ry on SPT-y to leaf Lx

Agenda

- Brute-force method for multicast trees computation w/ congruency
- An efficient method for multicast trees computation w/ congruency

Components

- This proposal uses only a single tree per intermediate node as opposed to having N trees (one rooted at each edge bridge). In other words, in terms of # of trees computed by IGP, this proposal is as efficient as IP while meeting congruency objectives
- There are two components to this proposal
  a) Deterministic algorithm in presence of ECMP
  b) Multicast replication using the unicast topology tree at the intermediate node (a single tree as opposed to N trees)

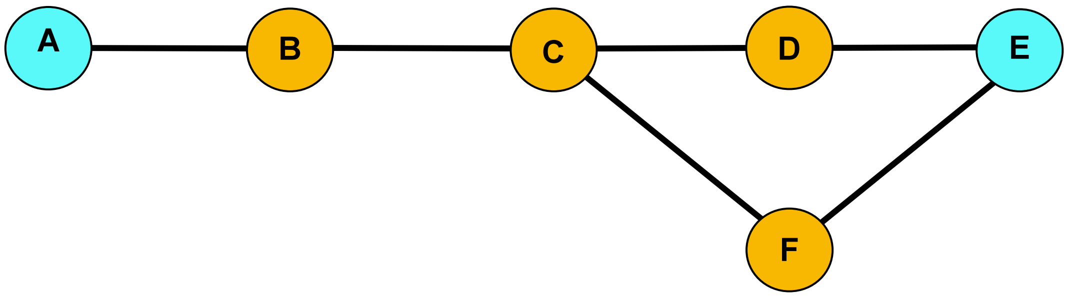
Deterministic Algorithm in Presence of ECMP

- When a router computes SPF, it iterates node from three lists:
  - Unknown list
  - Tentative (TENT) list
  - PATHS list
- When more than one path are discovered (for a given node) the router checks the system-ID/Router-ID of each node in the path and selects the path that contains the lowest (or highest) ID
- When a node is to be moved into TENT list, the router checks if the node already exists in TENT list (if so, ECMP exits).
- If ECMP exits, the node checks the set of IDs of the path and selects the best one (according to lowest/highest ID present in path)

Deterministic Algorithm in Presence of ECMP – Cont.

- For hop by hop forwarding, the algorithm can be optimized further for the router to just remember the highest/lowest ID seen while building the path as opposed to remembering the complete path and searching along the path to find the highest/lowest router-id.
- Using the above algorithm, a packet using hop by hop forwarding along the computed routes IS guaranteed to be symmetric.

Deterministic Algorithm in Presence of ECMP – Example



Where the numerical values of the node IDs are in the sequence A<B<C<D<E<F

- When A computes a path to E, the LOWEST ID it sees in the path is B and so cannot choose between the paths through D or F
- However, when C computes its path to E it WILL chose the lower of the two paths via D, and hence a hop by hop forwarded packet will follow the symmetric path

Multicast Replication Using Unicast Tree

- Each router advertises inside its LSP the multicast groups that it is interest in and its router-id as the source for these groups. It has to be noted that these groups are pretty stable and this will not trigger massive flooding in ISIS (specially in a 802.1ah provider network)
    - These advertisement are in the form of (S,G1-G20, G30, G40)
- When a node receives mcast LSP from its neighbor, it installs the mcast group addresses on the interface that has passed RPF check – e.g., on the interface toward the source (e.g., they get installed on the upstream interface and NOT downstream interfaces).
- This interface is marked as IIF for this source. There can only be a single IIF per source at a given node.
- An interface that is marked as IIF for (S,G), would also be marked as OIF for other sources of the same group – e.g., other (S',G) where S' != S
- For a node to transmit a multicast stream over an interface, then that interface must be in OIF list of that (S,G)

Pruning Unwanted Branches

• When a node add an interface to the OIF list for (S,G), it then sends a prune message over it.

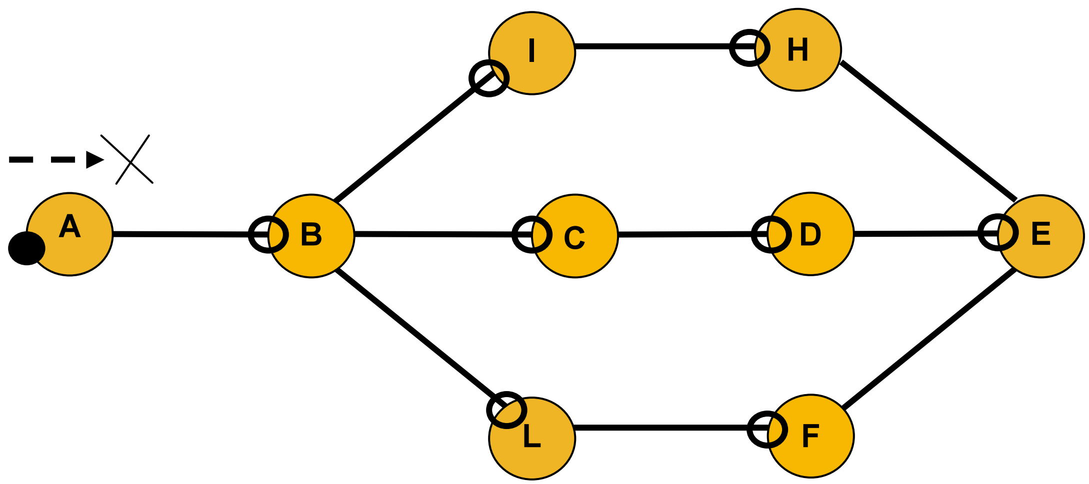
• If a neighboring node receives this prune message for (S,G) over its IIF interface for that S, then this prune message gets discarded because this message has been sent to a node which is on the shortest path toward S.

• If a neighboring node receives this prune message for (S,G) over a non-IIF interface for that S, then it means that this node is not on the shortest path toward S and thus it processes this message and removes that interface from OIF list for that (S,G) and sends a prune message back to the sender.

• The sender upon receiving this prune message, prunes its interface and if there are no more interfaces in OIF list and no G is being configured locally, then the prune message is propagated upstream over its IIF for (S,G).

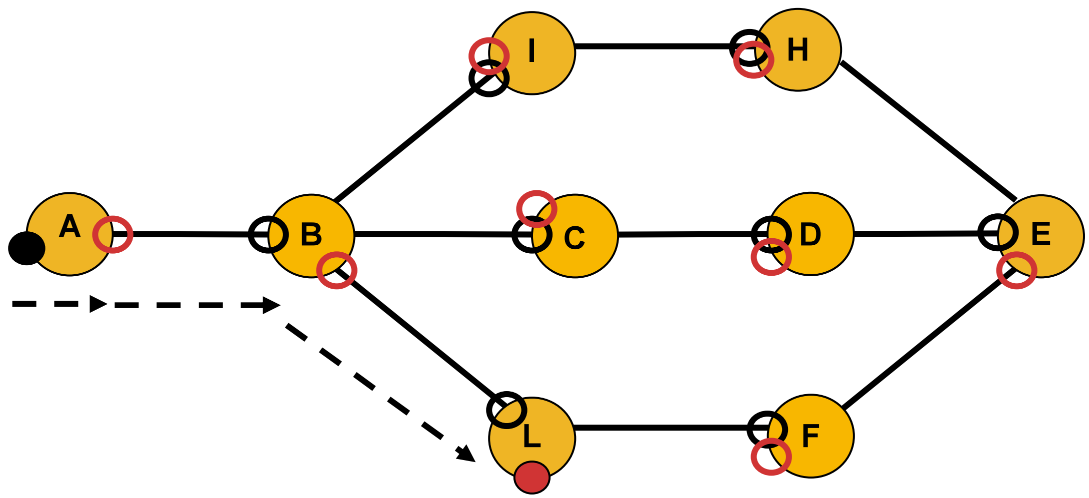
• Prune messages are encoded into hello packets of the routing protocol (ISIS or OSPF).

Example – Configuring



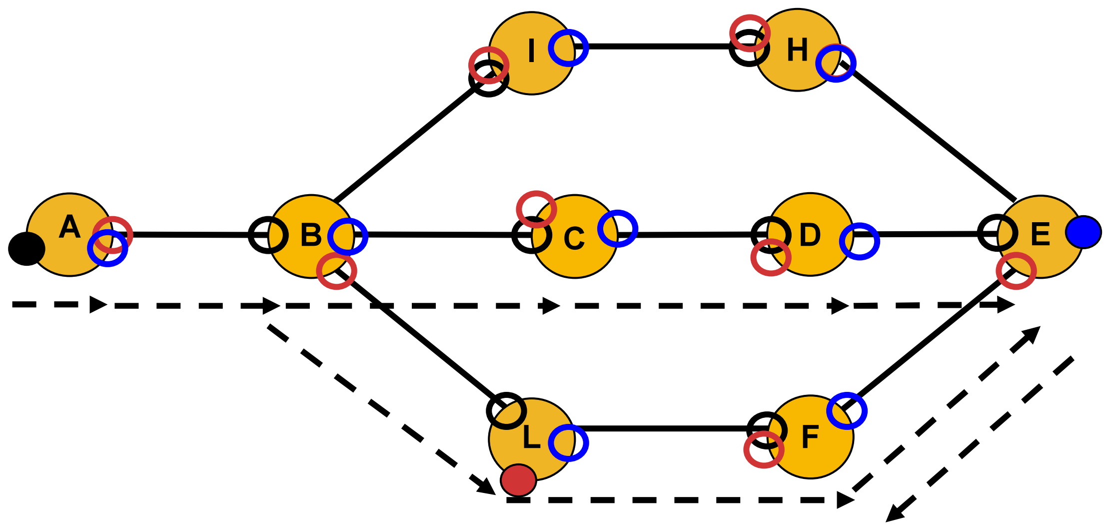
- Node A is configured with mcast group G
- Node A sends (A,G) in its LSP
- Each node upon receiving this LSP, configure its RPF interface as IIF for (A,G) – e.g., black circles. There is only one such interface at each node even in the presence of ECMP
- Since there is no interface configured as OIF in node A, no multicast stream (A,G) gets sent out of A

IIF & OIF Marking - Example



- Now, node L is also configured with group G
- Node L sends (L,G) in its LSP
- As before, each node upon receiving this message, configures one of its interfaces as IIF corresponding to (L,G) – e.g., red circles.
- The IIF of (L,G) is considered as the OIF of (A,G) and it is added to OIF list for (A,G). Also, a prune message for (A,G) is sent out over this newly added OIF.
- If the recipient node of this prune message, receives it over its IIF for (A,G), then it discards the message; otherwise, it removes that interface from OIF list.

Example – Adding a 2nd node to the Group



• Now, node E is also configured with group G

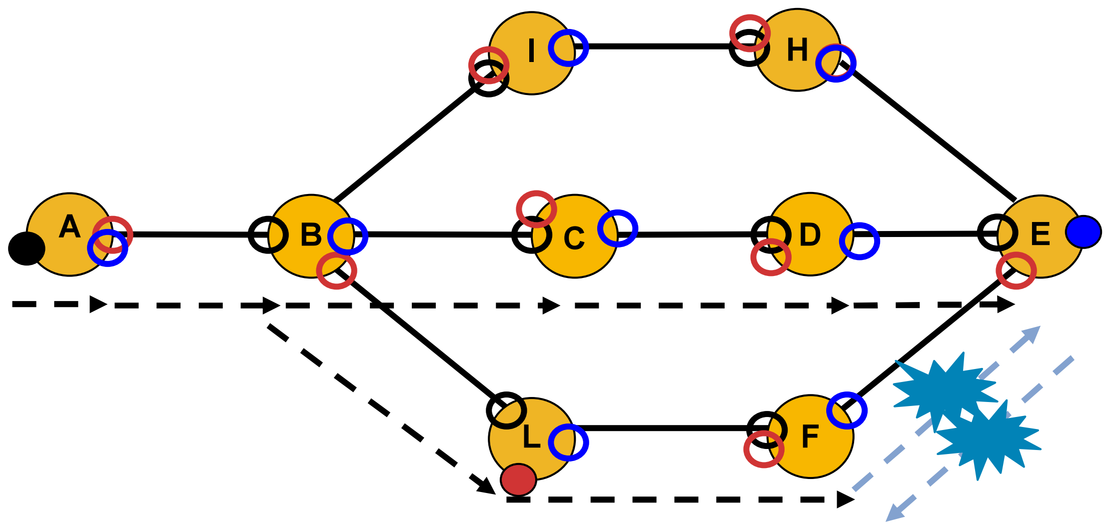
• Node E sends (E,G) in its LSP

• As before, each node upon receiving this message, configures one of its interfaces as IIF corresponding to (E,G) – e.g., blue circles.

• The IIF of (E,G) is considered as the OIF of (A,G) and it is added to OIF list for (A,G). Also, a prune message for (A,G) is sent out over this newly added OIF.

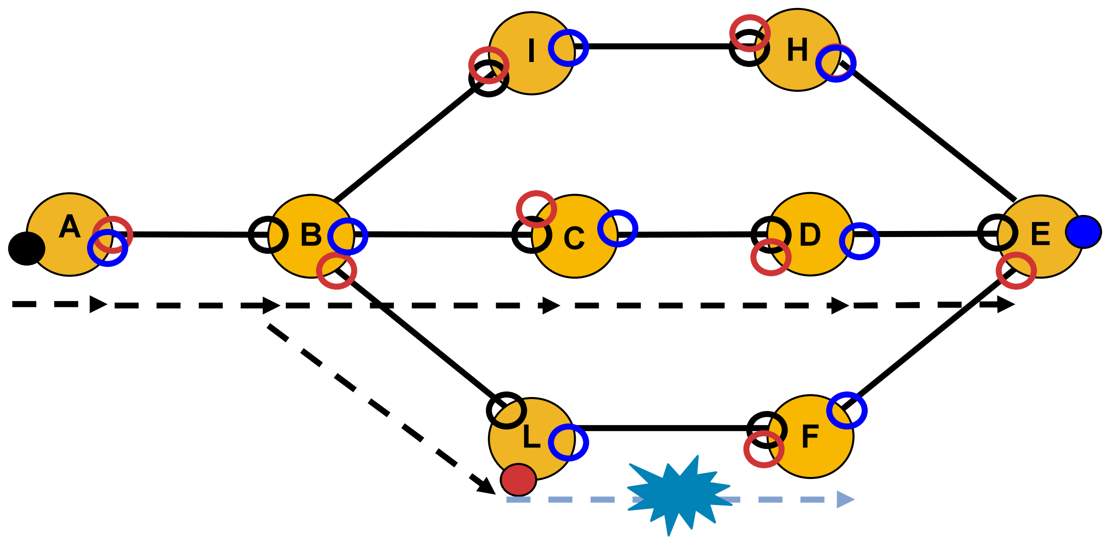
• As before, if the recipient node of this prune message, receives it over its IIF for (A,G), then it discards the message; otherwise, it removes that interface from OIF list.

Example – pruning unwanted interfaces



- Node F adds if(fe) as OIF for (A,G) and sends a prune message over it
- Node E upon receiving this prune message checks to see that if(ef) is not an IIF for (A,G) and then prunes this interface from OIF list of (A,G)
- Node E upon pruning if(ef) interface, it sends a prune message over this interface to node F
- Node F upon receiving this message, checks to see that if (fe) is not an IIF for (A,G) and then prunes this interface from OIF list of (A,G)
- Node F checks to see if its OIF list is empty, if so, then it propagates the prune message to its upstream node L over its IIF

Example – Propagating Prune msg



- Node L receives the prune message from node F over its non-IIF interface and thus prunes if (lf)
- Since group G is configured in node L, node L does not propagate this message any further

Pruning

- Pruning is done in two stages

  a) By installing only a single IIF based on shortest reverse path, the first level of pruning is performed since only a single interface is selected as IIF even in the presence of ECMPs – e.g., a path with no receiver will not receive the multicast stream because a head-end node doesn’t have any OIF for that path. This efficiency again is the side affect of congruent forward/reverse path assumption

  b) The second level of pruning is performed by sending explicit prune messages (embedded in hellos) to the neighbors

Signaling Efficiency

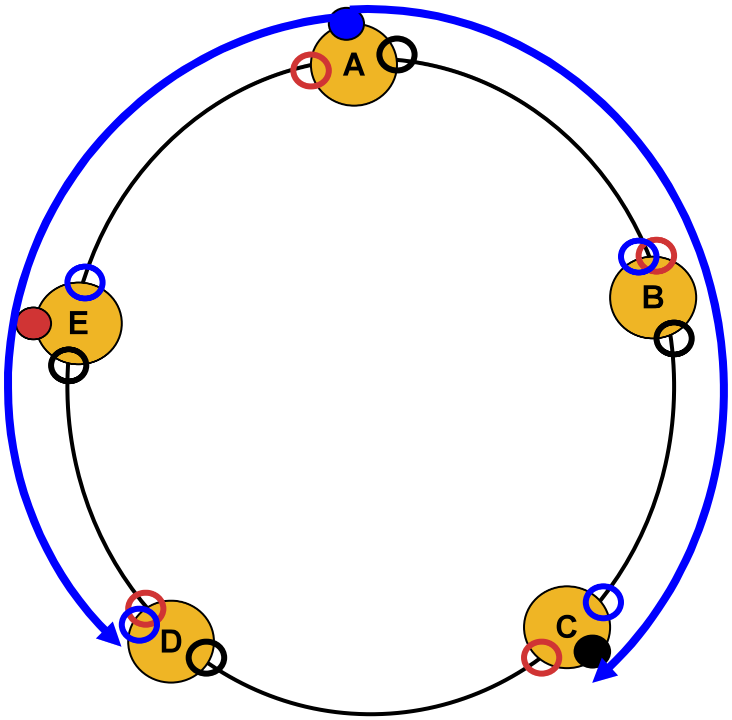
- In terms of signaling used for join & prune, this scheme is more efficient than
  a) PIM-DM: because of the first level of pruning, the multicast data does not go everywhere – it only goes to the nodes which have receiver. The second-level of pruning ensures that these receivers only receive the multicast stream once.
  b) PIM-SM: It doesn't wait for the signaling to be propagated from the receiver. Because of congruent forward/reverse path assumption, the signaling can originate right away from the source using LSP messages

Optimization for Prune Msgs

- Don't send prune messages to your neighbors if you know the prune message gets discarded
  - The prune message is discarded if it is received over IIF for that (S,G)
- If the previous example, if E sends the list of nodes that can be reached via its if(ef) to F (via hello message), then F knows ahead of time whether to prune if(fe) without sending a prune message to node E.
  - If node F knows that A cannot be reached via if(ef) by E, then F knows that it should prune if(fe)

Appendix A:
Congruency Requirements

Ring Topology - Example



The above algorithm works for all types of topologies including Ring topology.