A Distributed Algorithm for Constrained Flooding of IGP Advertisements

`draft-allan-lsr-flooding-algorithm-00`

Dave Allan, Ericsson

What the draft is all about

- The general problem discussed in draft-li and others is how to produce a constrained flooding topology for the IGP for dense graphs
  - immune to single failures
  - Reduces the number of copies of flooded LSAs received by an IGP speaker
- This draft discusses a distributed algorithm for computing a flooding topology with desirable properties for
  - Bi-partite style dense graphs
    - Common link metrics for inter-tier links between a pair of tiers
  - Modified bi-partite graphs with intra-tier links
  - It MAY be applicable to other topologies, this is FFS

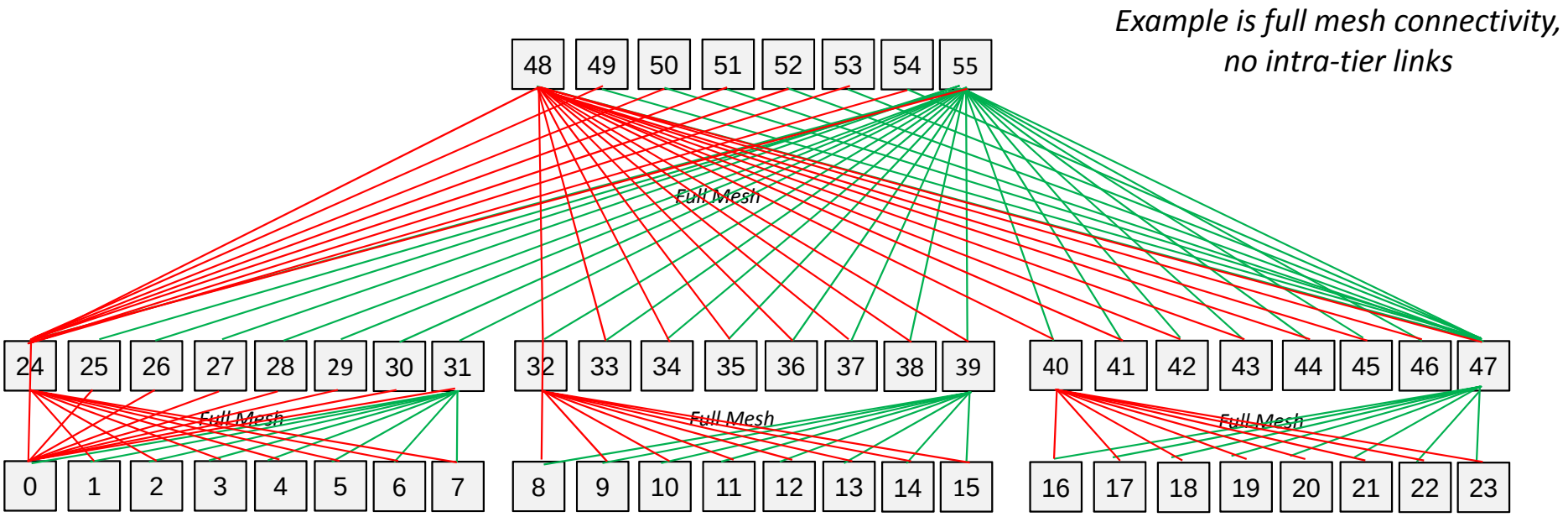
Approach

- Use two diversely rooted spanning trees constructed such that each node in a dense graph is bi-connected to the flooding topology
  - Spanning trees are computed by each node from information in the IGP
- The flooding topology is the sum of the spanning trees
  - So the first copy of an LSA received by a node is the one propagated on both trees irrespective of how it arrived
  - The actual flooding is split horizon between upstream and downstream for LSAs received from an upstream interface
    - vs. the traditional “flood on all interfaces but that of arrival”
- Net result is in a fault free network, all nodes participating in the flooding topology will receive two copies of a flooded LSA

What Makes it Work

- The tie breaking algorithm for spanning tree construction is the “secret sauce”
  - Lifted from 802.1aq
- Use a ranking of lexicographically sorted list of node IDs to tiebreak when multiple equal cost paths are found
  - XORing the list with an “algorithm mask” of zero or -1 prior to ranking allows “bookends” of diversity to be selected, so each tree is constructing using one of the algorithm masks
- The result is at each set of nodes that are equidistant from the root, the low and high node IDs are selected as the transit nodes to the next tier consistently when constructing the two spanning trees
  - Tree using algorithm mask 0 will always select the low node ID
  - Tree using algorithm mask -1 will always select the high node ID
- The ability to incrementally tie break in a consistent fashion also makes the algorithm quite frugal → any portion of the shortest path is still the shortest path

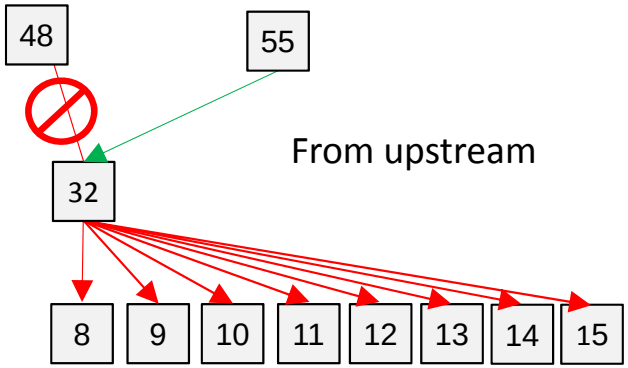
A Visual Representation of the results

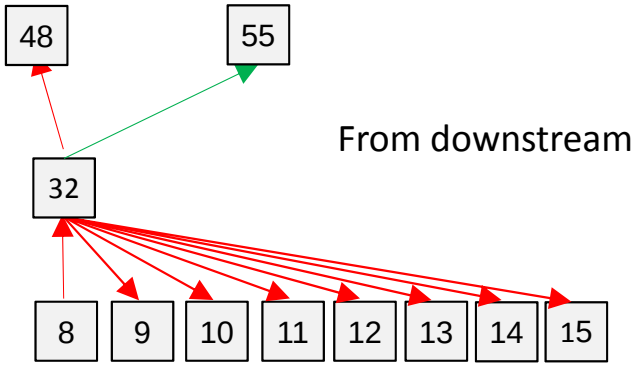


The blueprint for the flooding topology is two diversely rooted spanning trees.
In the illustrated example they are nodes 0 for the red and 55 for the green trees.
The red spanning tree was constructed with algorithm mask 0
The green spanning tree was constructed with algorithm mask -1

Flooding Rules

- Quite simple
  - If received from an upstream adjacency, and a new LSA, flood on downstream member adjacencies and non participating adjacencies
  - If received from a downstream adjacency and a new LSA, flood on all non participating adjacencies and all member adjacencies except the adjacency of arrival
  - If received on an ambiguous adjacency (upstream for one, downstream for the other), treat as if from a downstream adjacency





Required Protocol Changes

- The ability for a node to advertise capability to participate in the flooding topology via the IGP
- Knowledge of the roots, ideally advertised by the IGP
  - Or sufficient information to allow distributed root election

- The draft does NOT document the changes, it simply identifies the requirements

Limitations

- Draft-li-dynamic-flooding identified the desirability of limiting the degree of any node in the flooding topology

- This algorithm does ***not*** do that
  - The transit nodes in each tier will generate copies of the LSA to be flooded to the next tier with the degree of the physical topology
    - Note that this is asymmetrical
      - Some nodes need to flood a copy to the number of peers in the physical topology
      - Nodes only receive two copies

  - This does limit the maximum diameter of the flooding topology to 2 times the # of tiers in a fault free network
    - Flooding topology diameter corresponds to worst case physical diameter

Draft Contents

- The draft discusses:
  - Problem space
  - Algorithm for constructing the flooding topology
  - Flooding rules
  - Root selection
  - Interactions with non-participants in the flooding topology
  - Flooding topology reoptimization
  - Suggests some strategies for dealing with catastrophic multiple failures

- The draft does not
  - Define protocol elements → simply discusses what is needed
  - Define root selection procedures → only defines requirements for root selection

Summary – Characteristics of the Solution

- Structure = interconnected 1+1 multipoint to multipoint trees
- Protocol changes required:
  1. advertisement of the two roots
  2. advertisement of FT capability
- Computation of the flooding topology = order (2 x N(lnN))
- Max diameter (hops)
  - Fault free worst case = (2 x distance leaf to spine)
  - Single failure worst case = (2 x distance leaf to spine) + (distance between roots - 1)
    - Maximum diameter for a bipartite style graph occurs when an inter root member link fails
- Typical & maximum number of LSAs a node will receive = 2
  - This is within the flooding topology, and exclusive of boundary nodes with a legacy flooding domain

Next Steps

- Update root selection criteria
  - Sidebar discussions revealed that distance between roots matters with respect to diameter of the flooding topology in some failure scenarios
    - Ideal distance therefore is 2 hops

- Collect more feedback and figure it out from there