

# OSPF & ISIS Flooding Reduction

draft-cc-ospf-flooding-reduction-00

draft-cc-isis-flooding-reduction-00

Huaimo Chen ([huaimochen@huawei.com](mailto:huaimochen@huawei.com))

Dean Cheng ([dean.cheng@huawei.com](mailto:dean.cheng@huawei.com))

Mehmet Toy ([mehmet.toy@verizon.com](mailto:mehmet.toy@verizon.com))

Yi Yang ([yyietf@gmail.com](mailto:yyietf@gmail.com))

# Introduction

## ➤ Motivation

- Flooding EVERY link, reduces scalability and slows convergence
- Propose a solution to reduce flooding and have a better IGP

## ➤ Flooding Topology

Given an entire network topology in an IGP area, a flooding topology is a sub network topology that all nodes are interconnected but with less links

## ➤ Revised Flooding Procedure (Requirements)

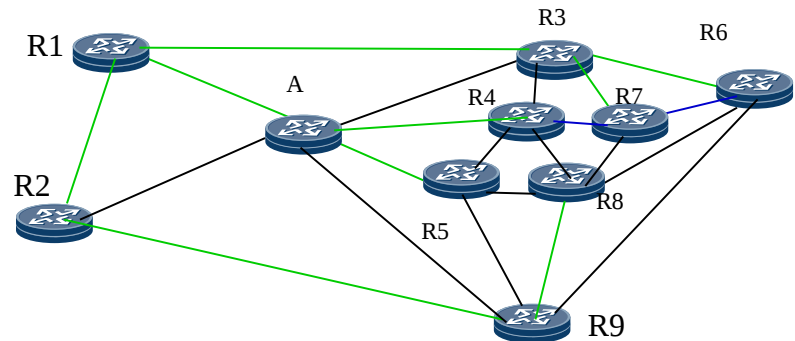
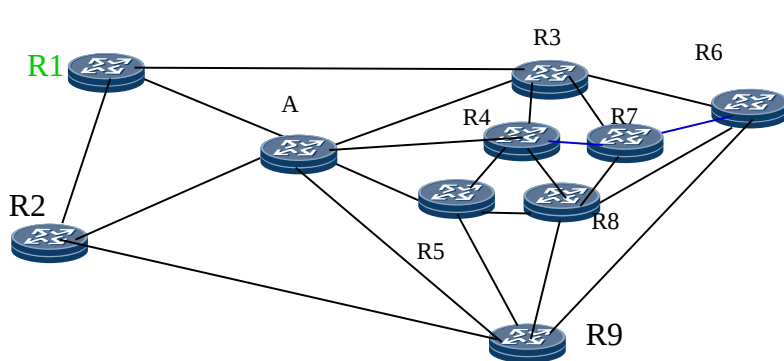
1. MUST be backward compatible
2. **MUST flood a Link State (LS) to every node in network in any case**, including
  - $n$  ( $n > 1$ ) nodes down
  - $m$  ( $m > 1$ ) links down
3. MUST reduce (almost minimize) LS flooding

# Build Flooding Topology: Brief

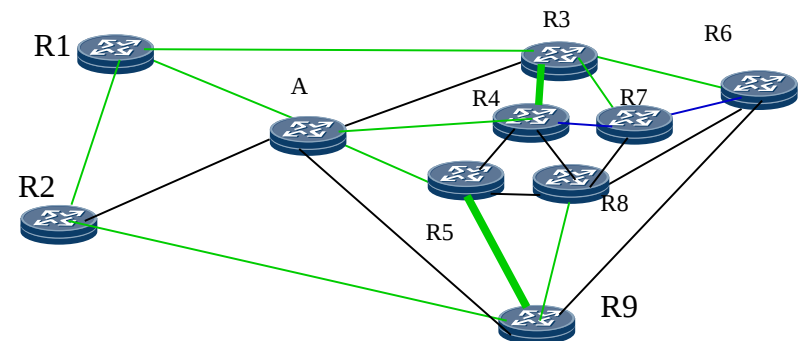
➤ Many algorithms may be proposed. A simple and efficient one:

1. Select a node R according to a rule such as the node with the biggest/smallest node ID
2. Build a tree using R as root of the tree in  $O(N)$
3. and then connect  $k$  ( $k \geq 0$ ) leaves to the tree to have a flooding topology

Each node in the network uses the algorithm to generate a flooding topology itself, **no flooding topology is flooded in the network.**



Link on flooding topology



# Build Flooding Topology: Build Tree

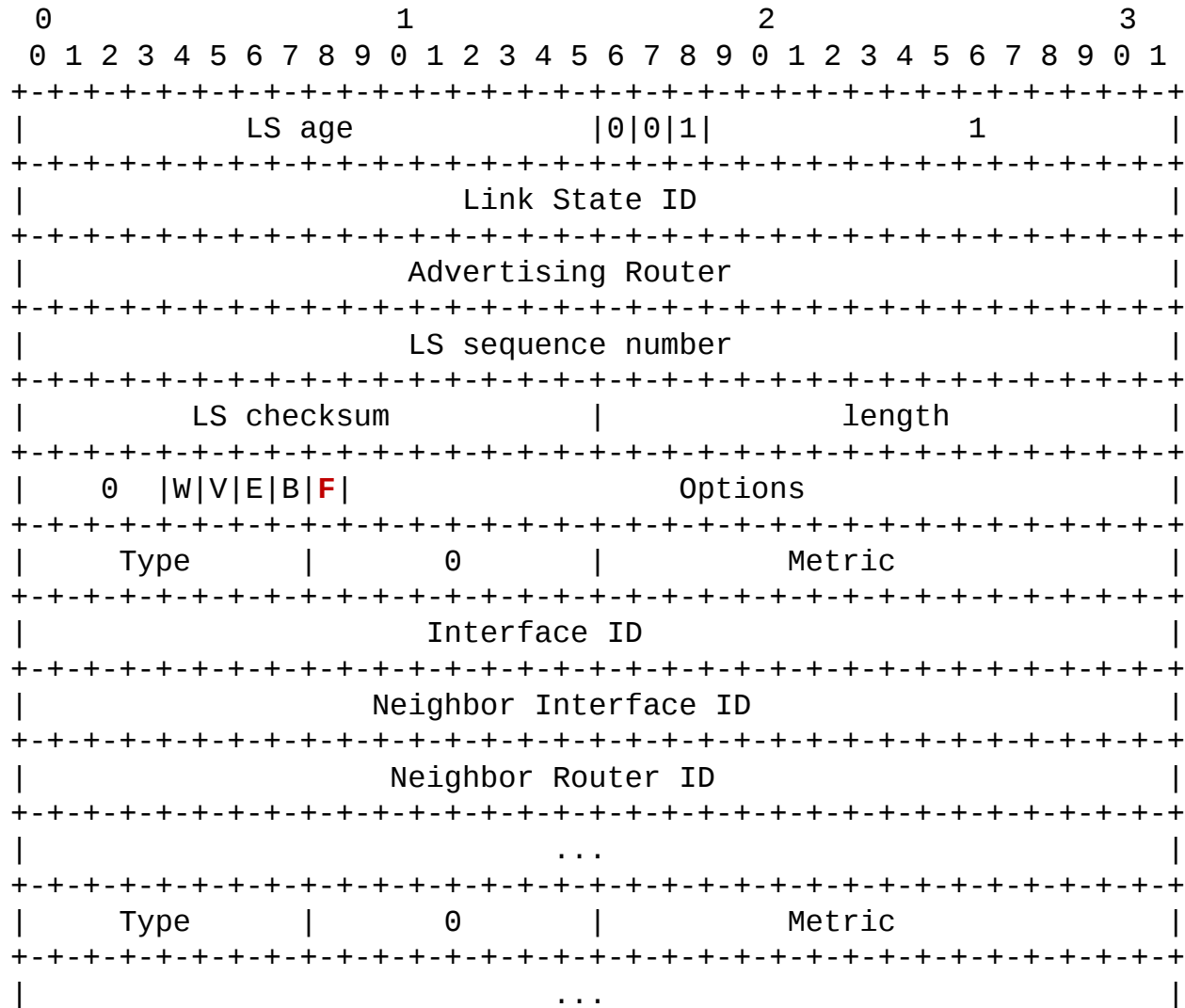
Two types of algorithms are considered.

- One type of algorithms builds a tree **without checking whether a router supports for the link state flooding reduction (LSFR)**. It assumes that all the routers in the domain support for LSFR.
  - The other type of algorithms **considers whether a router supports for LSFR** while building a tree to let the routers supporting for LSFR connected continuously.
    - ◆ **New flag F** is defined in an ISIS router capability TLV and OSPF router LSA
- Flag F:
- set to **1 indicates that the router supports for LSFR**
  - set to 0 indicates that the router does not support for LSFR
- ◆ Select a node R according to a rule such as the node having biggest/smallest node ID and supporting for LSFR (i.e., flag F set to 1).



# Extensions to OSPF v3: New Flag

- 1-bit flag **F** indicating if a router supports for flooding reduction in router LSA

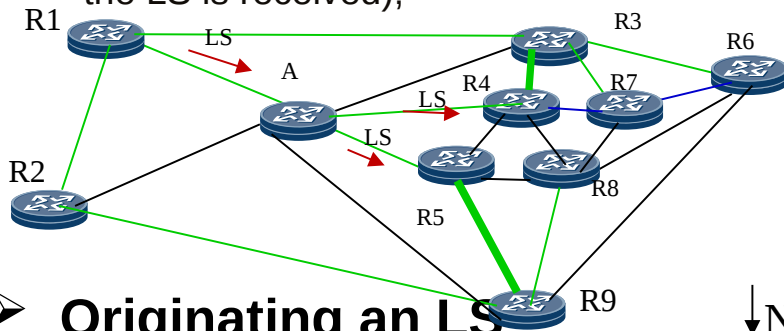




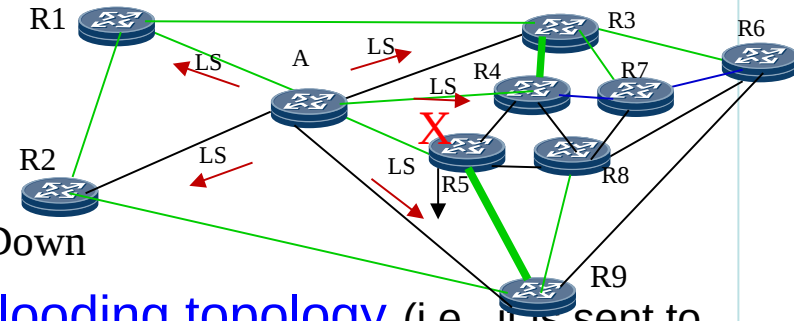
# Revised Flooding Procedure

## ➤ Receiving an LS

LS flooding is according to the flooding topology (i.e., When node A receives a newer LS from a neighbor node over a link, it sends the LS to node A's neighbors over the other links that are attached to node A and on the flooding topology excluding the link from which the LS is received);



Link on flooding topology



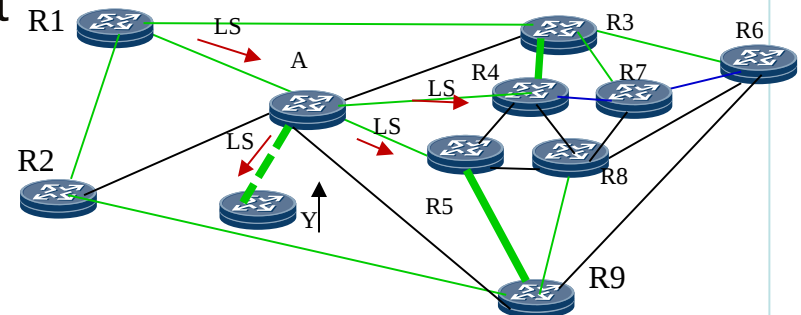
## ➤ Originating an LS

- Refreshing LS is according to the flooding topology (i.e., it is sent to the links on the flooding topology);
- Changed LS is according to the real network topology (i.e., it is sent to all the links).

New node Y connected to node A: Node A assumes  $A=Y$  on flooding topology until flooding topology is re-built

Assume Link on flooding topology

↑ Node goes Up





# Summary

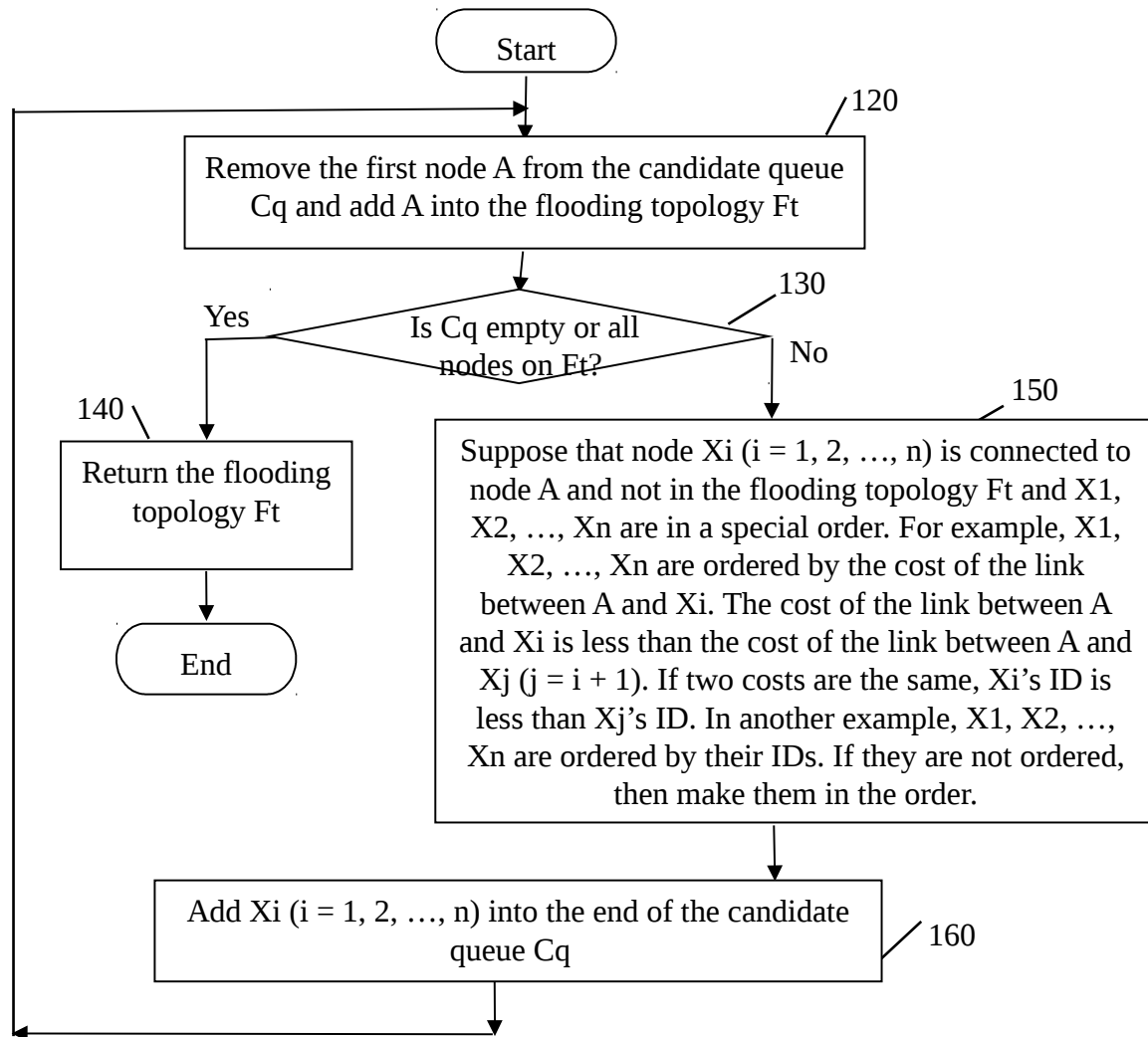
- Each node builds Flooding Topology(FT), No FT flooding in network
- Revised Flooding Procedure Meets Requirements
  1. Backward compatible
  2. Link State (LS) to every node in network in any case
  3. Reduce LS flooding

Welcome comments

# Flooding Topology Generation: Build Tree (1/

m)

An algorithm for building a tree from node R as root starts with a candidate queue Cq containing R and an empty flooding topology Ft:



# Build Flooding Topology: Build Tree (2/m)

An algorithm for building a tree from node R as root with consideration of flag F starts with a candidate queue Cq containing R associated with previous hop PH=0 and an empty flooding topology Ft:

