DALHOUSIE UNIVERSITY

Distributed Online LSP Merging Algorithms

Li Lei and Srinivas Sampalli

Faculty of Computer Science
Dalhousie University
Halifax, NS, Canada

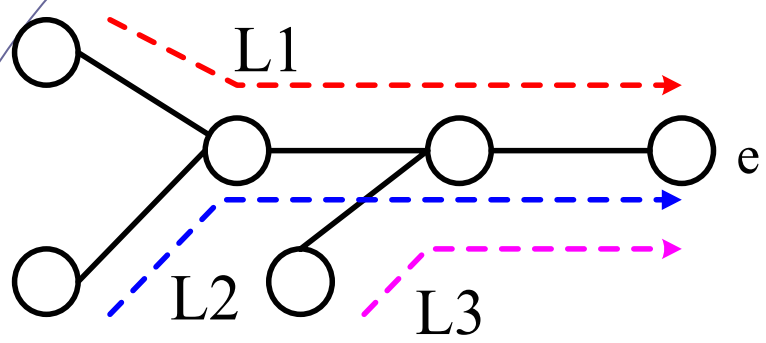
Outline

- Introduction
- Motivation
- On-the-fly Merging Algorithm
- Upstream Wave Merging Algorithm
- Future work

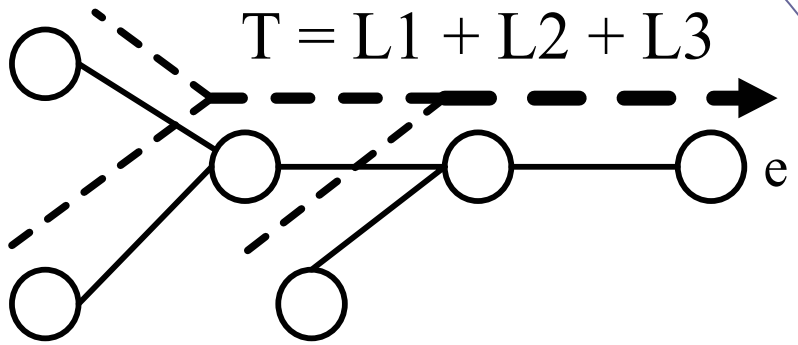
Introduction

- MPLS traffic engineering
  - Constraint-based routing
  - Fast forwarding packet by label swapping
- Next hop label forwarding entry (NHLFE) table
  - mapping incoming label to outgoing label for each LSP
  - Processing delay is determined by the table size
- Solutions to reduce performance degradation on large size networks
  - LSP Merging
  - Hierarchical MPLS

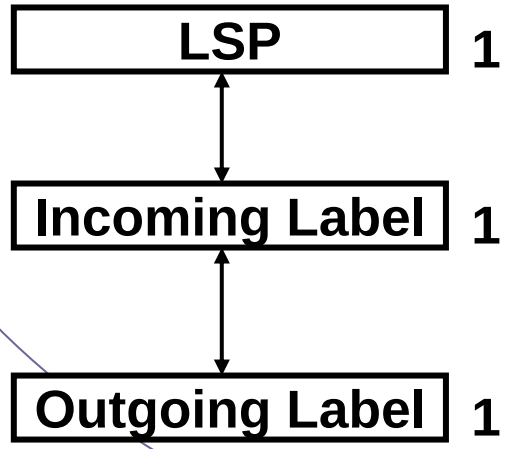
LSP Merging Example

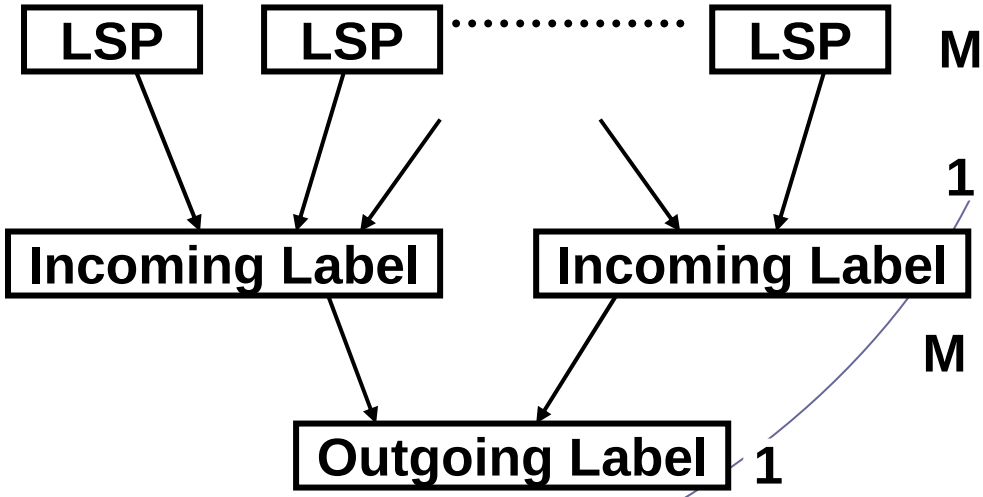


Point-to-Point LSP



Multipoint-to-point LSP Tree





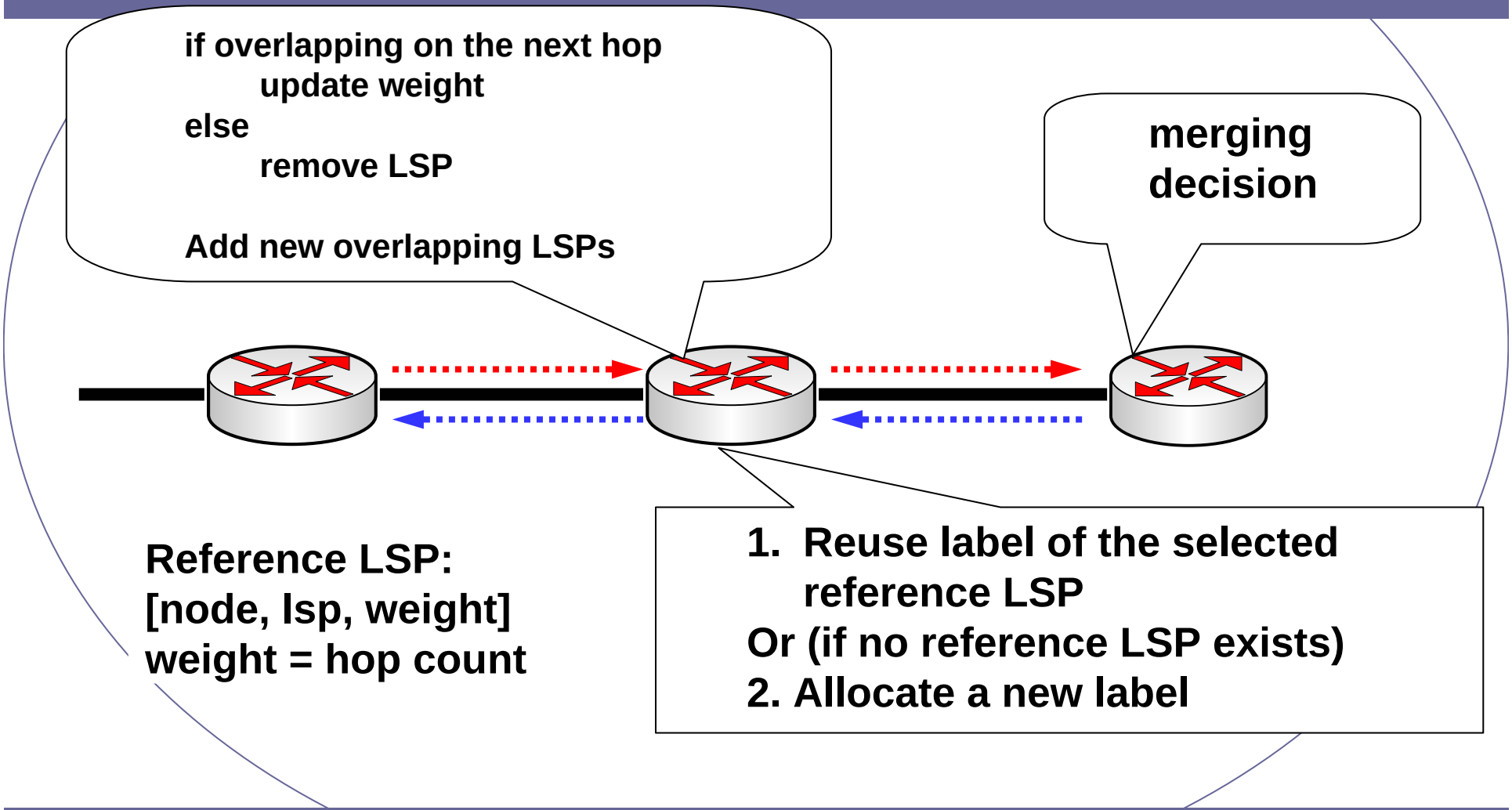
Motivation

- Centralized merging algorithms
  - central control components
  - global routing information
  - lack of scalability
  - performance degrades in on-line use
- Distributed merging algorithm
  - MPLS itself is a distributed control plane
  - Only local information is required
  - Easy deployment
  - On-line use purpose

On-the-fly Merging Algorithm

- Two-way MPLS signaling process
  - Path detection along the route
  - Label distribution in reverse
- Collect merging information
  - Reference LSPs
  - Weight (Hop count)
- Make merging decision at the egress node
- Reuse labels of selected LSP

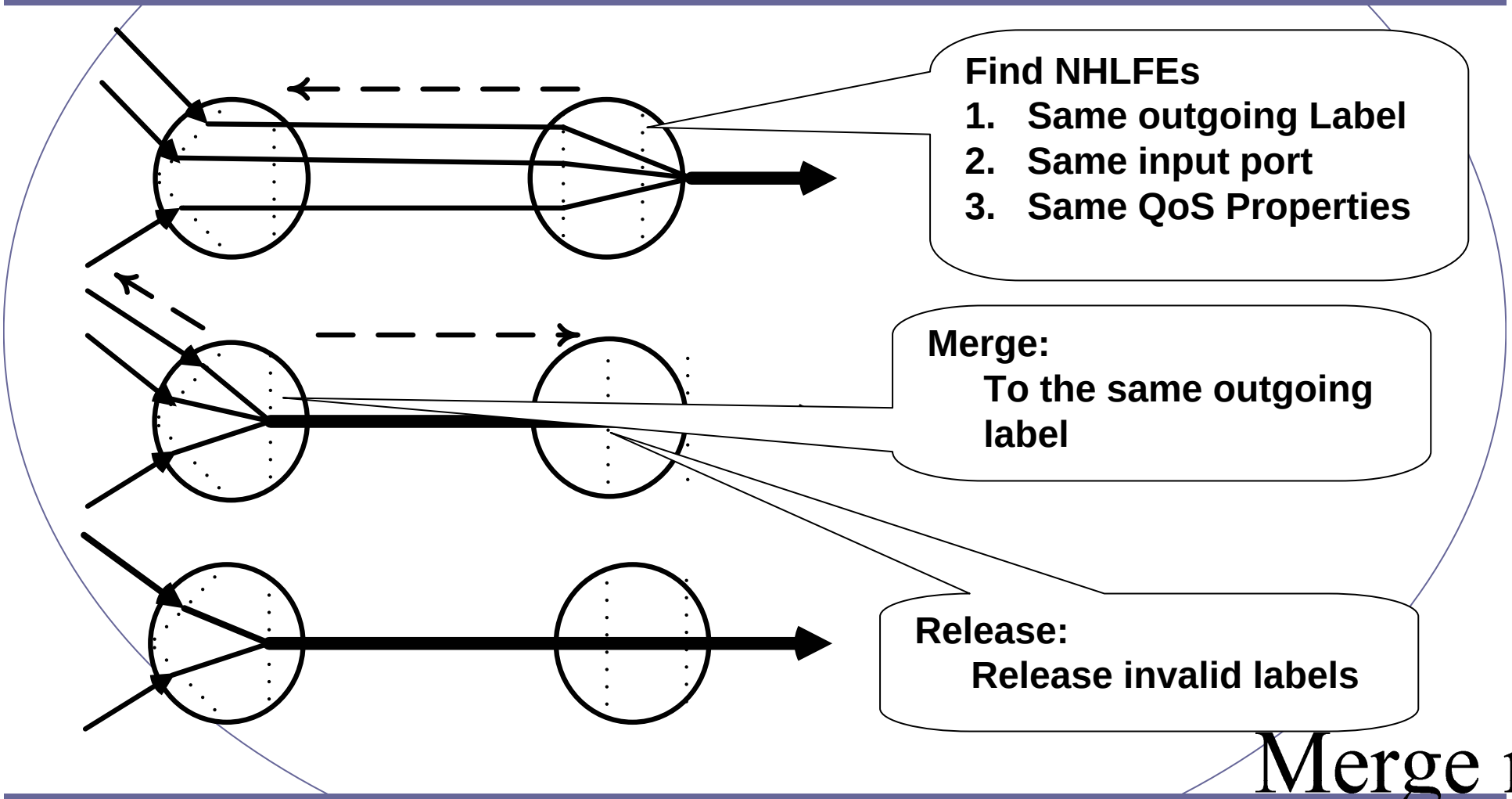
On-the-fly Merging Algorithm



Upstream Wave Merging

- Exhaustive merging algorithm
  - Split LSP trees only use on-the-fly merging
- Starts from the egress node
- Merges LSPs link by link
- Periodically triggered
- Neighbor-Neighbor messages
  - merge message
  - release message

Upstream Wave Merging



Merge mes

Comparison of two algorithms

On-the-fly merging

- Pros
  - Fast merging
  - No new message
- Cons
  - Split LSP trees
  - Insensitive to topology changes

Upstream wave merging

- Pros
  - Exhaustive merging
  - Adaptive to topology changes
- Cons
  - New signaling messages
  - Periodically scheduled

Future Work

- Performance analysis of proposed algorithms
- Integration with MPLS-TE
  - signaling protocols
  - integration with preemption, protection and restoration mechanisms