Ring Protection: Wrapping vs. Steering

Necdet Uzun and Pinar Yilmaz

AuroraNetics, Inc.

Contents

- Objectives
- What are wrapping and steering
- Single/dual fiber cut
- Comparison of wrapping and steering
- Simulation results

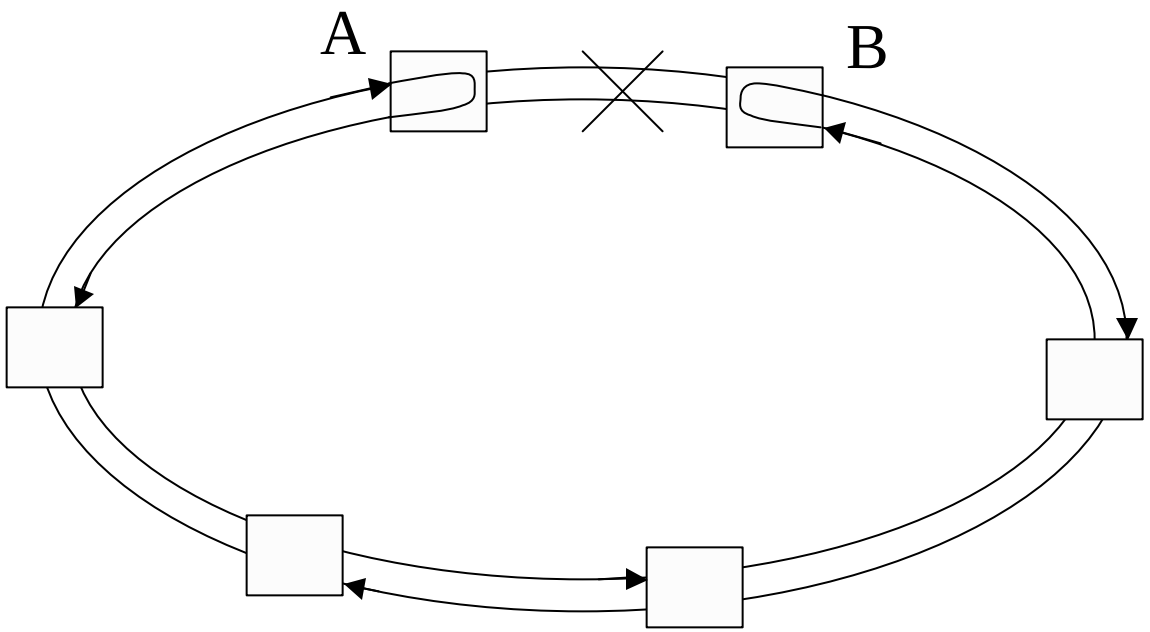
Objectives

- Fast 50 ms protection switching
- Priority discrimination
- High available bandwidth during failure
- No dedicated protection bandwidth

Service outages

- Call dropping (1second)
- Video on demand flickers (< 1 second)
- TCP timeouts (few seconds)
- IGP link removal
- EGP route removal
- Disruption of communication – business revenue loss, etc.
- Legal implications

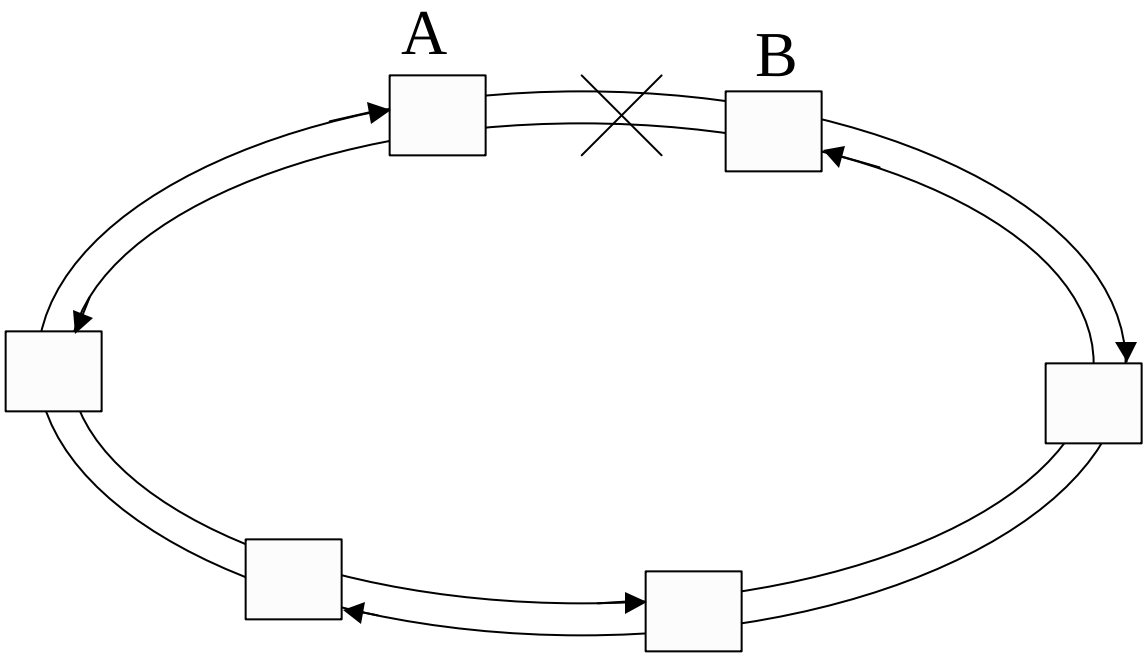
What is wrapping



•Node A (or B) adjacent to a failure point forwards all transit traffic from its Outer (Inner) ring to its Inner (Outer) ring

- •Requires only two nodes (A and B) to detect the failure event
- •Packets may not follow the shortest available route after wrap (until a new topology discovery)

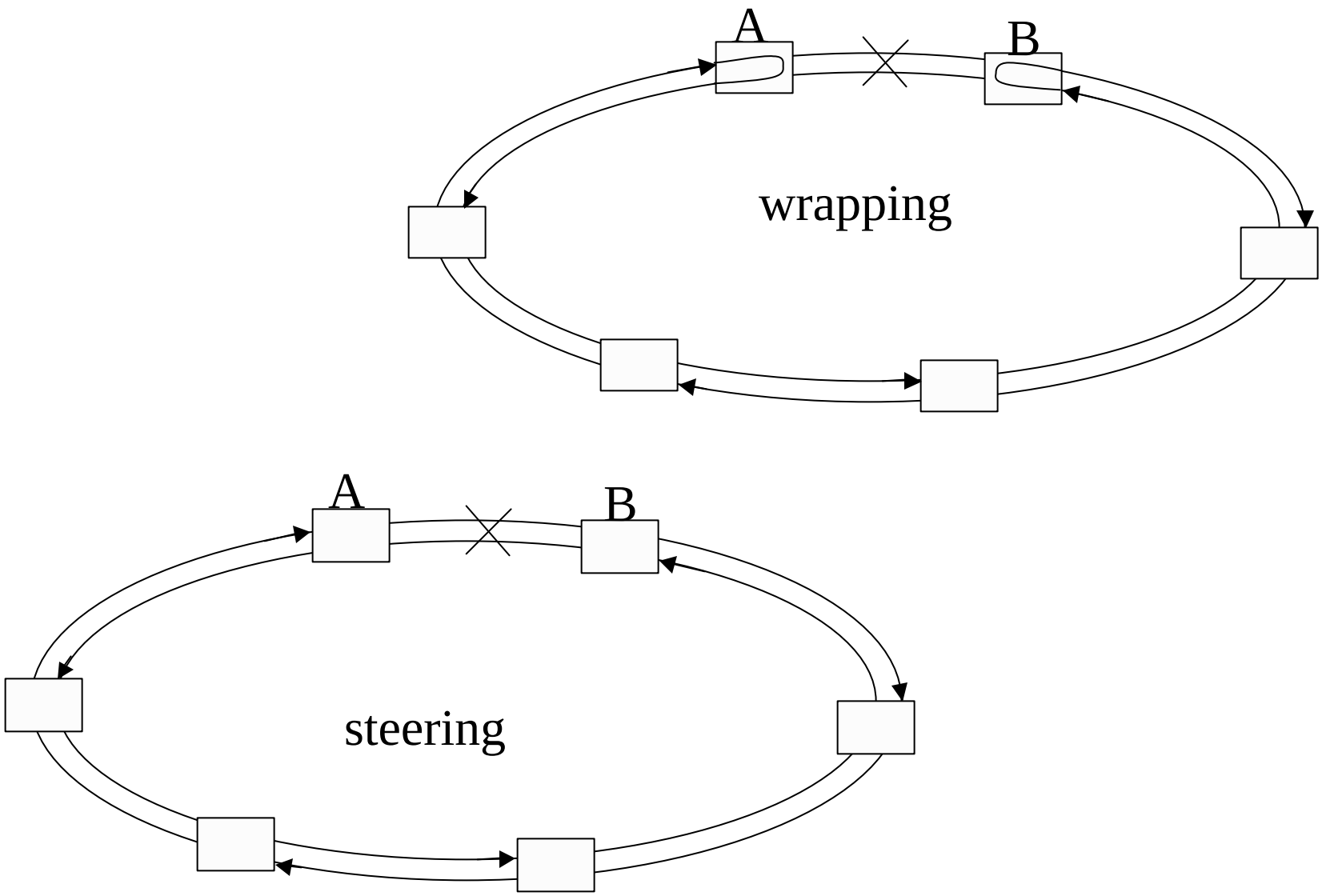
What is steering



•Each node forwards its transmit traffic to either one of Inner or Outer ring so that all packets reach their destinations without the need to pass the failure point

- •Requires all nodes to be informed of the failure event to trigger topology discovery
- •After new topology is discovered, add traffic is switched to the other ring if it becomes the shortest available path
- •Special handling of multicast packets (send to both rings –> problematic when single fiber cut: 2 copies delivered to a single destination)

Dual fiber cut



Dual fiber cut comparison

Wrapping

- ***Fiber cut detection and wrapping***
- ***Sub millisecond*** worth of data is lost

{Fiber cut detection time + Wrap time} LineRate (i.e., [0.4ms +0.1ms] x 10Gb/s = 5Mb)

Steering

- Requires ***Fiber cut detection time + N node delays + node response time*** to switch traffic to opposite ring
- ***{Fiber cut detection time + N node/prop. delays + node steering response time} LineRate + up to (N/2) transit buffer*** worth of data is lost

CASE-1 (fast steering, small ring): [0.4ms + 32 nodes x 0.1ms + 1ms] x 10Gb/s + 16 nodes x 16KB = 48Mb << 20 km ring spans

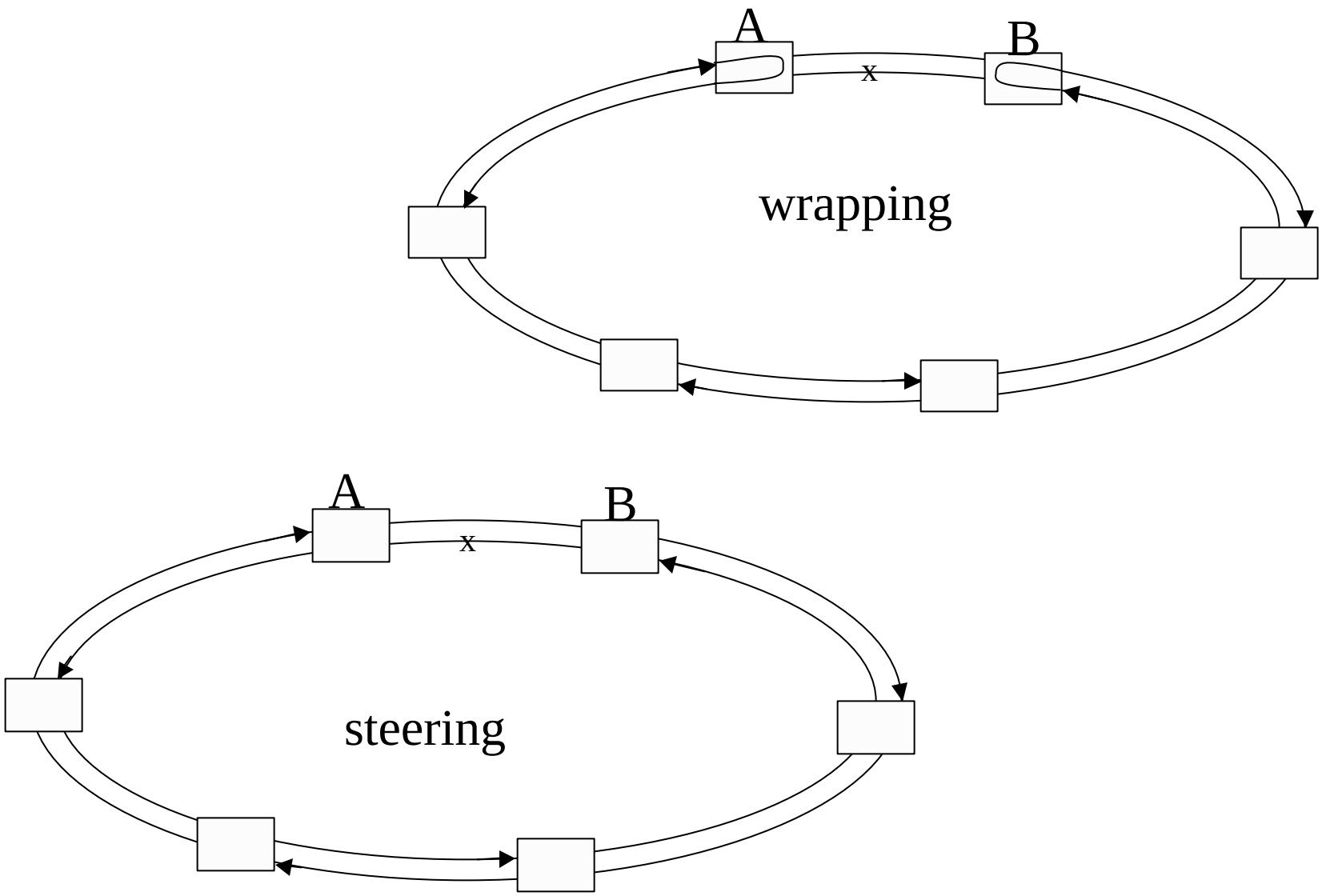
CASE-2 (fast steering, big ring): [0.4ms + 32 nodes x 1ms + 1ms] x 10Gb/s + 16 nodes x 16KB = 370Mb << 200 km ring spans

CASE-3 (slow steering, small ring): [0.4ms + 32 nodes x 0.1ms + 1s] x 10Gb/s + 16 nodes x 16KB = 10Gb ?? << 20 km ring spans

Dual fiber cut

- Recommend: First wrap then steer for fast protection switching
- Desired: No out of order transmission of packets desired during wrapping and steering

Single fiber cut



Single fiber cut comparison

Wrap

•Fiber cut detection and wrapping within ***Fiber cut detection time + Wrap time*** on node A, within ***Fiber cut detection time + Wrap time + 1 node/prop. delay*** on node B

•Sub millisecond worth of data is lost ***{Fiber cut detection time + Wrap time + 1 node/prop. delay} LineRate***

CASE-1: [0.4ms + 0.1ms + 0.1ms]x 10Gb/s = 6Mb << 20 km ring spans

CASE-2: [0.4ms + 0.1ms + 1ms]x 10Gb/s = 15Mb << 200 km ring spans

Steering

•Requires ***Fiber cut detection time + N node/prop. delays + node response time*** to switch transmit traffic to opposite ring

•***{Fiber cut detection time + N node/prop. delays + node steering response time} LineRate + up to (N/2) transit buffer*** worth of data is lost

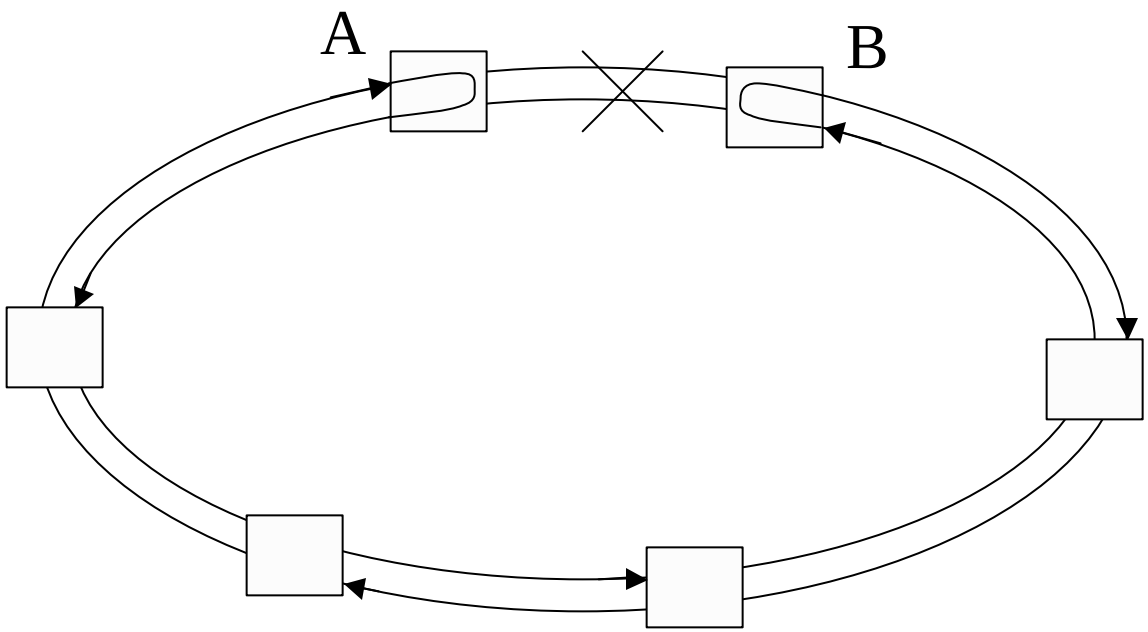
CASE-1 (fast steering, small ring): [0.4ms + 32 nodes x 0.1ms + 1ms] x 10Gb/s + 16 nodes x 16KB = 48Mb << 20 km ring spans

CASE-2 (fast steering, big ring): [0.4ms + 32 nodes x 1ms + 1ms] x 10Gb/s + 16 nodes x 16KB = 370Mb << 200 km ring spans

CASE-3 (slow steering, small ring): [0.4ms + 32 nodes x 0.1ms + 1s] x 10Gb/s + 16 nodes x 16KB = 10Gb ?? << 20 km ring spans

First wrap then steer!

Wrapping then steering



- Neighboring node A (and B) of a failure point forwards all transit traffic from its Outer (Inner) ring to its Inner (Outer) ring
  - A and/or B detect the failure event and wrap ASAP then inform other nodes to start a new topology discovery
  - After new topology is discovered, add traffic is switched to the other ring if it becomes the shortest available path