



Unleashing Segment Routing TE with SR Flexible Algorithms

Jose Liste, Technical Marketing Engineer (jliste@cisco.com)

iNOG12 - February 2019

Why are we here?

- Have you ever wished you could **tailor IGP computation** to fit your own traffic engineering needs? Such as by computing:
 - Paths considering a subset of the routers in your network
 - Paths that minimize cumulative delay to a destination based on measured per-link delay
 - Paths traversing only encrypted MACsec links
 - Paths traversing only high speed interfaces

Why are we here?

- All these use cases are now possible !!!
- And furthermore, they are possible at scale and without adding any extra state into the network
- Meet SR IGP Flexible Algorithms !!!
 - A new member of SRTE family



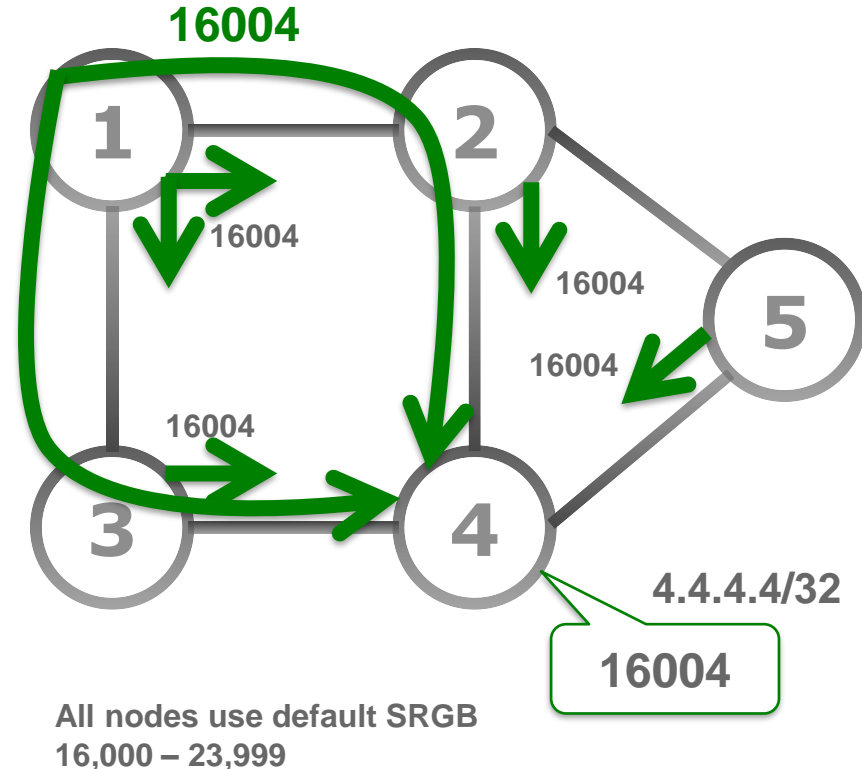
Agenda

- Overview
- Operation
- Use Cases

SR Flexible Algorithm Overview

Recap - IGP Prefix Segment

- IGP Prefix-SID
 - Advertised as label value
 - Operator-allocated value from SRGB
 - Advertised as index
- Distributed by ISIS/OSPF
- Shortest-path to the IGP prefix
 - Equal-Cost Multi-Path (ECMP)-aware
- Global Segment
 - Programmed in every node



SR IGP Flexible Algorithm

- Complements the SRTE solution with customizable Prefix-SIDs
- Leverages the SRTE benefits of simplicity and automation
 - Automated sub-50msec FRR (TILFA)
 - On-Demand Policy (ODN)
 - Automated Steering (AS)

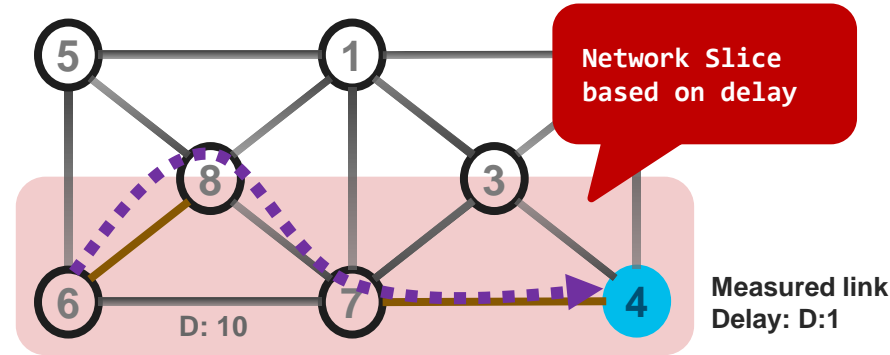
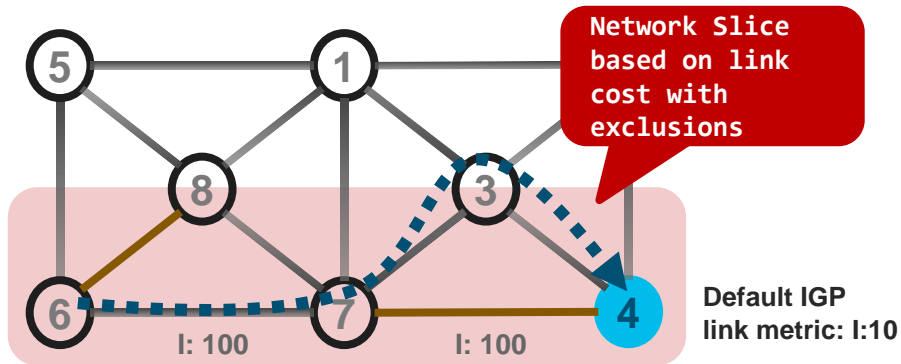
SR IGP Flexible Algorithm

- Complements the SRTE solution with customizable Prefix-SIDs
- We call “Flex-Algo”
 - The algorithm is defined by the operator, on a per-deployment basis
- Flex-Algo K is defined as
 - The minimization of a specified metric: IGP, delay, ...
 - The exclusion of certain link properties: link-affinity, SRLG, ...

SR IGP Flexible Algorithms

• Examples

- Operator defines Flex-Algo 128 as “minimize IGP metric while avoiding links with link-affinity “brown”
- Operator defines Flex-Algo 129 as “minimize delay metric”



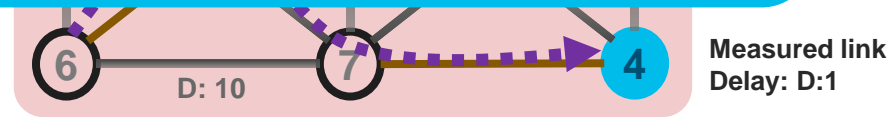
SR IGP Flexible Algorithms

• Examples

- Operator defines Flex-Algo 128 as “minimize IGP metric while avoiding links with link-affinity “brown”
- Operator defines Flex-Algo 129 as “minimize delay metric”

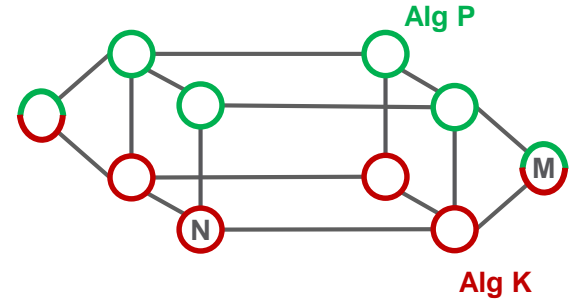
And the question is ... How many labels are needed to enforce traffic on the Flex-Algo path?

A single SID == SRTE with a single SID !!!



SR Flexible Algorithm Operation

Flex-Algo Operation



1

• Flex-Algo Membership

- Each node **MUST** advertise Flex-Algo(s) that it is participating in
- A Flex-Algo instance can be enabled on all or a subset of nodes
- Each node can participate in multiple Flex-Algo(s)

• Example:

- Node N is enabled to participate in Flex-Algo instance K
- Node M is enabled to participate in Flex-Algo instances K and P

Flex-Algo Operation

2

Flex-Algo Prefix SIDs

- If a node advertises participation in a Flex-Algo likely it also advertises a prefix SID for that Flex-Algo

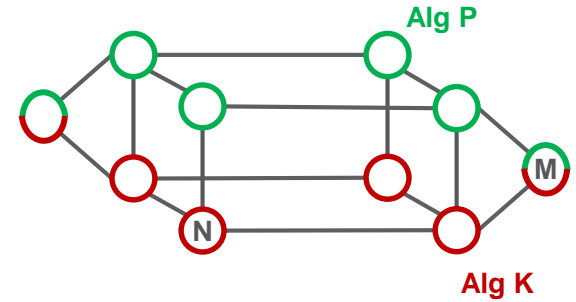
• Example:

• Node N advertises

- Prefix SID 1600N for ALGO 0
- Prefix SID 1700N for ALGO K

• Node M advertises

- Prefix SID 1600M for ALGO 0
- Prefix SID 1700M for ALGO K
- Prefix SID 1800M for ALGO P



Flex-Algo Participation and Prefix-SID

- Each Prefix SID is related to an algorithm

0									1									2									3								
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1				
Type									Length									Flags									Algorithm								
SID/Index/Label (variable)																																			

- Each node advertises its ALGO capability

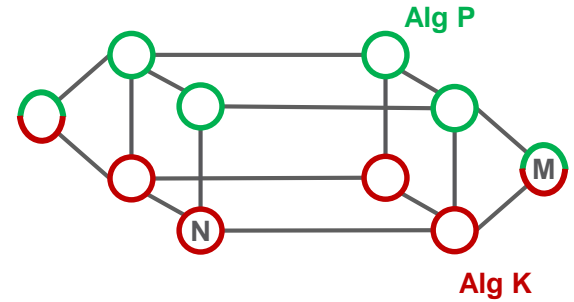
The SR-Algorithm sub-TLV has following format:

0									1									2									3								
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1				
Type									Length																										
Algorithm 1									Algorithm 2									Algorithm ...									Algorithm n								

Flex- Algo Operation

2 • Flex- Algo Prefix SIDs (cont.)

- No additional loopback address
- Flex- Algo Prefix SID's can be advertised as additional prefix- SID's of the existing loopback address
- Example:
- Node M advertises `loopback0 1.1.1.M/32` with
 - Prefix SID 1600M for ALGO 0
 - Prefix SID 1700M for ALGO K
 - Prefix SID 1800M for ALGO P

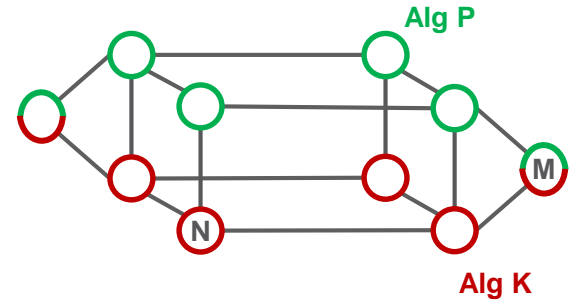


Flex-Algo Operation

3

• Flex-Algo Definition

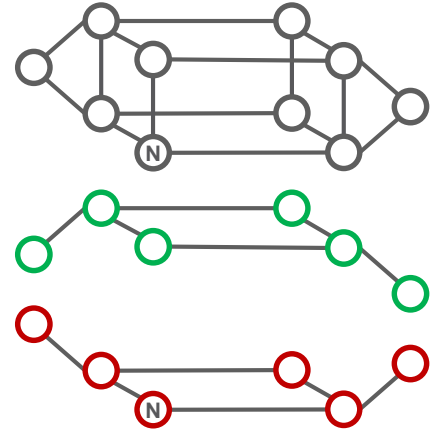
- Each node **MUST** have a consistent definition of the Flex-Algo(s) that it is participating in
- Local configuration
 - likely automated during day-0 provisioning
- Learned from a central entity via IGP flooding
 - new top TLV defined for Flex-Algo definition advertisement
- Example:
 - Flex-Algo instance K == minimize delay metric



Flex-Algo Operation (cont.)

- 4 • Flex-Algo Computation
 - N prunes any node not a member of K
 - N prunes any link that is excluded by K
 - Resulting topology is called $\text{Topo}(K)$
 - N compute shortest-path tree on $\text{Topo}(K)$ with metric defined by K

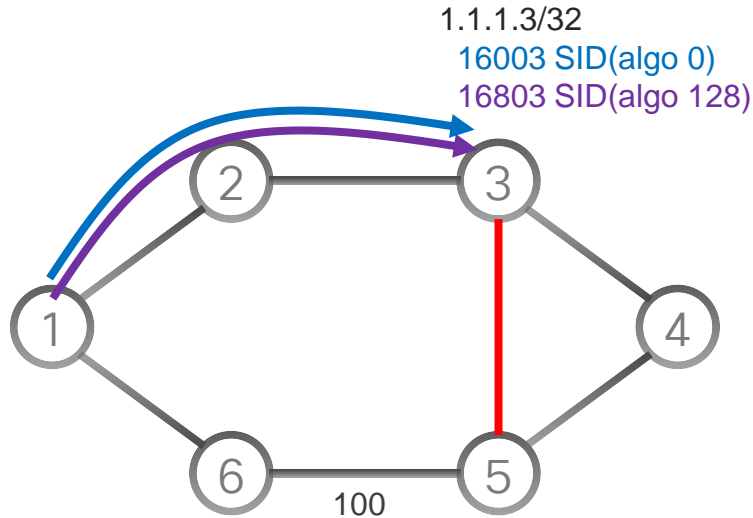
- 5 • Flex-Algo Prefix SID FIB installation
 - N installs any reachable Prefix-SID of K in the forwarding table along the computed shortest-path on $\text{Topo}(K)$



Topology Independent LFA (TI-LFA)

- TI-LFA algorithm is performed within Topo(K)
- Backup path is expressed with Prefix-SIDs of Algo K
- Benefits: the backup path is optimized per Flex-Algo !!!

Example – Primary paths per Algo



Each node in this topology supports SR alg0, alg128
Default IGP link metric: I:10

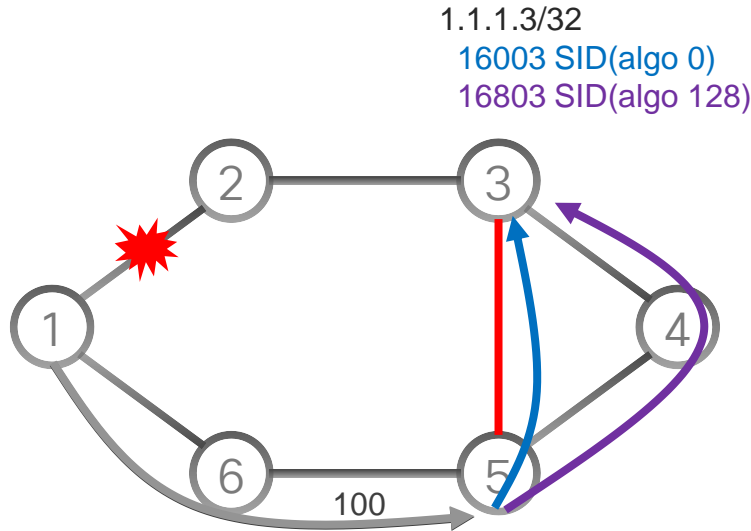
At node 1 for destination 3

16003 => 16003 via 2

16803 => 16803 via 2

All nodes participate in Flex- Algo 128
which is defined as min IGP metric
and avoid **red** affinity

Example – TI-LFA Backup path per Algo



At node 1 for destination 3

16003 => 16003 via 2

→ backup: <24065, 16003> via 6

16803 => 16803 via 2

→ backup: <24065, 16803> via 6

The usage of Algo-128 Prefix-SID 16803 ensures that the Algo 128 backup path also avoids the red link

Reminder: 240XY is the Adj SID from node X to node Y

OSPF and SRv6

- Same applies to OSPF
- Same applies to SRv6

IETF

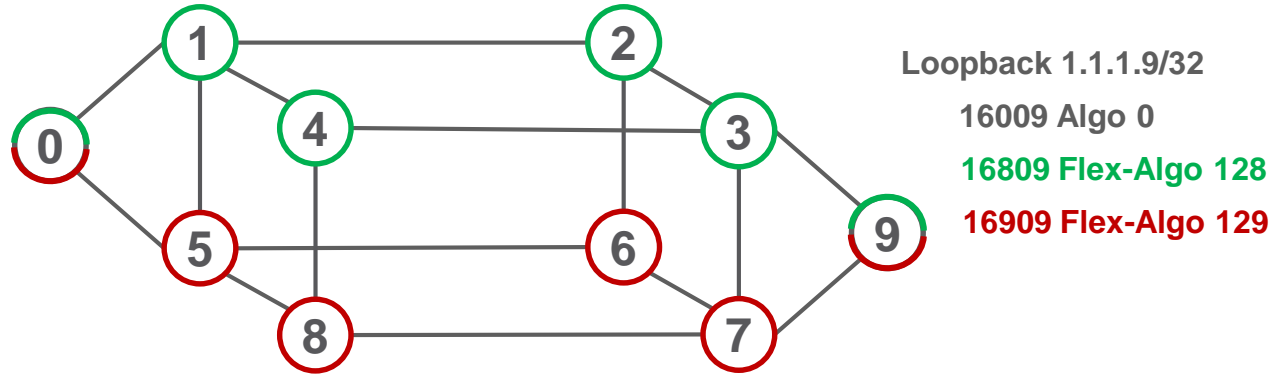
- [draft-ietf-spring-segment-routing](#)
 - Prefix-SID per Algorithm
- [draft-ietf-spring-segment-routing-policy](#)
 - SR Policy architecture, ODN, AS
- [draft-ietf-lsr-flex-algo](#)
 - Customization of Algo and consistency
- [draft-ketant-idr-bgp-ls-flex-algo](#)
 - Flex-Algo definition advertisement with BGP Link-State
- [draft-ietf-isis-te-app](#)
 - Used to flood Flex-Algo specific link affinities
- [RFC7810](#) (IS-IS Traffic Engineering (TE) Metric Extensions)
 - Used to advertise extended TE metrics – e.g. link delay

IETF

- [draft-ietf-spring-segment-routing](#)
 - Prefix-SID per Algorithm
- [draft-ietf-spring-segment-routing-policy](#)
 - SR Policy architecture, ODN, AS
- [draft-ietf-lsr-flex-algo](#)
 - Customization of Algo and consistency
- [draft-ketant-idr-bgp-ls-flex-algo](#)
 - Flex-Algo definition advertisement with BGP Link-State
- [draft-ietf-isis-te-app](#)
 - Used to flood Flex-Algo specific link affinities
- [RFC7810](#) (IS-IS Traffic Engineering (TE) Metric Extensions)
 - Used to advertise extended TE metrics – e.g. link delay

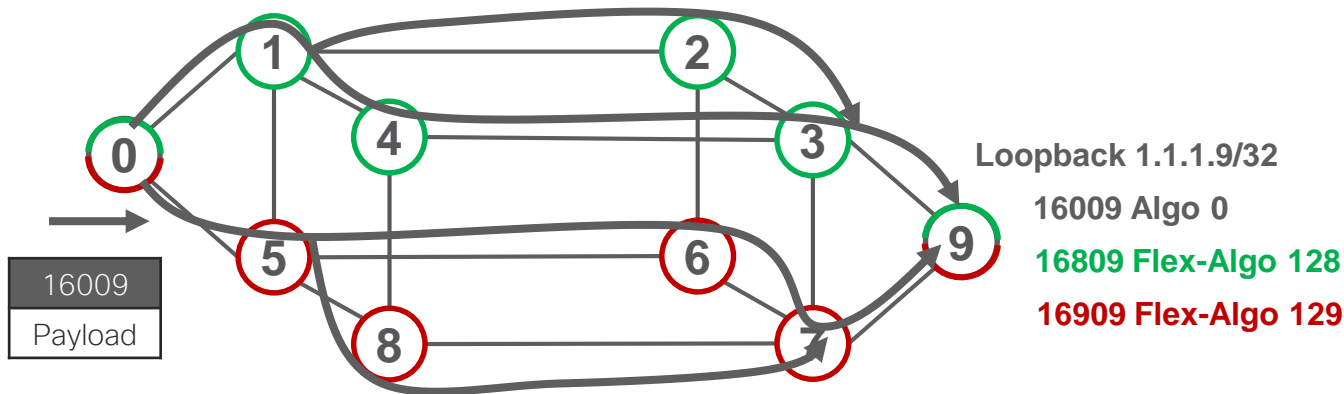
SR Flexible Algorithm Use-Cases

Use-Case - Multi-Plane Networks



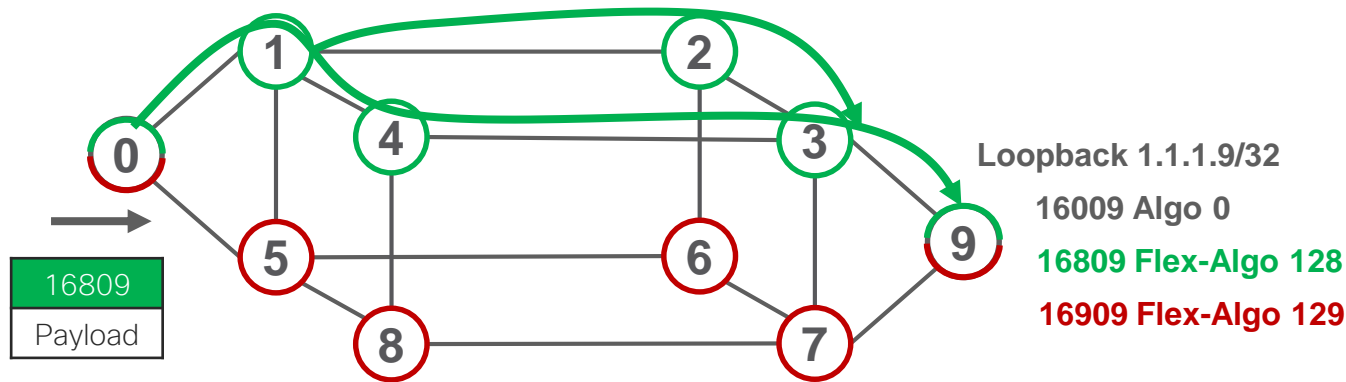
- All the nodes support Algo 0: minimize IGP metric
- Green nodes also support 128: minimize IGP metric
- Red nodes also support 129: minimize Delay

Use-Case - Multi-Plane Networks (cont.)



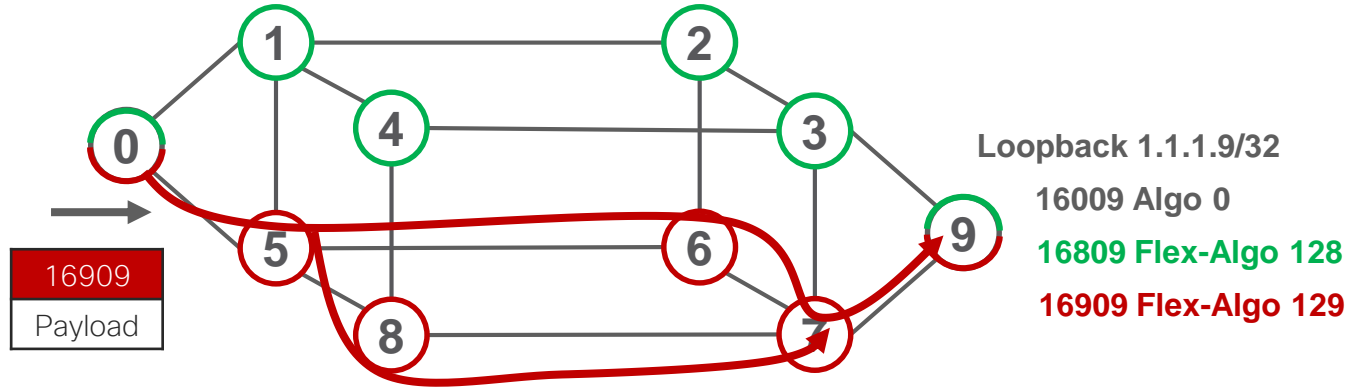
- Path to Node 9 across Algo 0

Use-Case - Multi-Plane Networks (cont.)



- Path to Node 9 across **Flex-Algo 128**

Use-Case - Multi-Plane Networks (cont.)



- Path to Node 9 across **Flex-Algo 129**

Automated Steering

- SRTE Automated Steering (AS) is leveraged for IGP Flex-Algo

```
segment-routing
  traffic-eng
    on-demand color 100
    dynamic mpls
      sid-algorithm 128
```

“Any 100-colored BGP route should be steered via the prefix-SID(ALGO 128) of the BGP nhop”

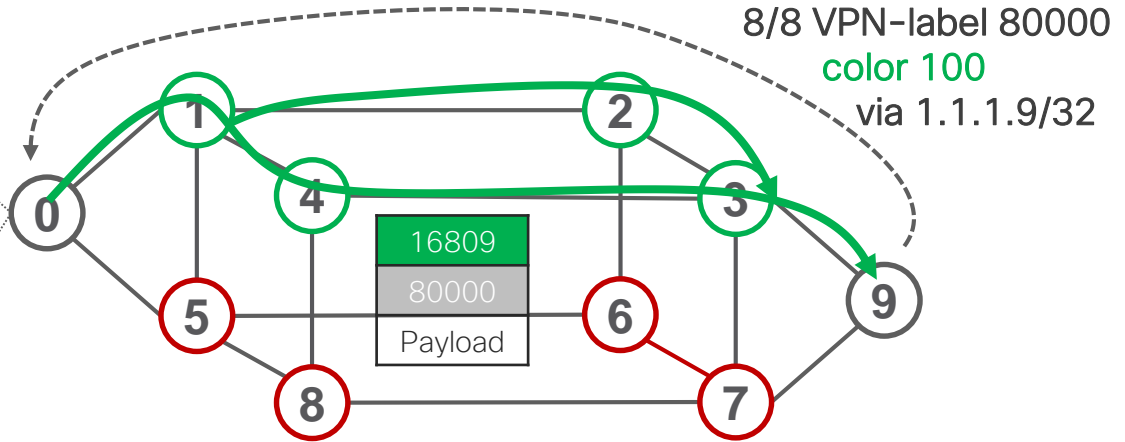
Automated Steering – Multi-Plane

```
segment-routing
traffic-eng
  on-demand color 100
  dynamic mpls
  sid-algorithm 128

router isis 1
  flex-algo 128
```

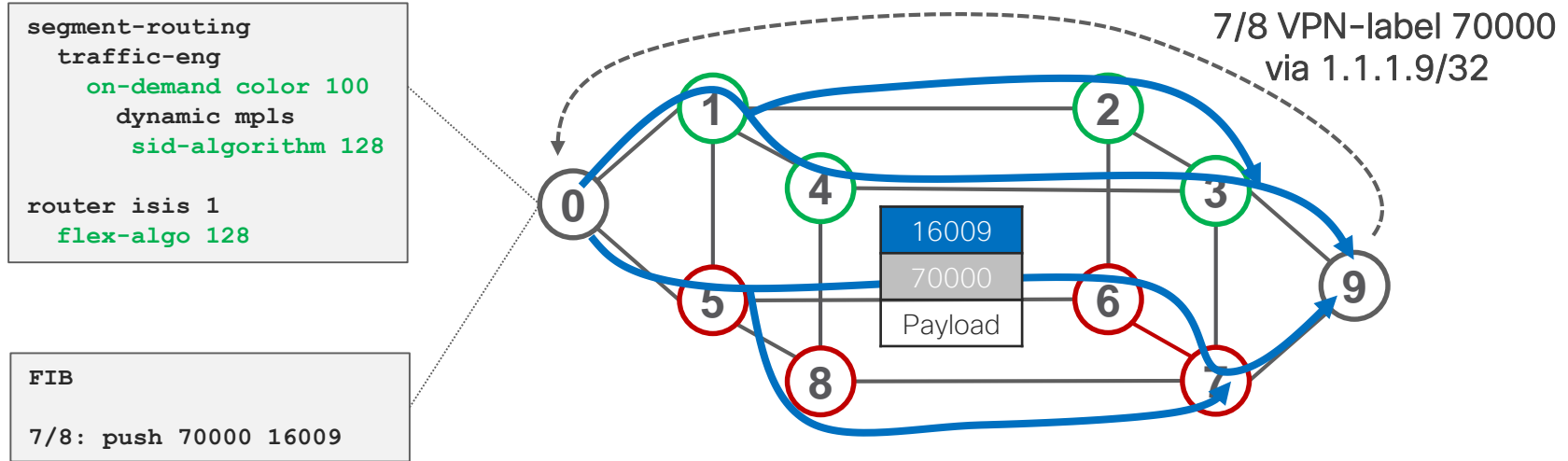
```
FIB

8/8: push 80000 16809
```



- Node 0 automatically steers any BGP route with color 100 from 9 via 16809 hence via the green plane only
- One single Flex- Algo Prefix-SID expresses the end-to-end SLA path

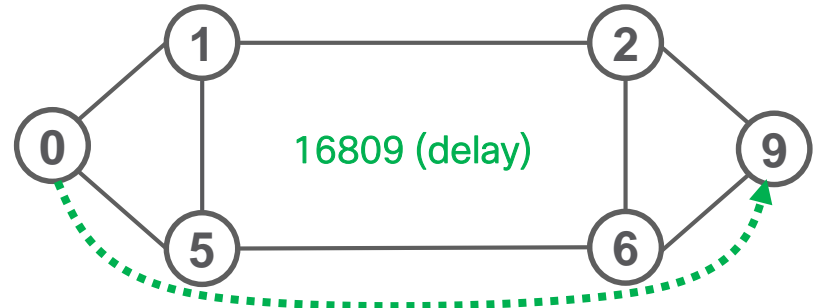
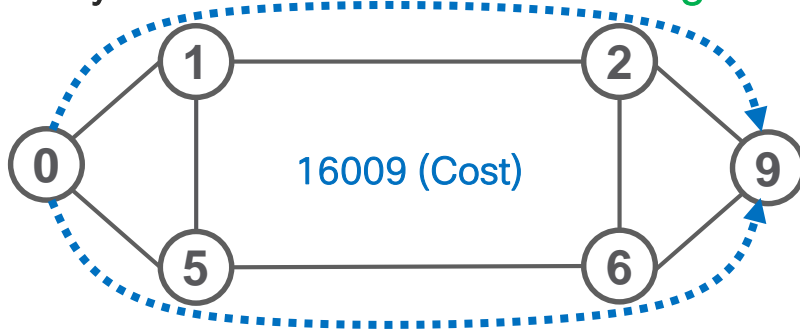
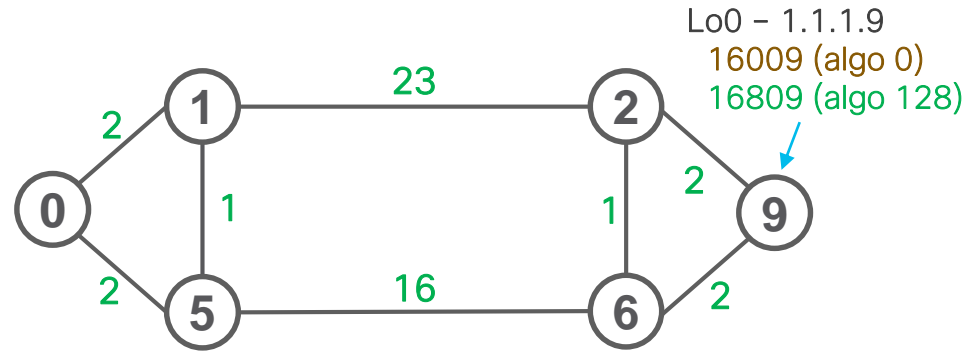
Automated Steering – Multi-Plane



- Node 0 automatically steers any BGP route **without color** from 9 via 16009 (any plane)

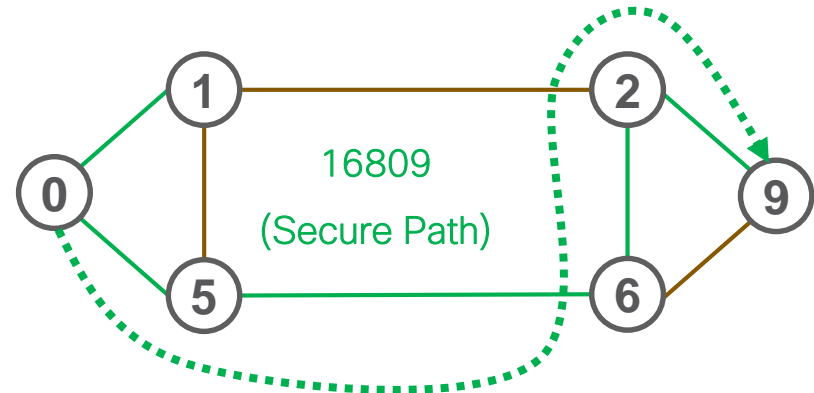
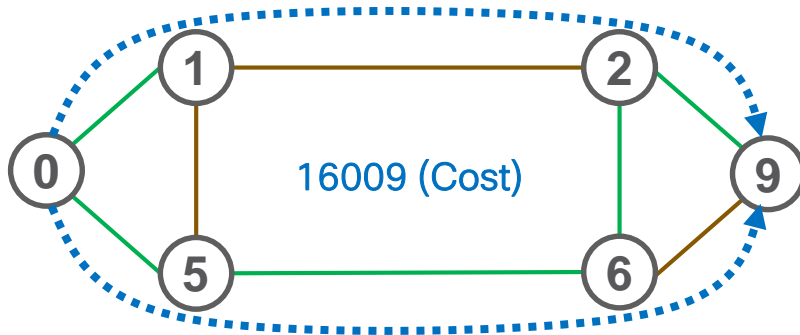
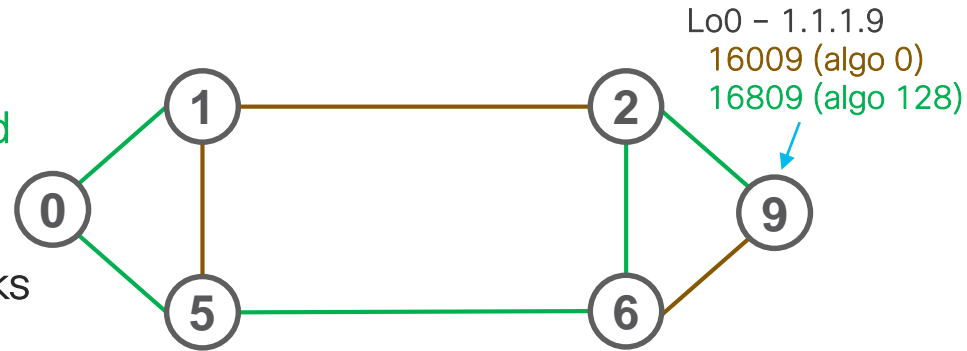
Use-Case – Delay vs Cost of Transport

- All nodes support Algo 0 & 128
- ISIS link metric 10
- Algo 128: minimize delay metric
- Per-link measurement of delay and advertisement as delay metric via ISIS
- Delay metric at that time shown in green



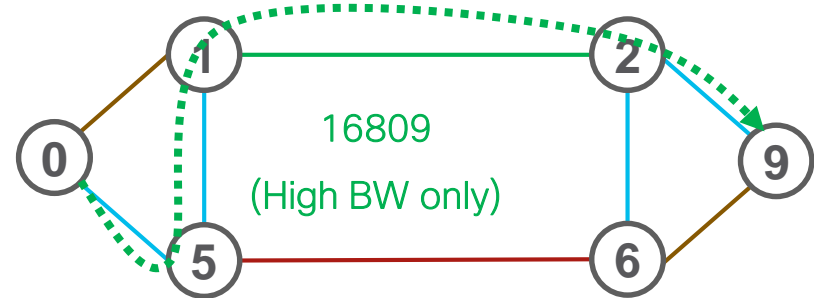
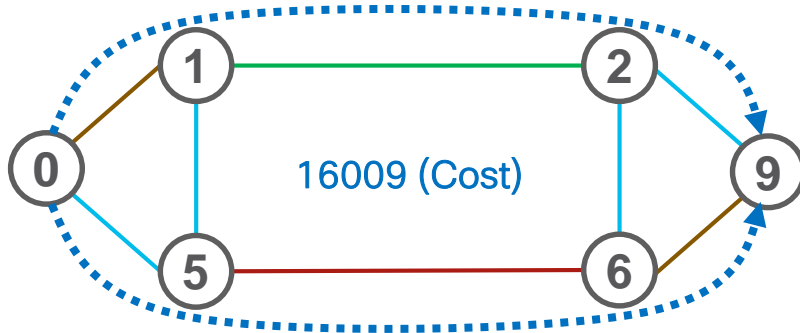
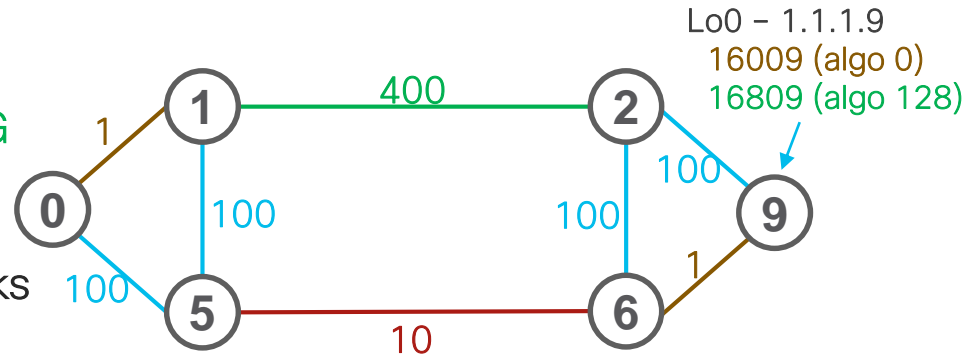
Use-Case – SRTE for Intelligent Secure Paths

- ISIS link metric 10
- Link colors shown **Unencrypted** / **Encrypted**
- All nodes support Algo 0 & 128
- Algo 128: minimize IGP while traversing links with encryption enabled (**exclude brown**)
- Per-link colors flooded in IGP



Use-Case - SRTE for High-BW Links Preference

- ISIS link metric 10
- Link colors shown 1G / 10G / 100G / 400G
- All nodes support Algo 0 & 128
- Algo 128: minimize IGP while traversing links of 100G or more (exclude brown and red)
- Per-link colors flooded in IGP



Demonstration

If we had more time ...

- Inter-Domain path computation with Flex-Algo

Conclusions

SR IGP Flexible Algorithm

- **Complements the SRTE solution by adding new Prefix-Segments with specific optimization objective and constraints**
 - minimize igp-metric or delay or te-metric
 - avoid link-affinity or SRLG
- **TE path** from anywhere to anywhere automatically **computed by IGP**
- **Single SID is used to enforce traffic** on the Flex-Algo specific path
- Leverages the SRTE benefits of **simplicity and automation**
 - Automated **sub-50msec FRR** (LFA / TI-LFA)
 - **On-Demand Policy** (ODN)
 - **Automated Steering** (AS)

Resources / Stay Up-To-Date



<http://www.segment-routing.net/>



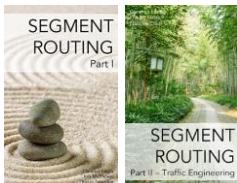
<https://www.linkedin.com/groups/8266623>



<https://twitter.com/SegmentRouting>



<https://www.facebook.com/SegmentRouting/>



[Segment Routing, Part I / II - Textbooks](#)

Thank You