Multipath Traffic Engineering for Segment Routing

draft-stone-spring-mpte-sr

IETF 123 - Madrid

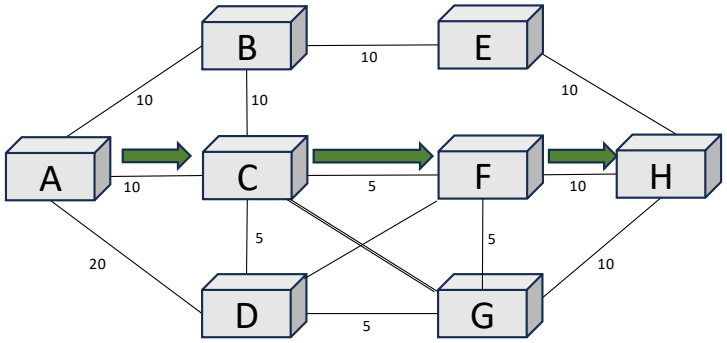
Andrew Stone (andrew.stone@nokia.com) – Presenter

Vishnu Pavan Beeram – (vbeeram@juniper.net)

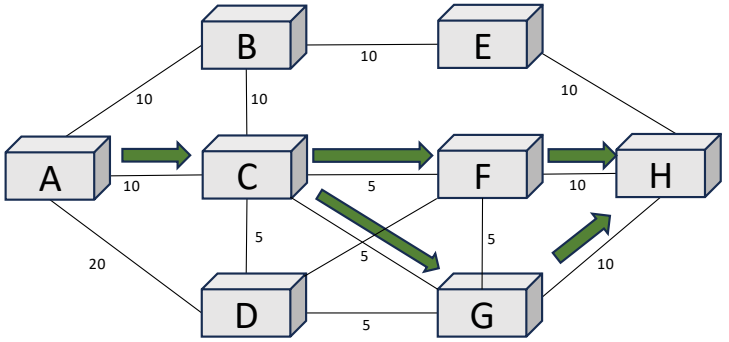
Multipath Traffic Engineering

- Introduced in draft-kompella-teas-mpte
- Different transport tunnel types concept comprised of:
  - Traffic engineered constraints
  - ECMP paths
  - Non-ECMP Paths – slack x of the shortest path
  - Multi-egress nodes and multi-ingress (maybe)
  - Unequal weight load balancing per hop
- Transport tunnel is represented as a Directed Acyclic Graph (DAG) instead of a Path
- Computed by a headend ingress or a PCE
- Can be centralized or distributed solution for signaling

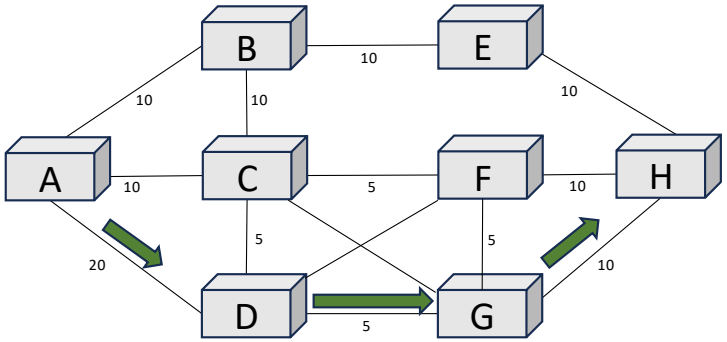
Multipath Traffic Engineering



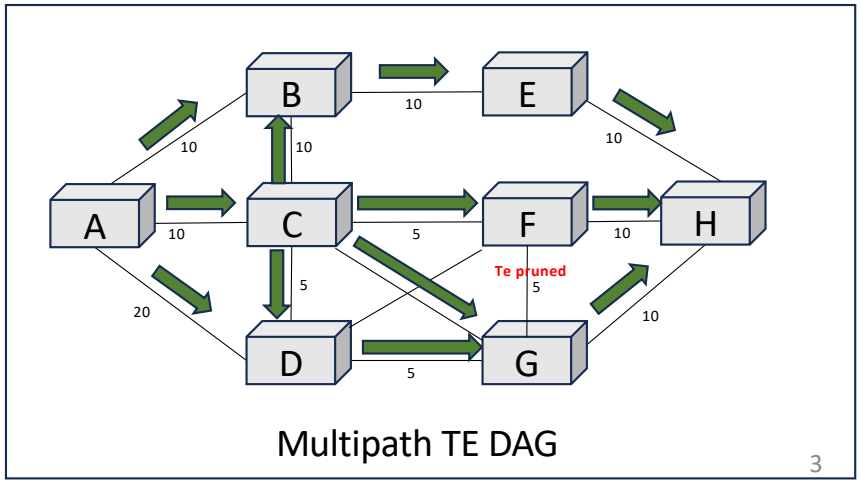
Shortest path



Shortest ECMP path



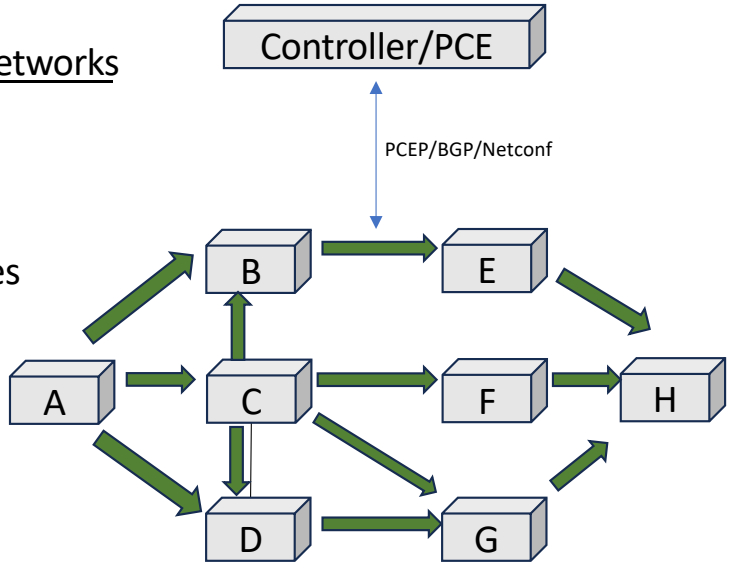
TE Path



Multipath TE DAG

draft-stone-spring-mpte-sr

- Describes a centralized solution to compute and signal MPTE DAG for SR networks
- Controller/PCE to compute, PCEP/BGP/Netconf to signal
- The DAG is comprised of Junction Segments deployed on Junction Nodes
  - Junction node is a transit node that forwards to one or more interfaces
  - Similarity to Replication Segment - but load balanced, not replicated
- Junction Segment is comprised of:
  - Incoming SID (Binding SID)
  - Outgoing SID list(s)
    - Next-Junction Segment(BSID) and SR-Path stack
    - Weight
- A Junction segment is *realized* with SR Policy with a single candidate path



….but just use SR Policy w/multiple segment lists

Yes !

But… depending on the topological graph and size of the DAG, this could be a significant number of paths.

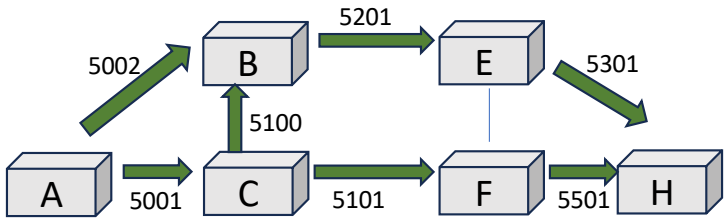
+ Deviating shortest path significantly can impact MSD of the SID list

+ Ingress only load balancing vs per junction node load balancing.

Decision to use many SID lists on headend or MPTE solution would be scenario dependent

Junction Segment

Partial, simple example with SR-MPLS



Headend-A

SID List:

[5002, 100], 0.5

[5001, 101], 0.5

Junction Segment B

Binding SID = 100

SID List:

[NodeSID-H]

Takes SR shortest path to H, skips the need for junction on E

Junction Segment C

Binding SID = 101

SID List:

[5100, 100], 0.4

[5101, 102], 0.6

Junction Segment F

Binding SID = 102

SID List:

[5501], 1.0

Could be optimized out like E, used as an example

- Node SIDs are value NodeSID-{letter}
- Link values are adjacency SIDs
- SID List Pos-0 is top of stack

Junction Segment

- Realized by SR Policy with a single candidate path
- An SR Policy is unique by {Headend, Endpoint, Color}
- Multiple endpoints -> use 0.0.0.0
- draft-kompella-teas-mpte describes instance of the DAG is the Multipath TED(MPTED), which is identified by the MID and has a version.
  - Useful to know DAG membership of the Junction Segment (SR Policy)
  - -> encode MID+Version in the color attribute (**)
    - Need to make sure color range is reserved block

Junction Segment C – SR Policy

Headend: C

Endpoint: 0.0.0.0

Color: <MID+VERSION>

Binding SID = 101

SID List:

[5100, 101], 0.4

[5101, 102], 0.6

** An idea...perhaps new candidate path property

Hierarchy

- The ingress SR Policy of the DAG can also have a binding SID
- The DAG can be re-used by multiple upstream SR Paths (could be a DAG itself too)

Headend-X

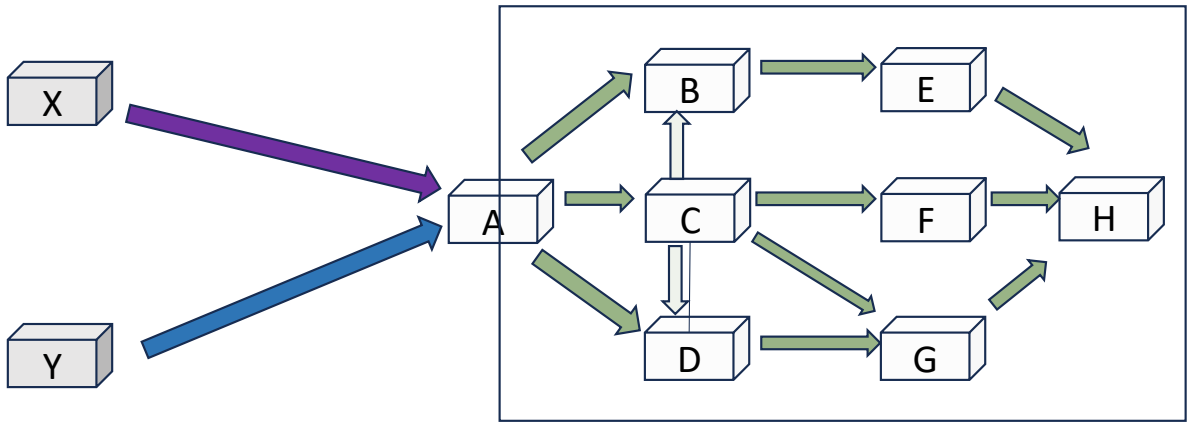
SID-List:

[NodeSID-A, 200]

Headend-Y

SID-List:

[NodeSID-A, 200]



Headend-A

BSID-200

draft-stone-spring-mpte-sr
Other properties

- Protocol support (signal/state feedback) – most already there (TBD MID+Version!)
- Junction Segment optimization - don't need on every router - leap-frog transit node(s) with SID list
- Protection - TI-LFA on each SID list as per usual
  - Node protection for Junction node (the BSID) is a more complicated discussion…
- Multi-egress -> yes.
- Multi-ingress -> … could be managed by controller in a top-down manner
- Broadcast-links -> yes
- Optimization – Local/global - DAG MBB:
  - Needs more considerations, procedures described and potentially protocol extensions
  - Can take the learnings and procedures from SR P2MP Policy

Informational or Standards track?

- No new extensions defined...yet
- Standards track for now – primarily informational content in document
  - May need extensions for things such as signalling MID+Version
  - May need extensions to deal with optimization/make before break scenarios

Next steps

- Open to discussions, feedback and collaboration
- Stay aligned with concepts in draft-kompella-teas-mpte as it evolves
- Decide on MID/Version/Color attribute – new segment type(?)
- Determine procedure details for optimization/MBB