Link Utilization & Convergence Considerations for SPB

Ali Sajassi, Mike Shand

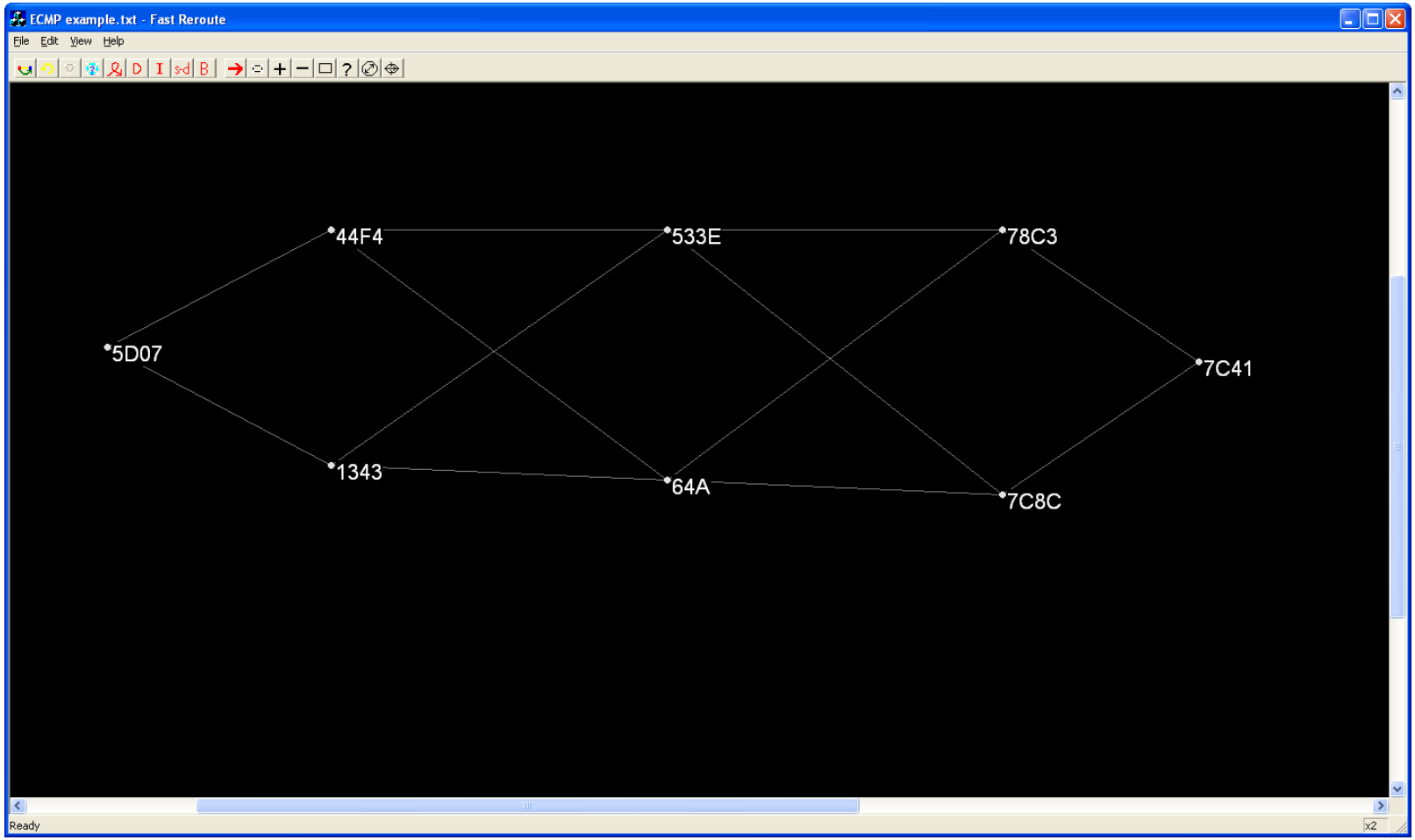
September 16, 2010

IEEE 802.1 Interim Meeting - York, U.K.

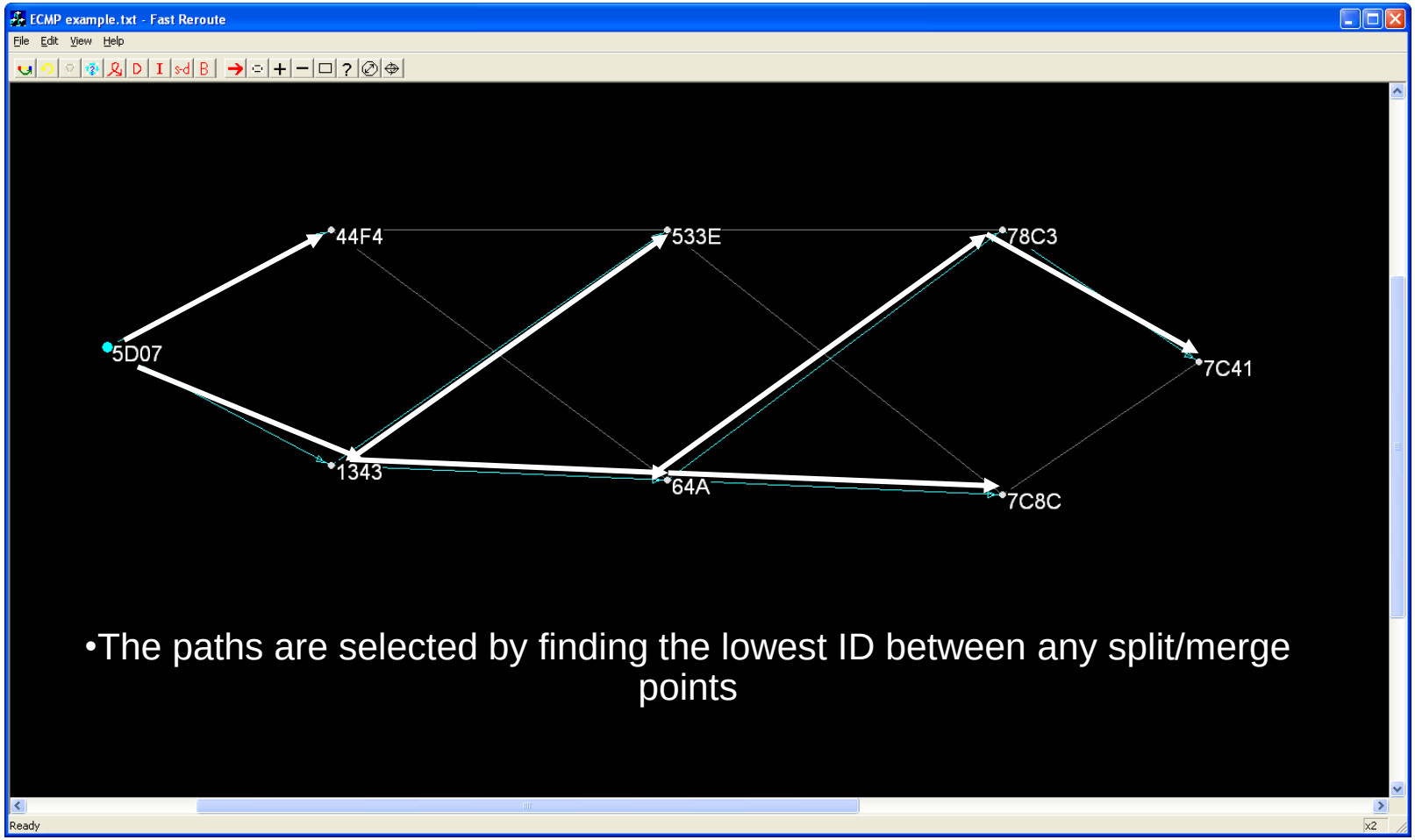
Agenda

- Link Utilization with Randomly Assigned Node IDs
- Link Utilization with Carefully Assigned Node IDs
- Link Utilization with per-hop hashing
- Convergence Time
- Conclusion

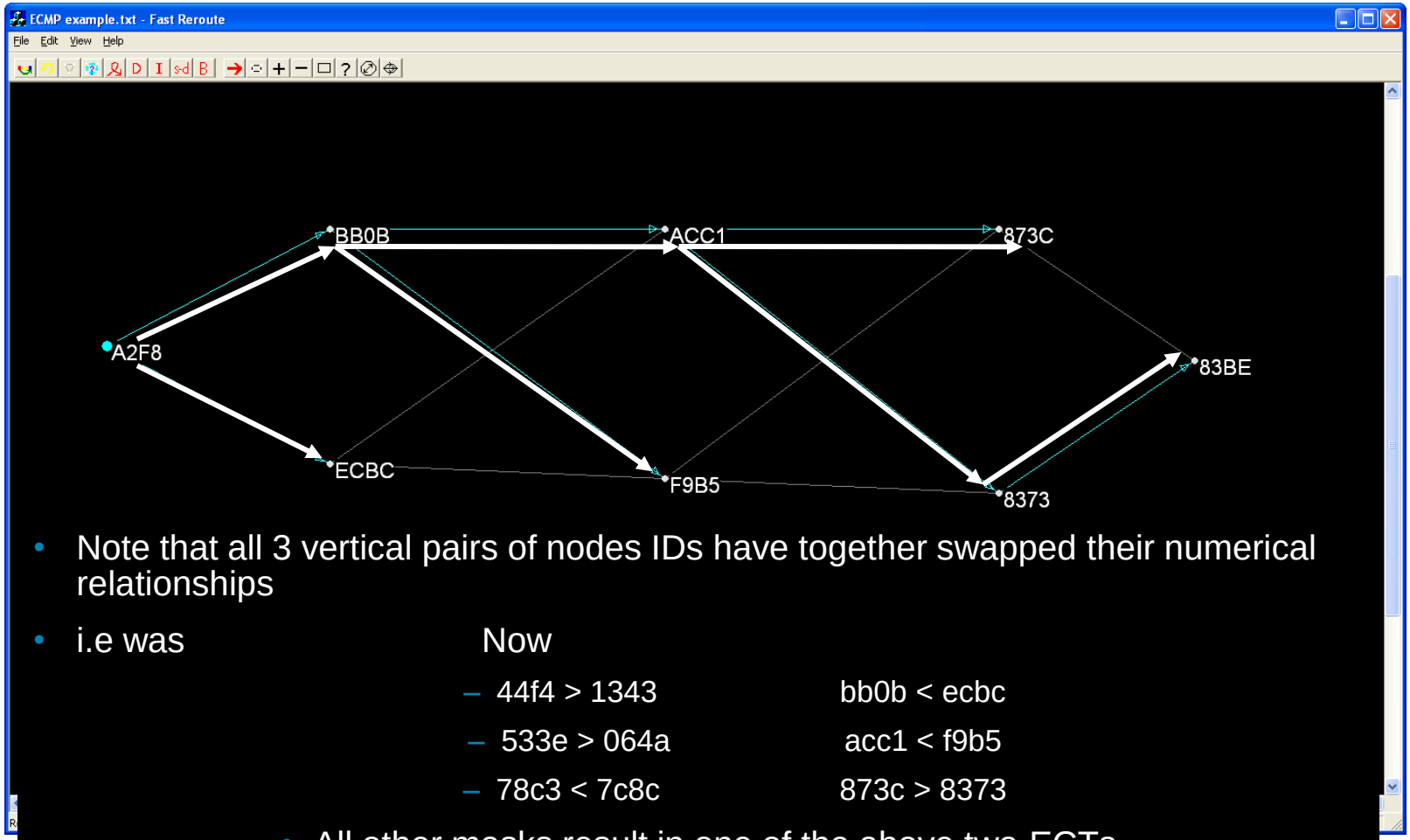
Consider the following topology with randomly assigned 16 bit node IDs



Primary ECT (mask 0x0000)

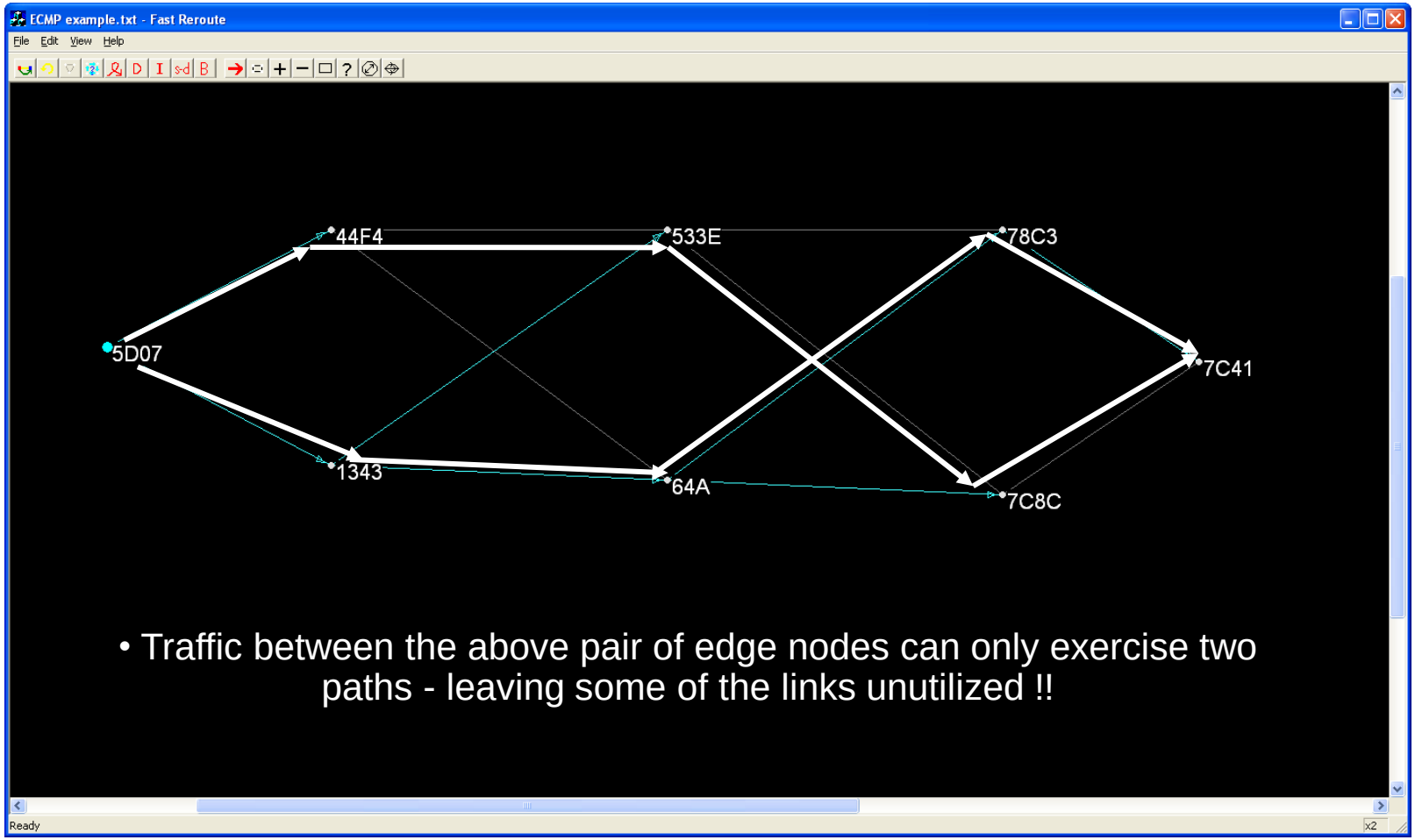


Xor IDs with 0xFFFF



- Note that all 3 vertical pairs of nodes IDs have together swapped their numerical relationships
- i.e was Now
  - 44f4 > 1343 bb0b < ecbc
  - 533e > 064a acc1 < f9b5
  - 78c3 < 7c8c 873c > 8373
- All other masks result in one of the above two ECTs

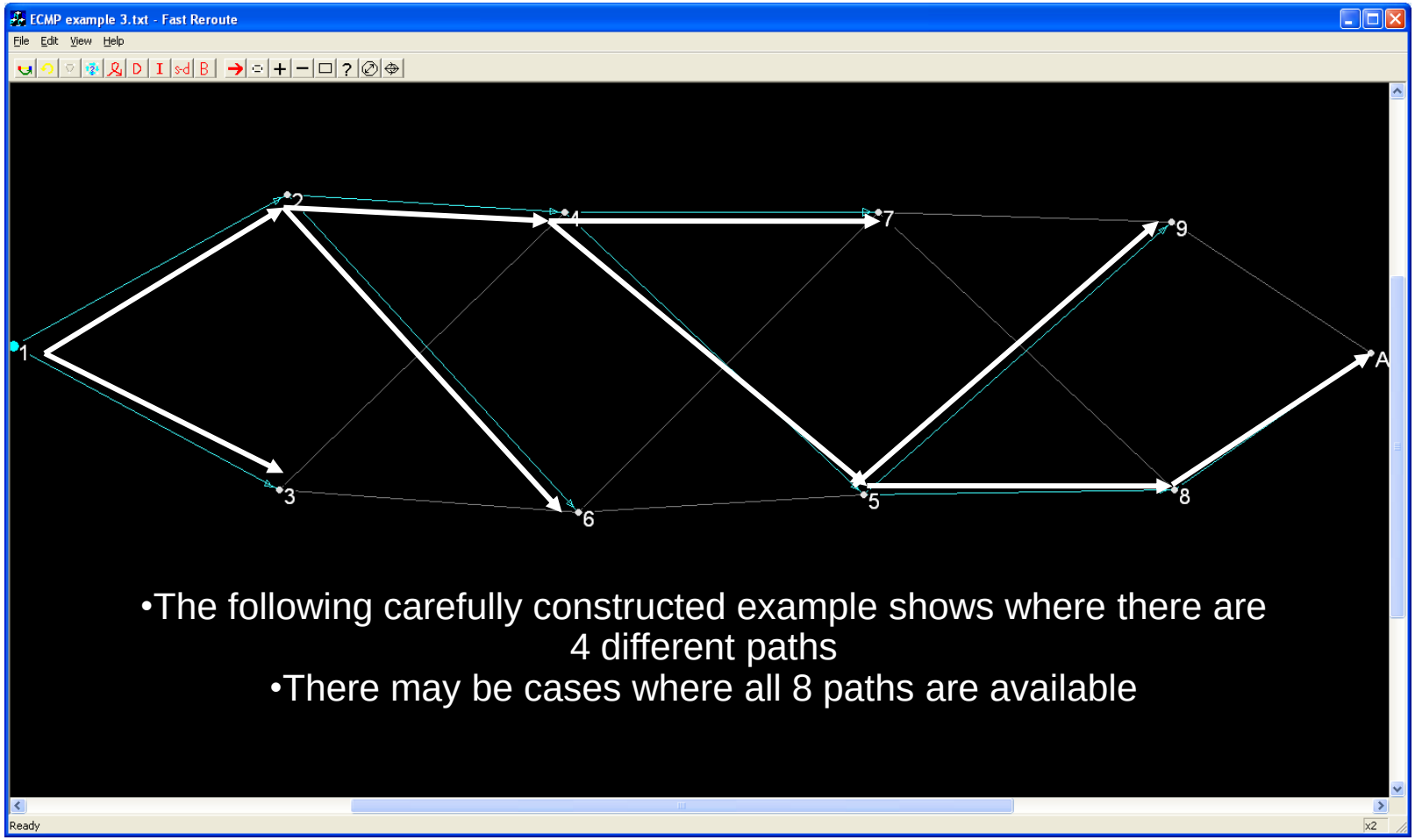
Link Utilization w/ Random Node ID Assignments



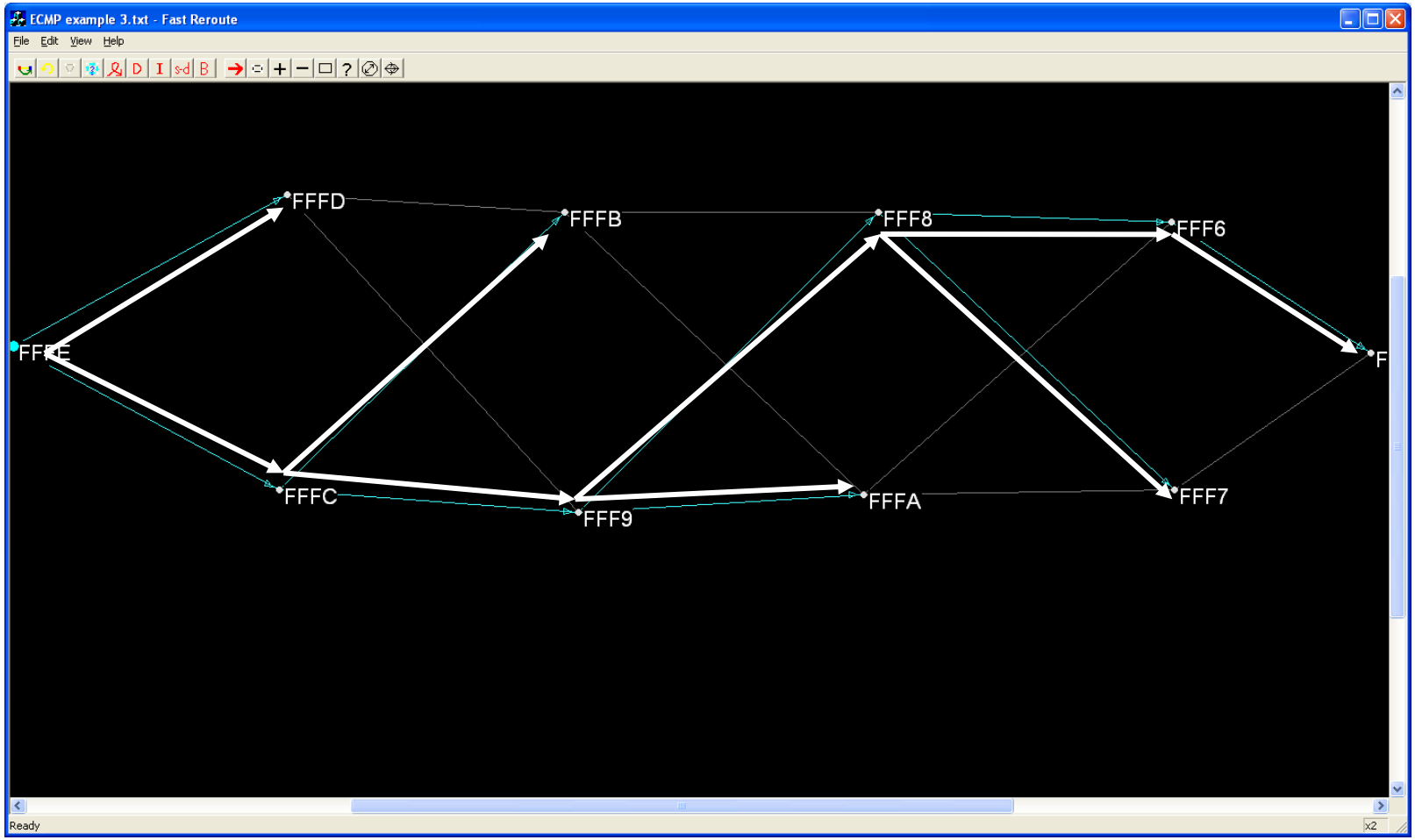
Agenda

- Link Utilization with Randomly Assigned Node IDs
- Link Utilization with Carefully Assigned Node IDs
- Link Utilization with per-hop hashing
- Convergence Time
- Conclusion

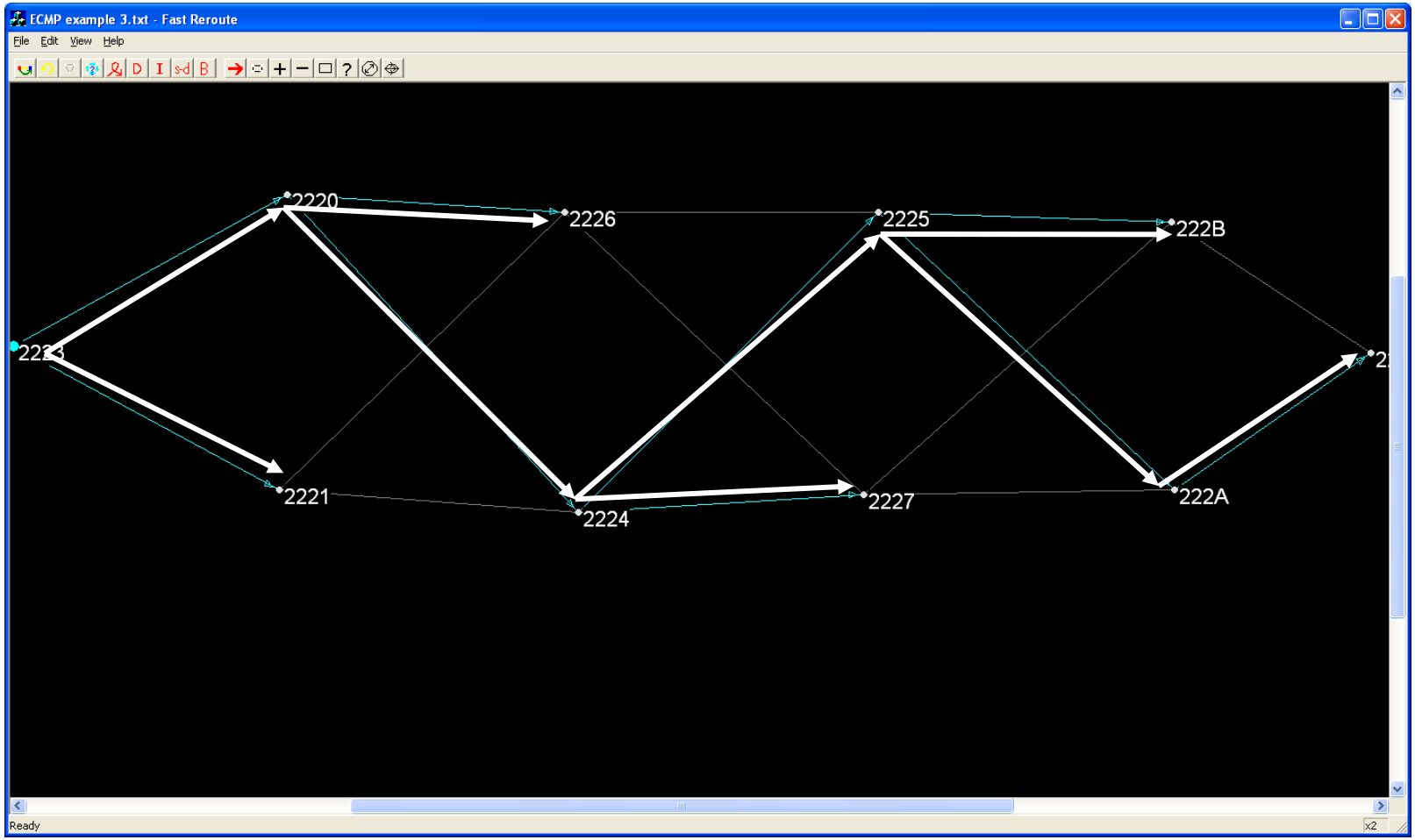
ECT-1


ECMP example 3.txt - Fast Reroute
File Edit View Help
1
2
3
4
5
6
7
8
9
A
•The following carefully constructed example shows where there are 4 different paths
•There may be cases where all 8 paths are available
Ready

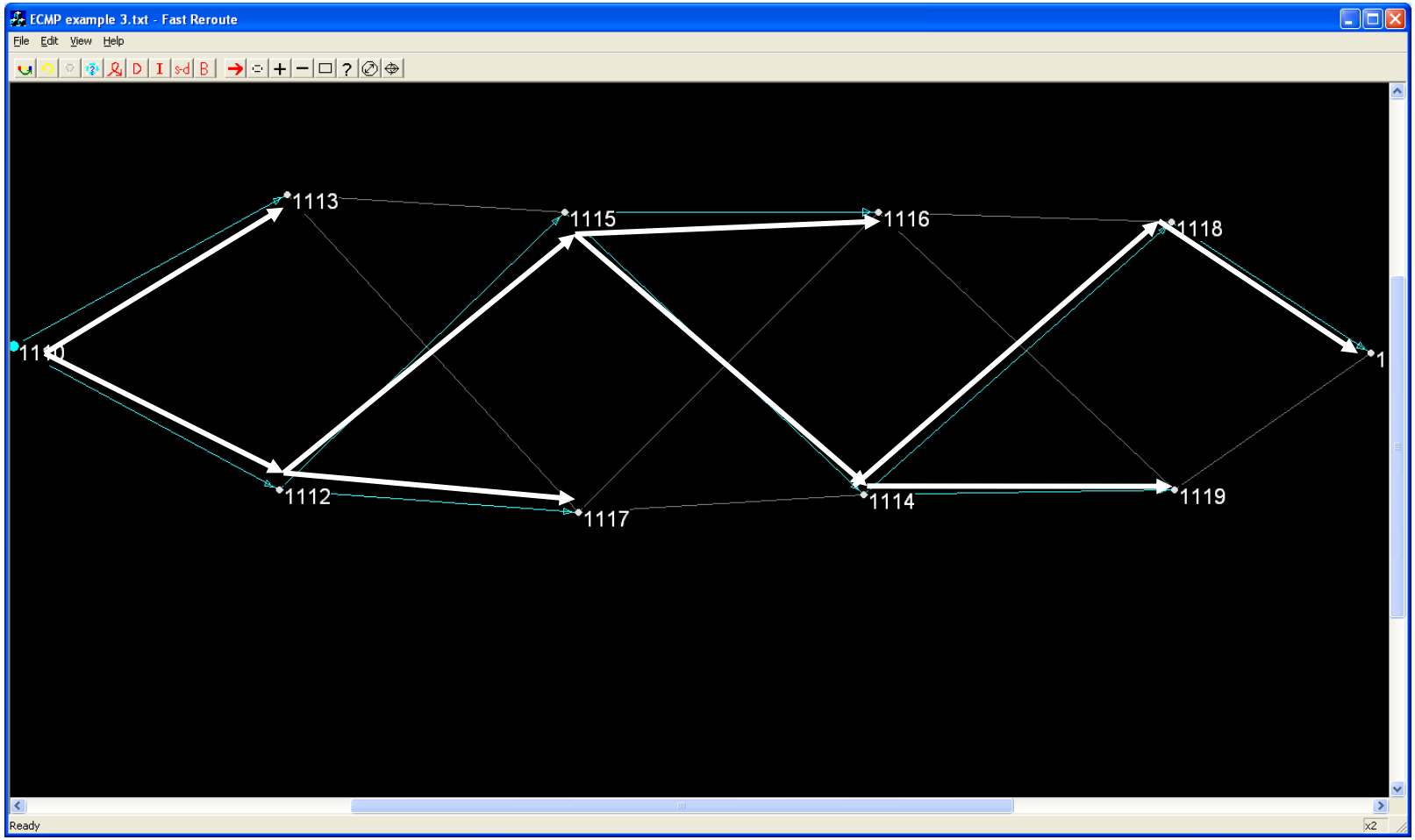
ECT-2


ECMP example 3.txt - Fast Reroute
File Edit View Help
FFFD
FFFB
FFF8
FFF6
FFFE
FFFC
FFF9
FFFA
FFF7
Ready

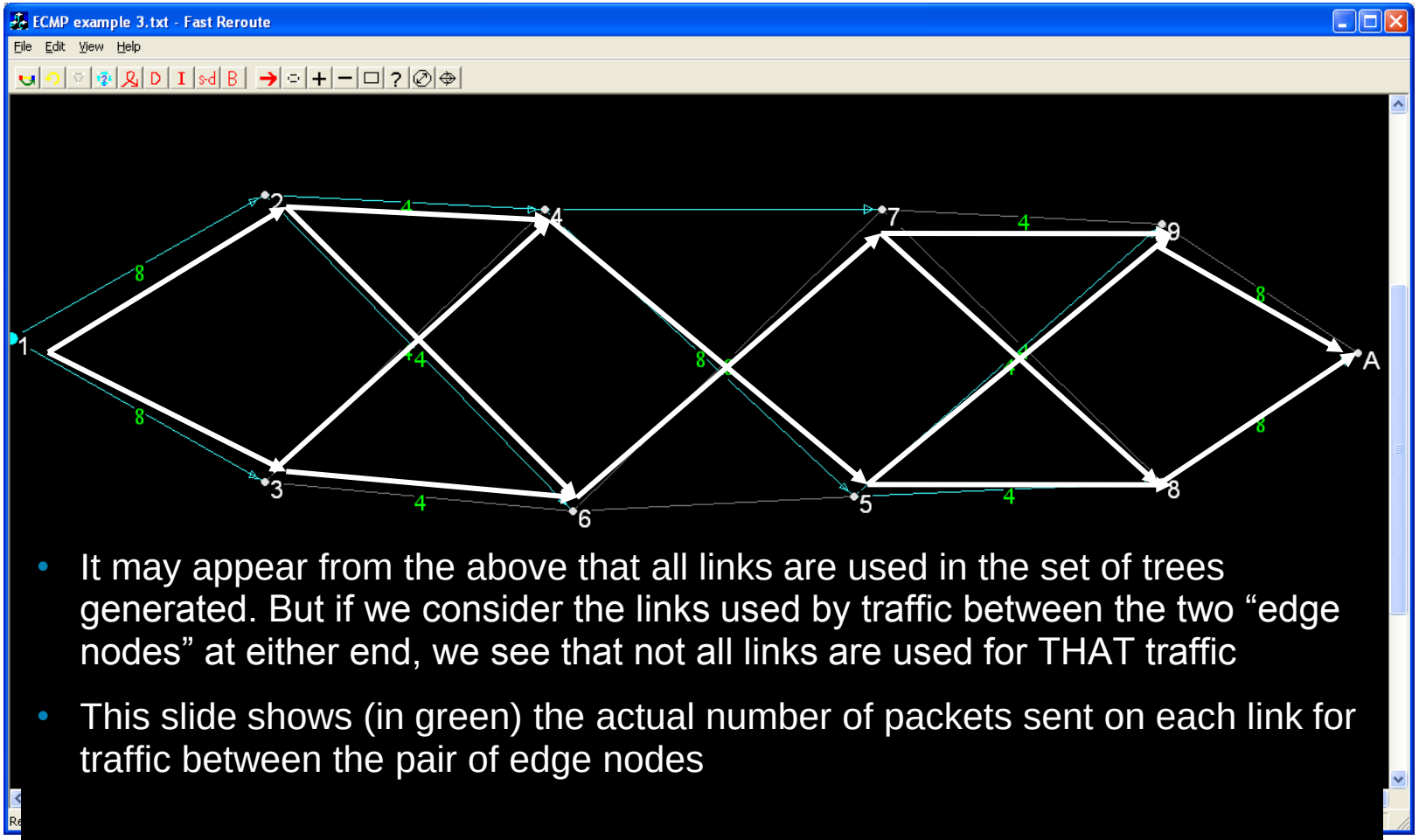
ECT-3


ECMP example 3.txt - Fast Reroute
File Edit View Help
2220
2226
2225
222B
2221
2224
2227
222A
Ready

ECT-4


ECMP example 3.txt - Fast Reroute
File Edit View Help
1113
1115
1116
1118
1112
1117
1114
1119
Ready

Link Utilization for Edge Traffic using .aq ECT

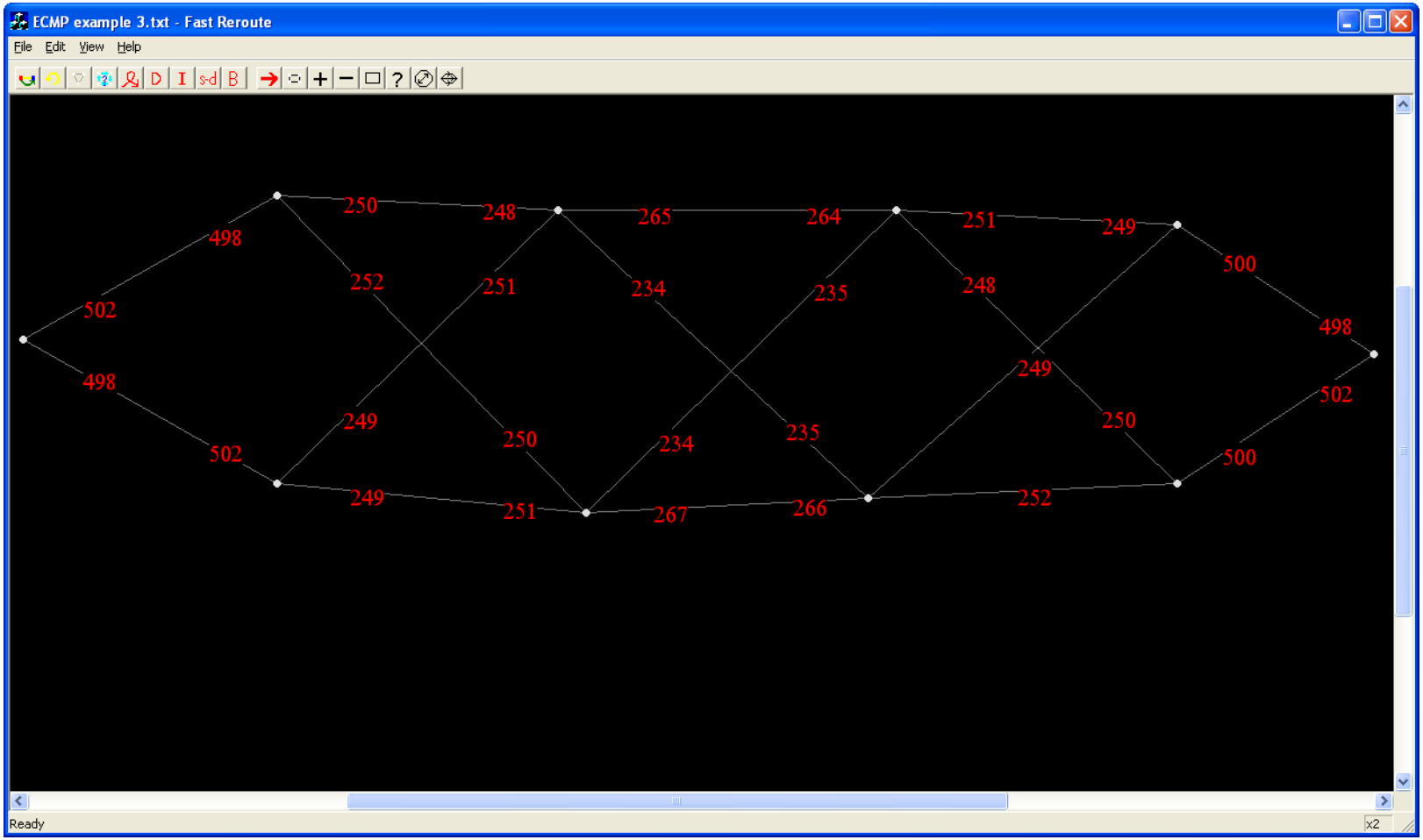


- It may appear from the above that all links are used in the set of trees generated. But if we consider the links used by traffic between the two “edge nodes” at either end, we see that not all links are used for THAT traffic
- This slide shows (in green) the actual number of packets sent on each link for traffic between the pair of edge nodes

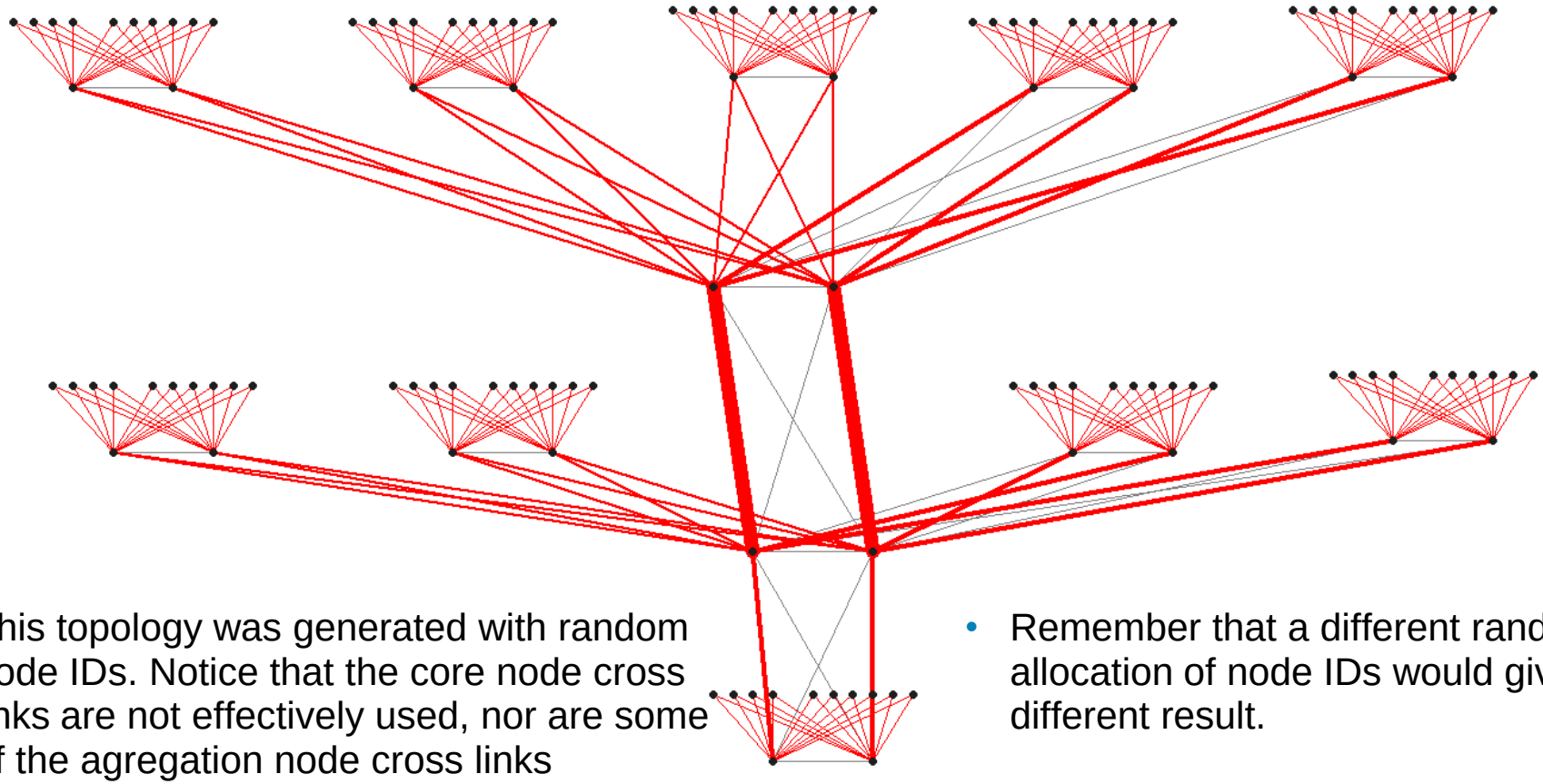
Agenda

- Link Utilization with Randomly Assigned Node IDs
- Link Utilization with Carefully Assigned Node IDs
- Link Utilization with per-hop hashing
- Convergence Time
- Conclusion

Link Utilization for Edge Traffic using per-hop Hashing

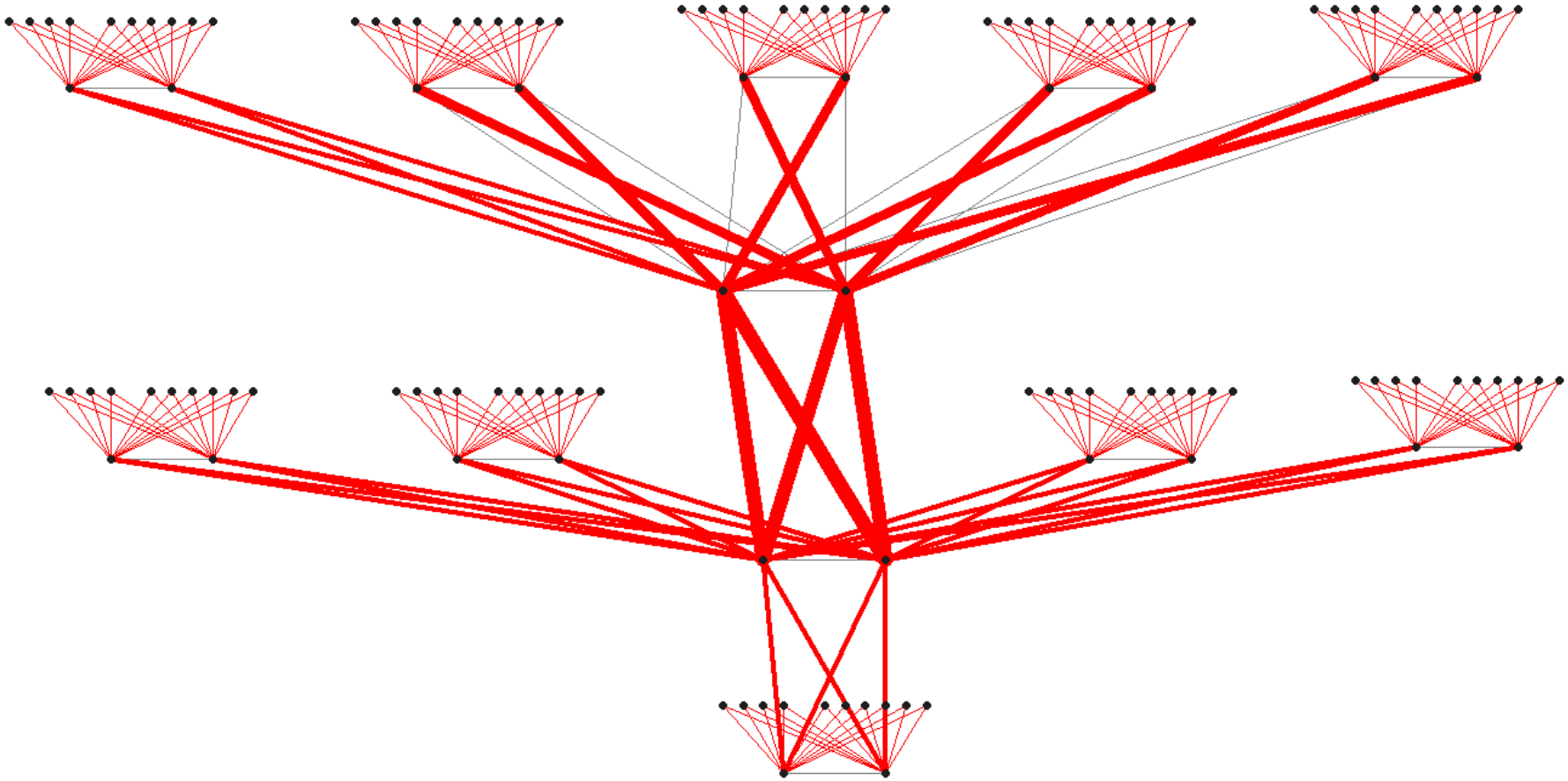


Link Utilization w/ Randomly Assigned Node IDs for a DC Network topology

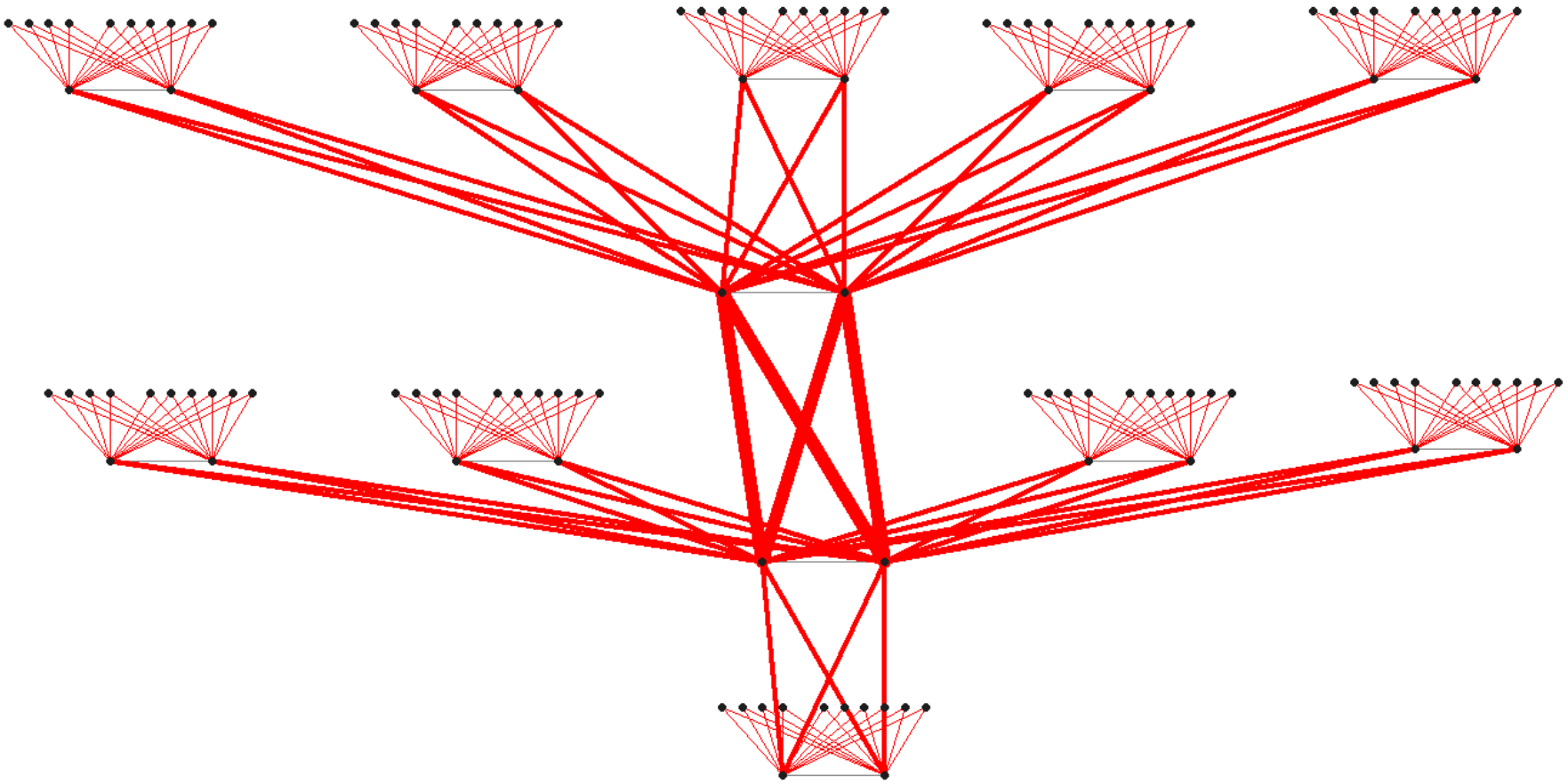


- This topology was generated with random node IDs. Notice that the core node cross links are not effectively used, nor are some of the agregation node cross links
- Remember that a different random allocation of node IDs would give a different result.

Link Utilization w/ Carefully Assigned Node IDs for a DC Network Topology



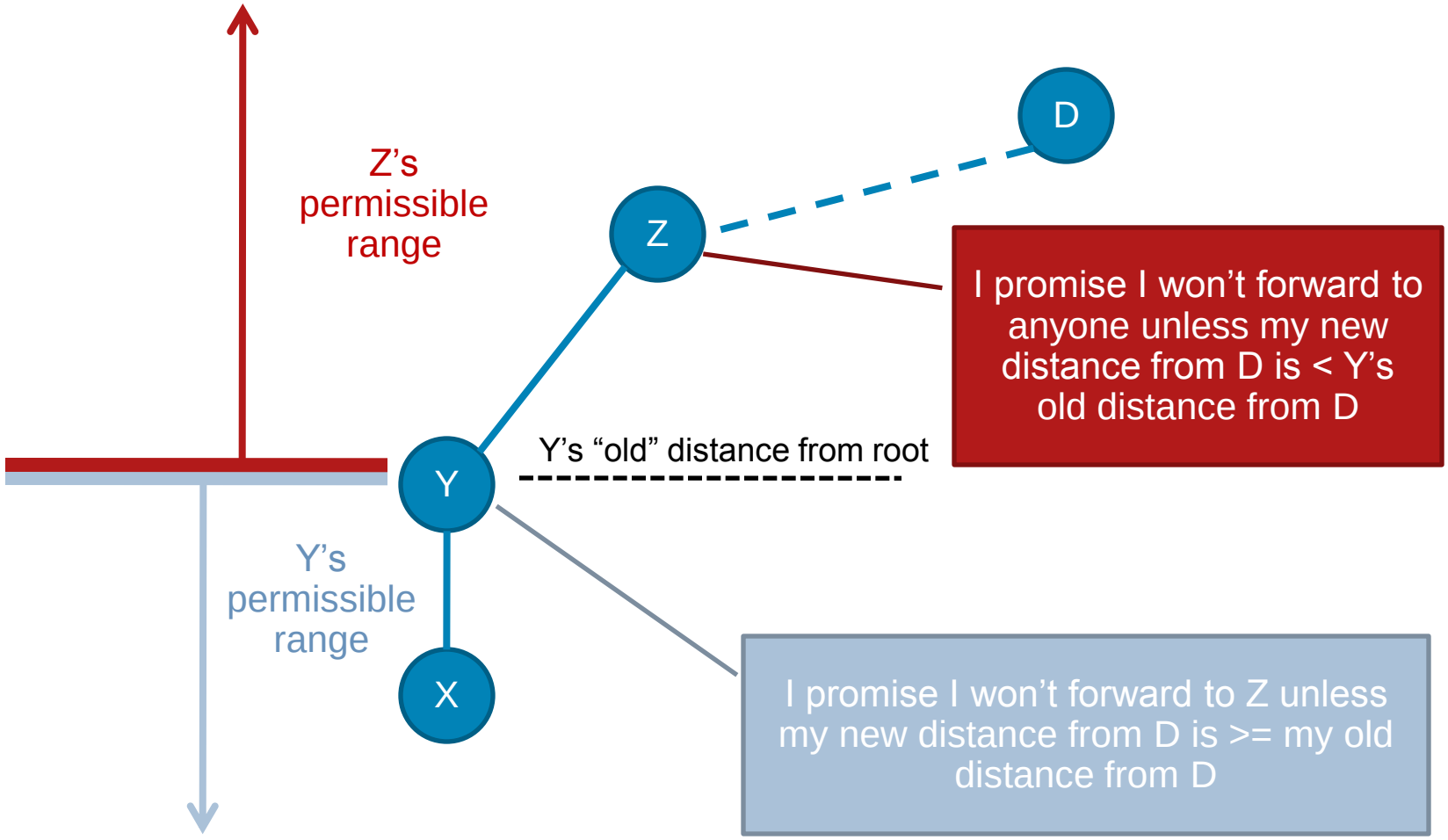
Link Utilization w/ per-hop Hashing for a DC network Topology



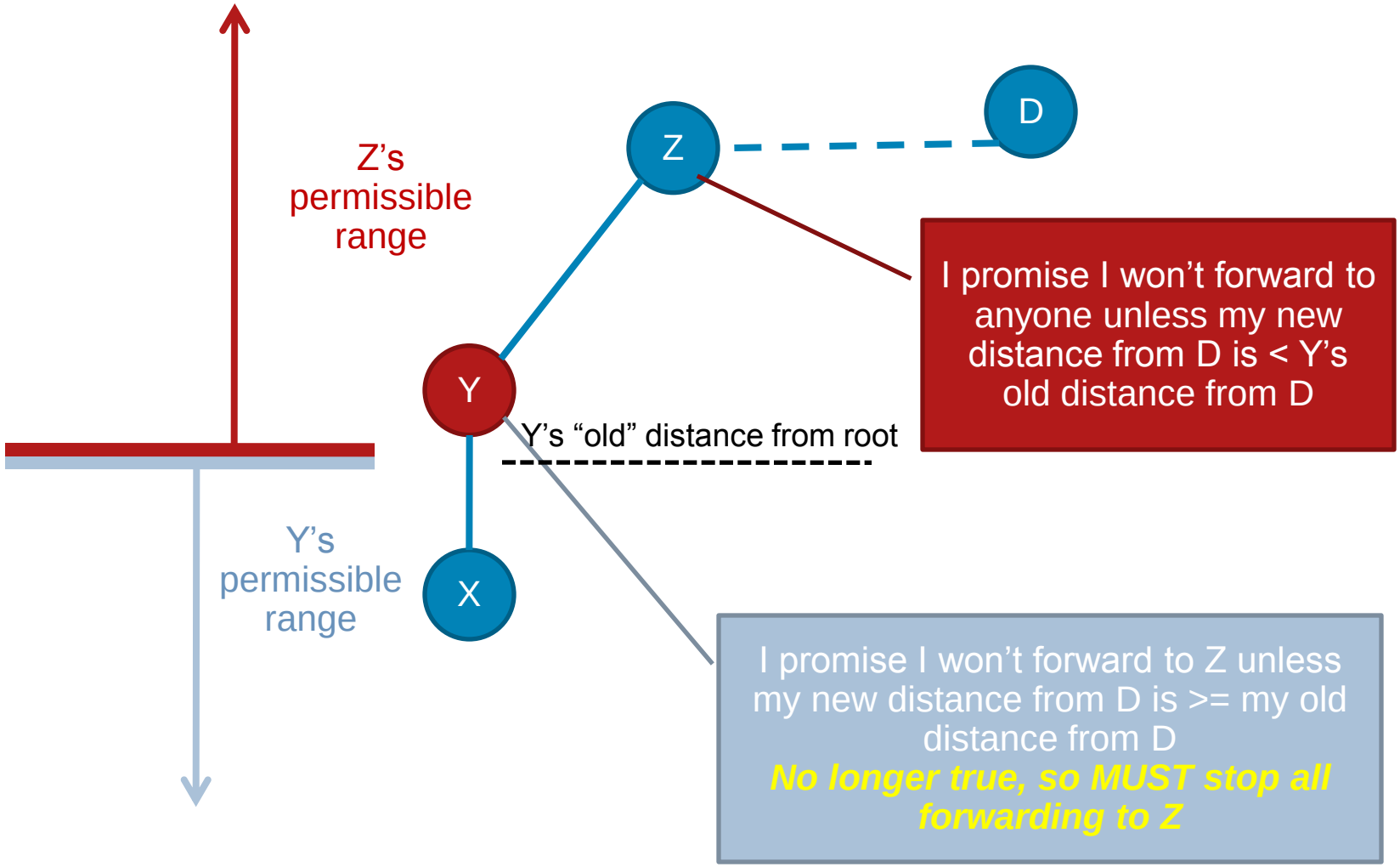
Agenda

- Link Utilization with Randomly Assigned Node IDs
- Link Utilization with Carefully Assigned Node IDs
- Link Utilization with per-hop hashing
- Convergence Time
- Conclusion

Forwarding Rules

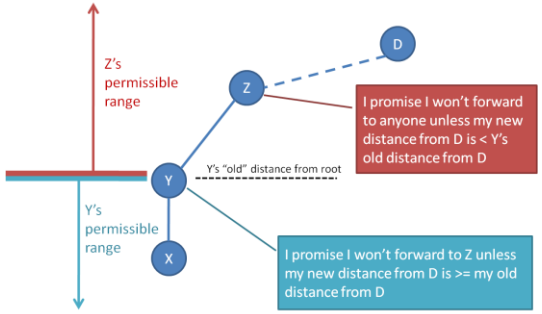


If Z-D decreases…



Why?

- Z guarantees that it will stop forwarding if it is NOT closer to D than Y's old position.
- If it moves further away than Y, a loop could form since Y COULD be downstream of Z, either directly (a loop across ZY), or indirectly (a loop via some other nodes)



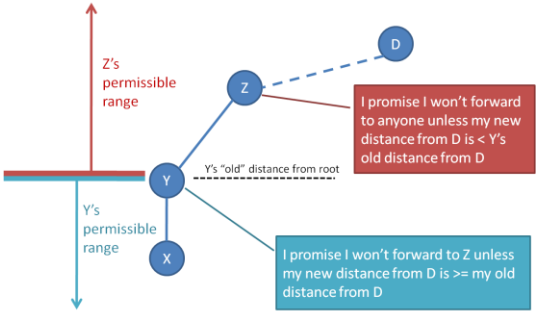
So why does it matter if Y moves closer?

- If Y moves ANY closer than its old position

  A loop **cannot** form if Z doesn't move, provided that Y doesn't actually move closer than Z

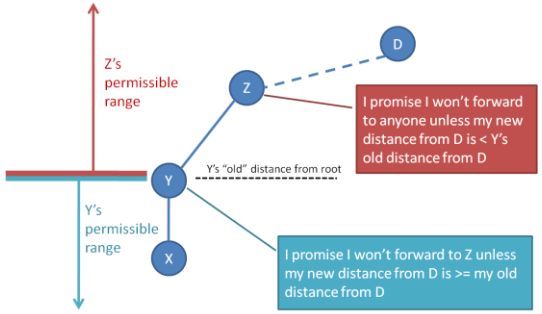
  So why does it matter?

- Z has only agreed to stop forwarding if it doesn't move further away that Y's **old** position, **NOT its new position**

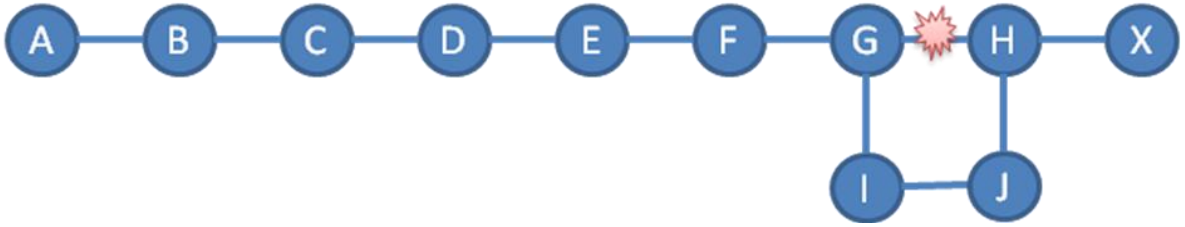


Multiple moves (from multiple topology changes) are possible

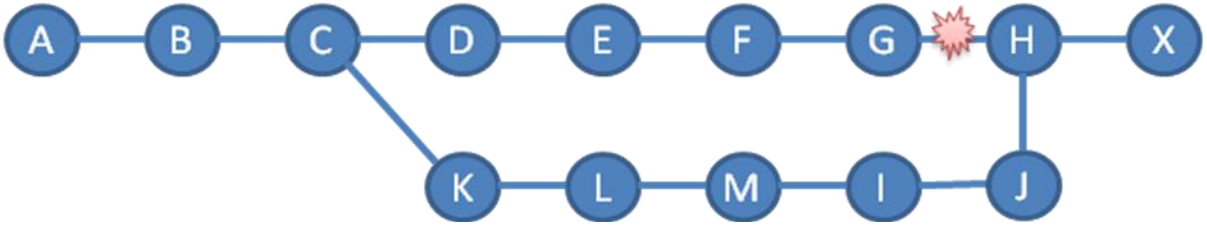
- If Y moves closer than its old position
- **AND** Z moves further away than Y's new position (which it is allowed to do)
- A loop could form
- So Y **MUST** stop forwarding if it moves any closer.



Convergence Time



Topology 1

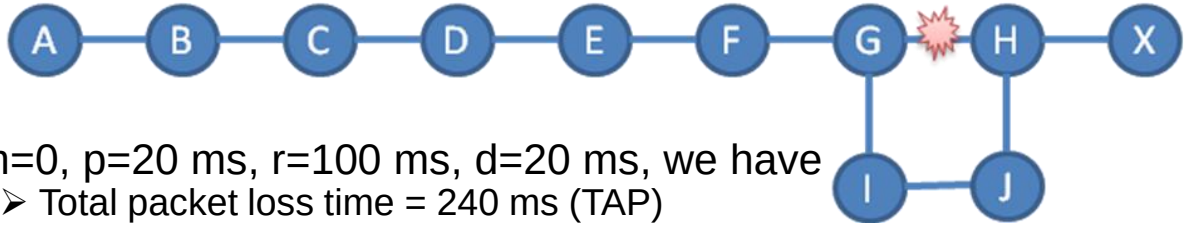


Topology 2

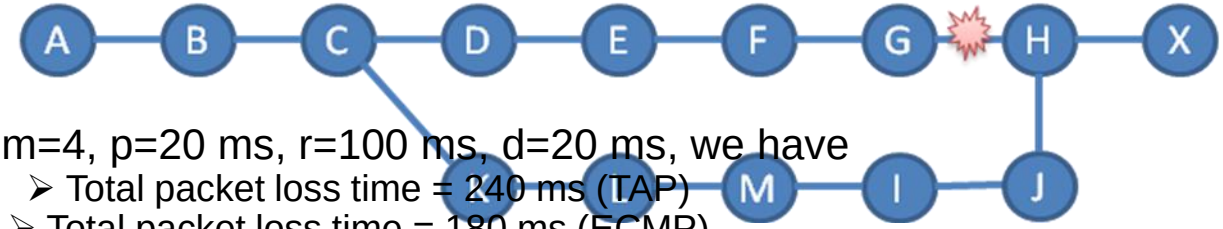
n = # of hops upstream of the failure having an inter-node cost less than the increase in cost caused by the failure
m = # of hops upstream of the failure before the re-route point
p = the inter-node LSP propagation time
r = the inter-node resynchronization time (i.e. the time for the databases and FIBs to be back in sync following a new LSP event)
d = the additional time to exchange digests in .aq

Convergence Time upon Failure

- with 802.1aq TAP : Total packet loss time = n*p+r+d
- with ECMP: Total packet loss time = m*p+r



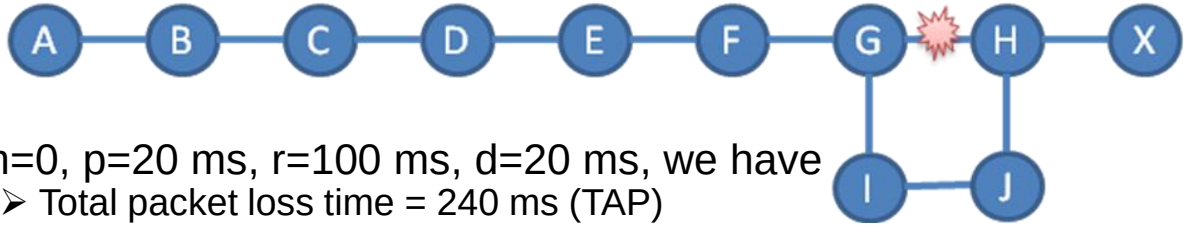
- Example, for n=6, m=0, p=20 ms, r=100 ms, d=20 ms, we have
  - Total packet loss time = 240 ms (TAP)
  - Total packet loss time = 0 ms (ECMP)



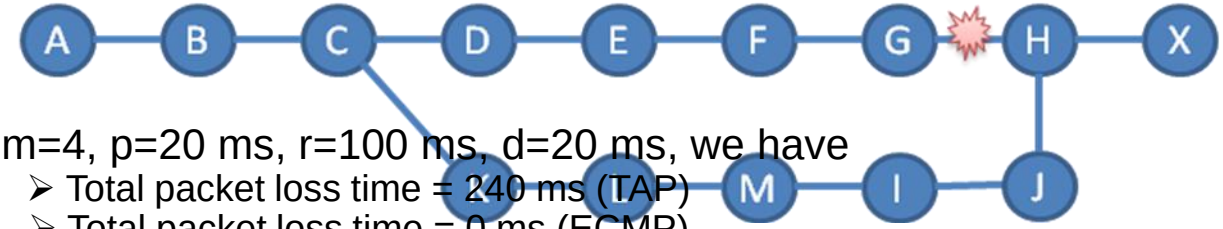
- Example, for n=6, m=4, p=20 ms, r=100 ms, d=20 ms, we have
  - Total packet loss time = 240 ms (TAP)
  - Total packet loss time = 180 ms (ECMP)

Convergence Time upon Recovery

- with 802.1aq TAP : Total packet loss time = n*p+r+d
- with ECMP: Total packet loss time = m*p+r



- Example: for n=6, m=0, p=20 ms, r=100 ms, d=20 ms, we have
  - Total packet loss time = 240 ms (TAP)
  - Total packet loss time = 0 ms (ECMP)



- Example: for n=6, m=4, p=20 ms, r=100 ms, d=20 ms, we have
  - Total packet loss time = 240 ms (TAP)
  - Total packet loss time = 0 ms (ECMP)

Agenda

- Link Utilization with Randomly Assigned Node IDs
- Link Utilization with Carefully Assigned Node IDs
- Link Utilization with per-hop hashing
- Convergence Time
- Conclusion

Conclusion/Recommendation

- ECMP helps with both link utilization and convergence time
- ECMP can be used for applications where utilizing links EVENLY in the network is important
- Use TTL to achieve ECMP via per-hop hashing
  - Simple to do & explain !!
  - Proven method
  - Every interested party (operator) is familiar with
  - Can easily be incorporated into a new I-tag
  - Simple to implement for most vendors (hashing function already exists because of LACP)