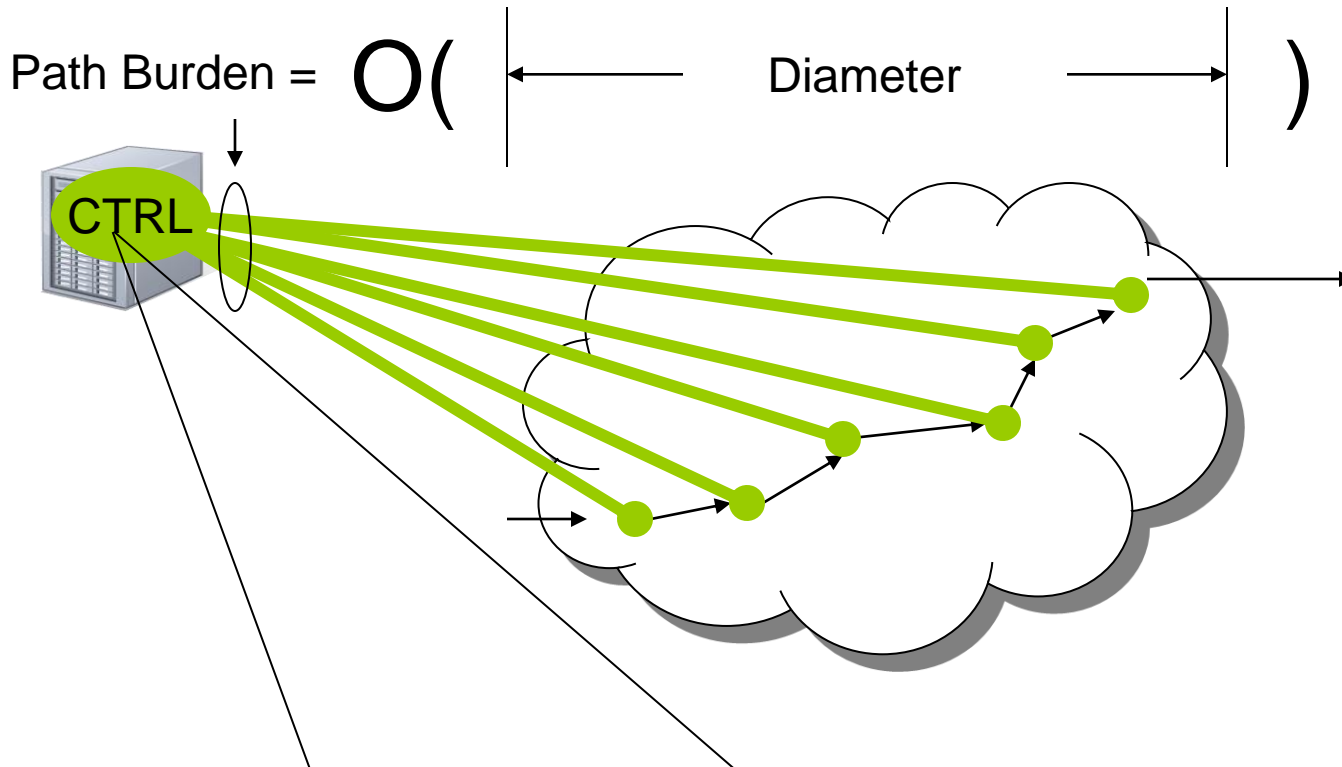# draft-ashwood-sdnrg-state-reduction-00

Peter Ashwood-Smith
peter.ashwoodsmith@huawei.com

Mourad Soliman
MouradSoliman@cmail.carleton.ca

Central Controller Burden Per Path

Path Burden = $O(\;|\longleftarrow \text{Diameter} \longrightarrow|\;)$

CTRL

A controller has to send path state to **<u>every</u>** device along the path.

Therefore as the diameter of the network grows the controller scale drops because its work/burden grows as function of **O(diameter)**

Example 1000 nodes with 10 paths per (S,D) pair with 5 hops each has Burden = O(50,000,000)

Tackle the distributed state problem by ...

Path Burden = $O($ 1 $)$

CTRL
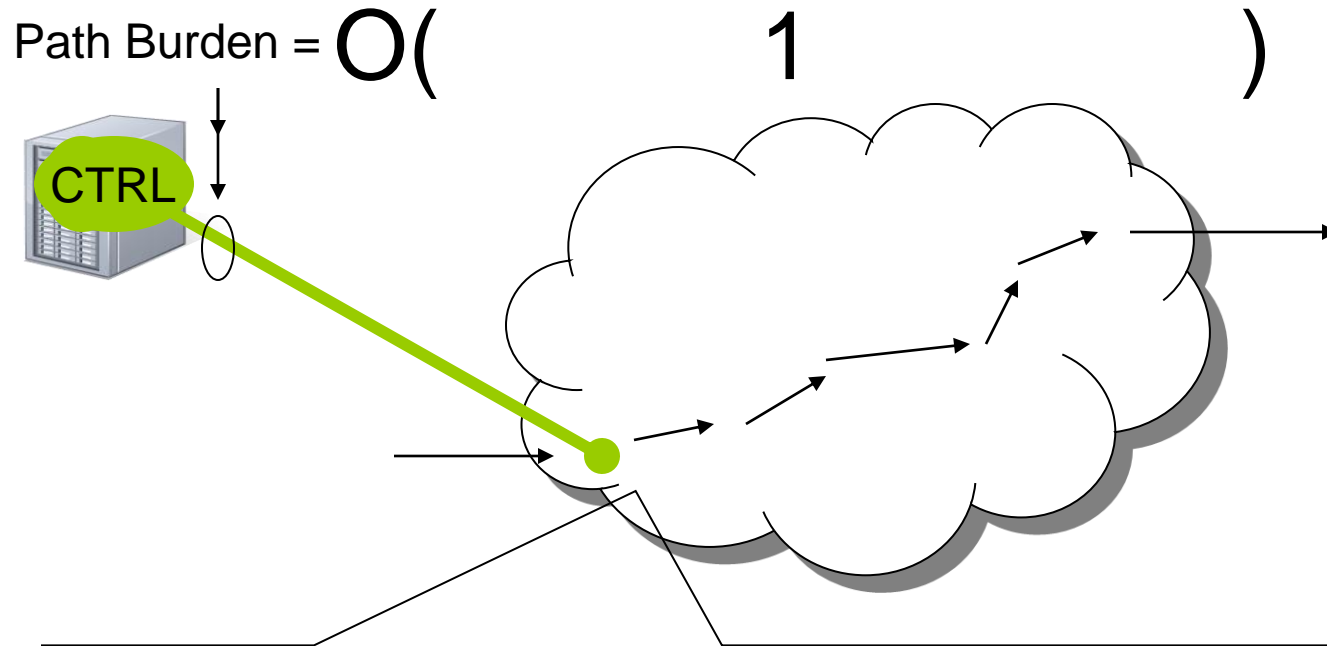
Install all state required to traverse network in only the ingress device.

Basically ingress switches attach a source route (links) to follow and tandem devices recognize and **strictly** follow it.

Simplest solution is a list of locally unique link identifiers & index.
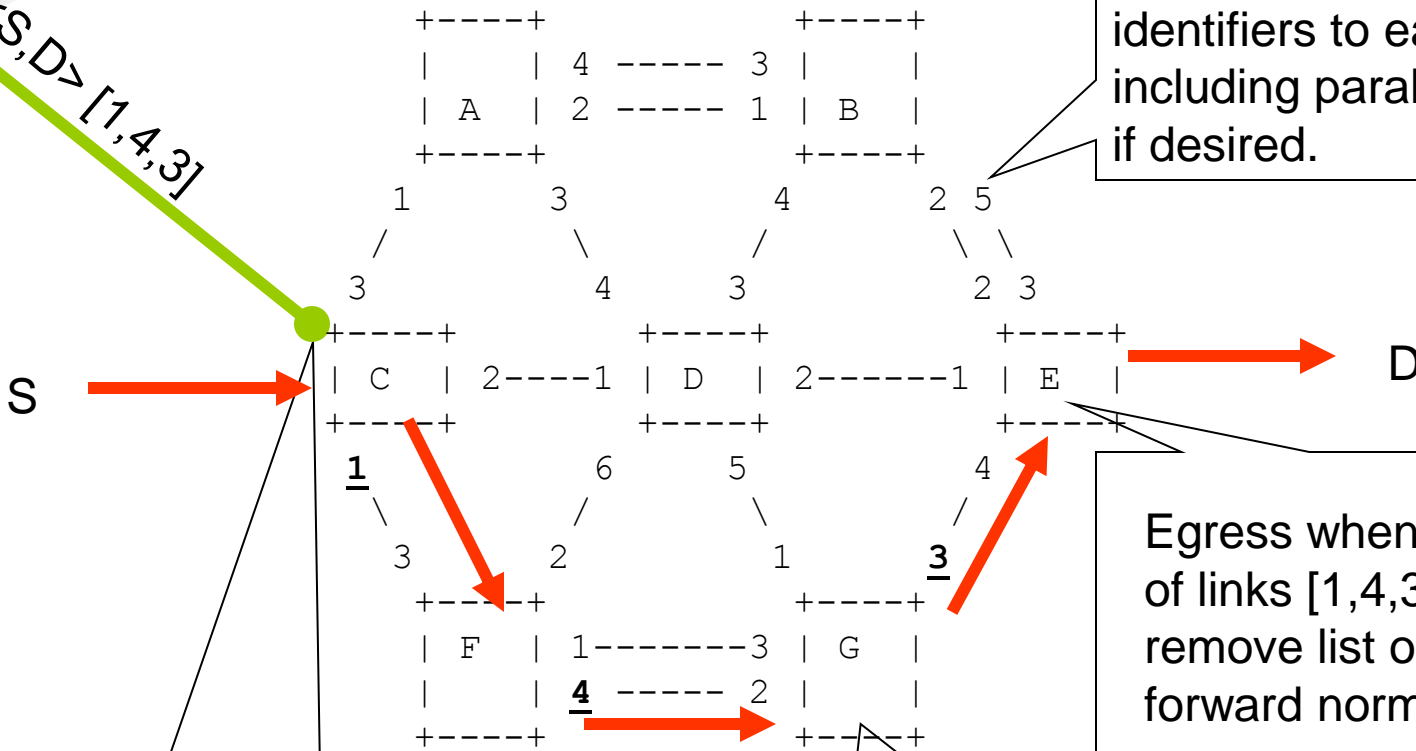Idealized = Strict Link Source Routing **SLSR**

Example 1000 nodes with 10 paths per S/D pair, 5 hops each has Burden = O(5,000,000)

Strict Link Source Routing - SLSR

CTRL

\<S,D\> [1,4,3]

```
                    +----+              +----+
                    |    | 4 ----- 3 |    |
                    | A  | 2 ----- 1 | B  |
                    +----+              +----+
                1         3           4      2 5
               /           \         /        \ \
              3             4       3          2 3
            +----+        +----+        +----+
  S         | C  | 2----1 | D  | 2------1 | E  |              D
            +----+        +----+        +----+
            1                6       5          4
             \             /         \        /
              3           2           1       3
            +----+              +----+
            | F  | 1-------3 | G  |
            |    | 4 ----- 2 |    |
            +----+              +----+
```

1 Nodes assign <u>local</u> identifiers to each link including parallel links if desired.

2 Ingress node, configured by controller installs \<S,D\> flow entry and header with list of links to follow [<u>1</u>,4,3]

**3** Tandem nodes have no state related to path. Just take list [1,4,<u>3</u>], extract current next link 3 and forward to that link.

4 Egress when at end of links [1,4,3,_] remove list of links forward normally.

SDN State Reduction Simulation
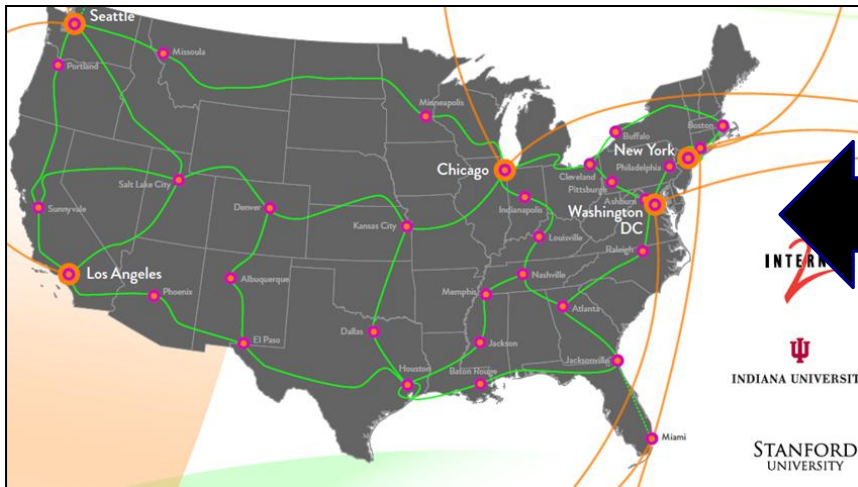
We simulated a network of 36 nodes, various flows/paths and compared source routed vs traditional forwarding.

Observed 3 x Improvement

Source-Routed OpenFlow

Convergence time for each source
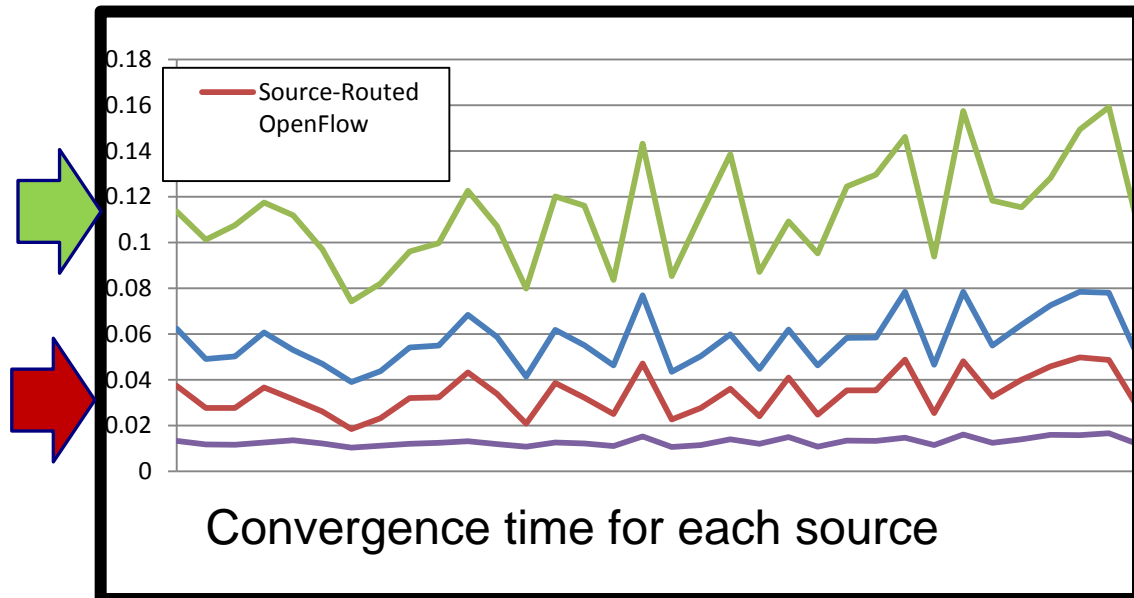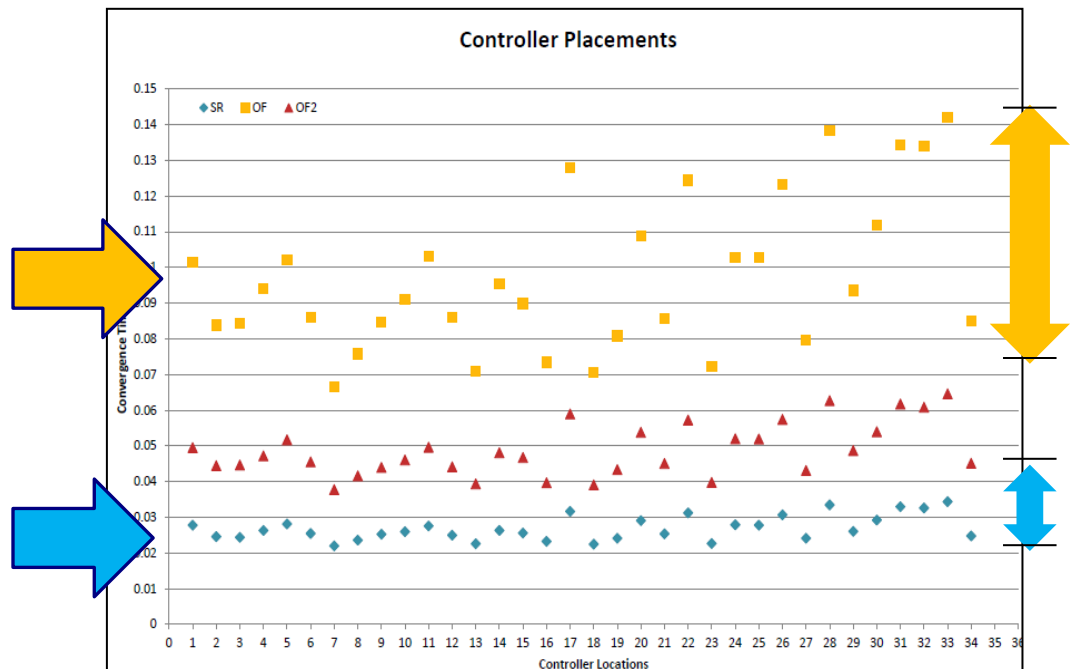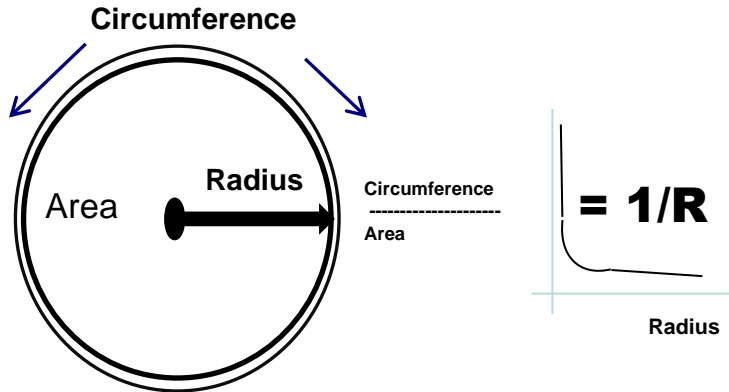
Observed 86 % reduced standard deviation in convergence for all controller placements.

SDN is very sensitive to controller position.

Source Routing reduces sensitivity by 86%

Controller Placements

Observed 1/R behavior relative to hop by hop

Circumference

Area **Radius** Circumference

Area $= 1/R$

Radius

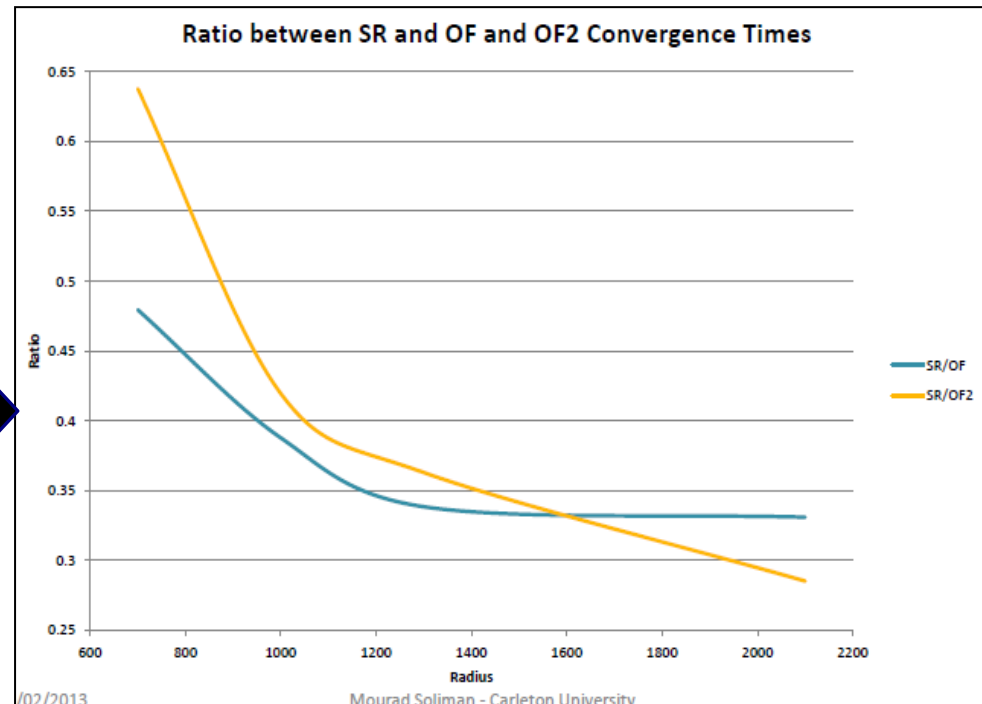**Source Route Burden is: O(Circumference)**

Hop by Hop Burden is: O(Area)

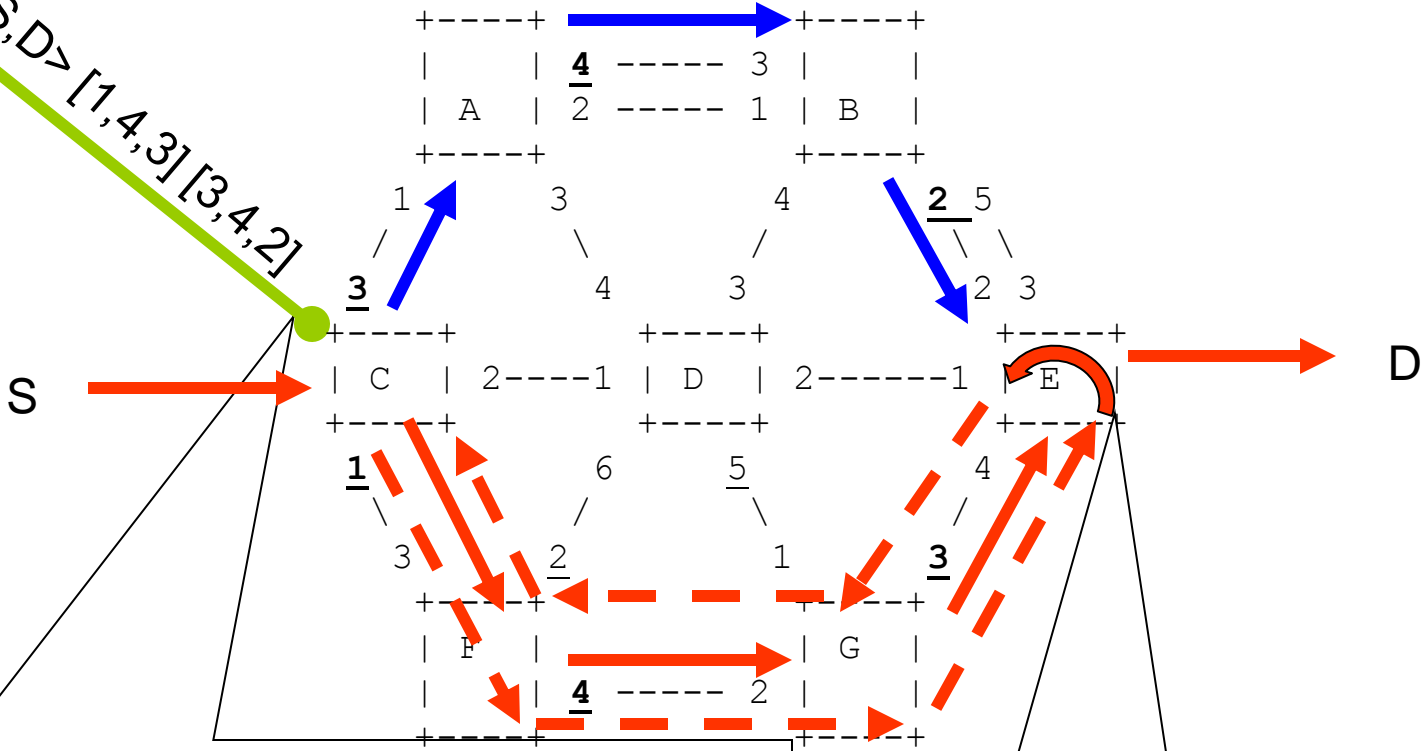**Plotted Source Route convergence divided by hop by hop convergence for different network sizes.**

We observed 1/R trend.

Ratio between SR and OF and OF2 Convergence Times

SR/OF
SR/OF2

Ratio

Radius

/02/2013   Mourad Soliman - Carleton University

Strict Link Source Routing – Protection – Head End

S

<S,D> [1,4,3] [3,4,2]

```
        +----+   ────────►   +----+
        |    |  4 ----- 3 |    |
        | A  |  2 ----- 1 | B  |
        +----+            +----+
      1      3       4      2 5
     /        \     /      / \
    3          4   3      2 3
  +----+    +----+    +----+
  | C  | 2----1 | D  | 2------1 | E |
  +----+    +----+    +----+
  1        6      5      4
   \      /        \    /
  3      2         1   3
  +----+    +----+
  | F  |    | G  |
  |    |  4 ----- 2 |    |
  +----+    +----+
```

S

D

1

Ingress node, configured
by controller installs <S,D> flow
entry and pushes PRIMARY list of
links to follow [1,4,3] and BACKUP   [3,4,2]
runs probe on [1,4,3], on fail switch to [3,4,2]

2

Egress Node returns probes
over reverse source routes.

Strict Link Source Routing – Protection - Local

S

<S,D> [1,4,3]

```
        +----+              +----+
        |    | 4 ----- 3 |    |
        | A  | 2 ----- 1 | B  |
        +----+              +----+
     1        3            4      2 5
    /          \          /        \ \
   3            4        3          2
 +----+       +----+    +----+
 | C  | 2----1 | D  | 2----    | E  |
 +----+       +----+          +----+
   1            6       5        4
    \          /         \      /
   3          2          1      3
 +----+              +----+
 | F  |              | G  |
 |    | 4 ---- 2 |    |
 +----+              +----+
```

S

D

3 On detour packet looks like any other SLSR packet. [1,2,5,3]

2 Ingress node, configured by controller installs <S,D> flow entry and pushes list of links to follow [1,4,3]

1 Nodes assign detours as just list of links to get around a failed link..eg if 4 fails replace with [2,5]

7/3/2013

9
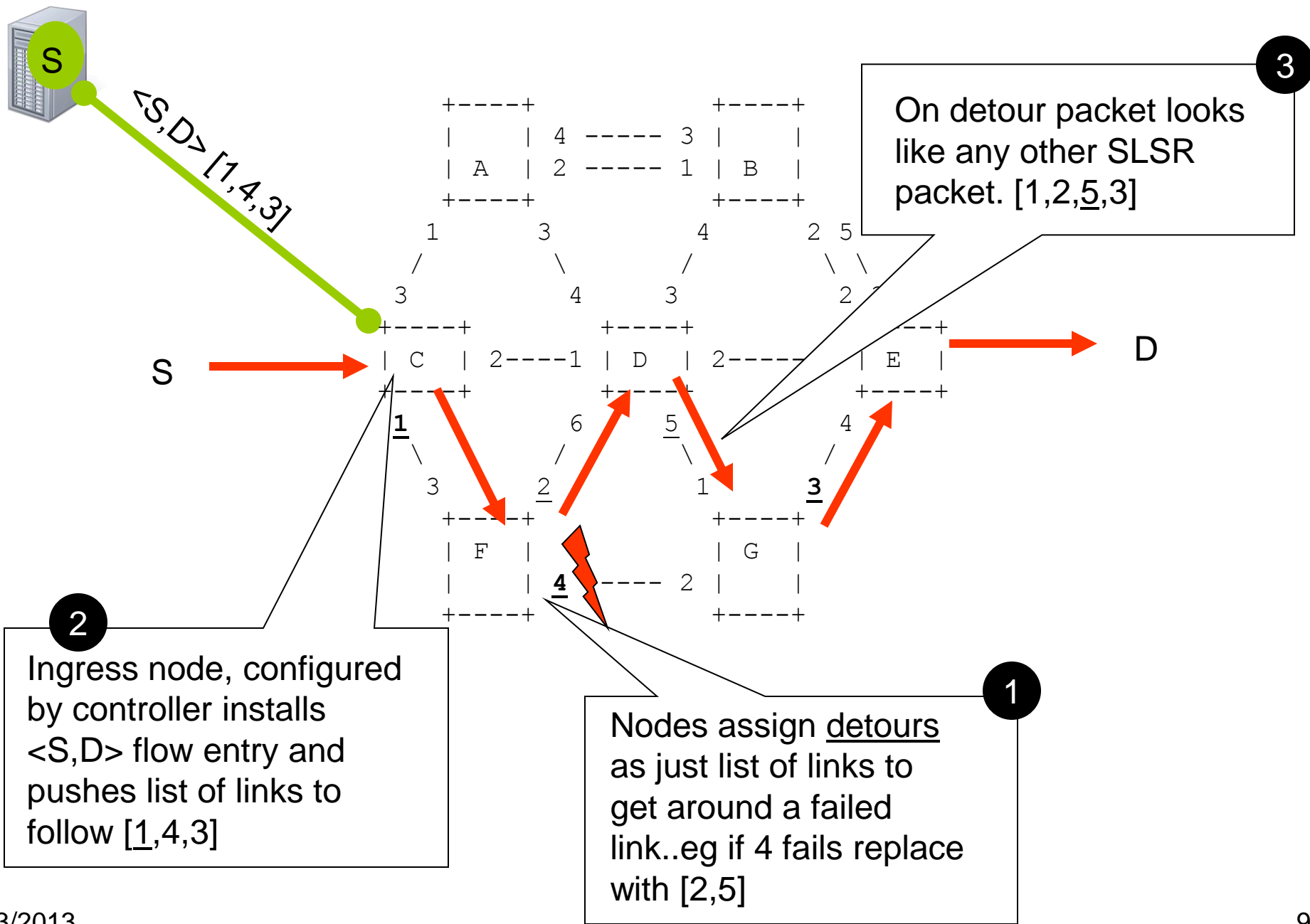
Implementation Options Comparison

Property	MPLS Stack = "Segment Routing"	New via POF = "Protocol Oblivious Forwarding"
Index size	20bits	Variable
Overhead 512 byte/5 hops	4%->3%->2%->1%	1% fixed
Reversible	No	Yes
Fast Reroute	Yes	Yes
1+1	Reverse probe?	Yes
Max Hops	H/W dependent	H/W dependent

POF is very flexible, so easy to explore MPLS changes..

❑**To reduce overhead could we:**
  ✓ **Define a 'mini' MPLS label, 16bits no TTL, no EXP?**

❑**To preserve hop history could we:**
  ✓ **Define a cycle() operation, ie pop() and append below BOS?**

Thank You

References

SDN State Reduction
>   tools.ietf.org/html/draft-ashwood-sdnrg-state-reduction-00

SDN Controller State Distribution Reduction
>   www.ieee802.org/1/files/public/docs2012/new-ashwood-sdn-optimizations-0712-v01.pdf

Protocol Oblivious Forwarding:
>   www.poforwarding.org/

Segment Routing:
>   tools.ietf.org/html/draft-previdi-filsfils-isis-segment-routing-00

Source Routing Forwarding with SDN:
>   conferences.sigcomm.org/co-next/2012/eproceedings/student/p43.pdf