Reconfigurable Network Testbed for Evaluation of Datacenter Topologies

W. Clay Moody
Clemson University

Co authors: Jason Anderson, K.-C. Wang, Amy Apon

Our Paper (briefly)

- **Word Cloud:**

well
using
one
run
set
switches
Flow
performance
edge
data
4
rules
1
four
time
Hadoop
real
low
traffic
key
29
two
per
network
map
Line
use
testbed
also
HPC
topology
FORCE
node
tool
6
top
since
5
3
mean
12
rack
job
2
first
vs1
used
SDN
reduce
New
flow
MapReduce
future
centers
hosts
software
three
enabled
systems
resources

- **MD5 Hash:** d5d744982be2f62da9972a19f2c0895f

Agenda

- Problem Statement
- Motivation
- Architecture Description
- Use Case: Hadoop
- Experiments and Results
- Conclusion

Problem Statement

Large Datacenters

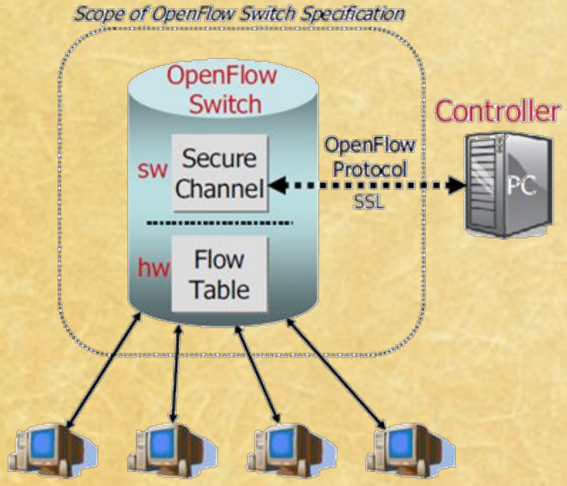
hadoop

OpenMP

LexisNexis

HPCC Systems

Many Applications



Potential Enhancements

Limited Budgets

Motivation

- Cheap (or free as in beer)
- Reprogrammable
- Representative
- Physically accessible
- Administrative privileges
- Ample Network Connections

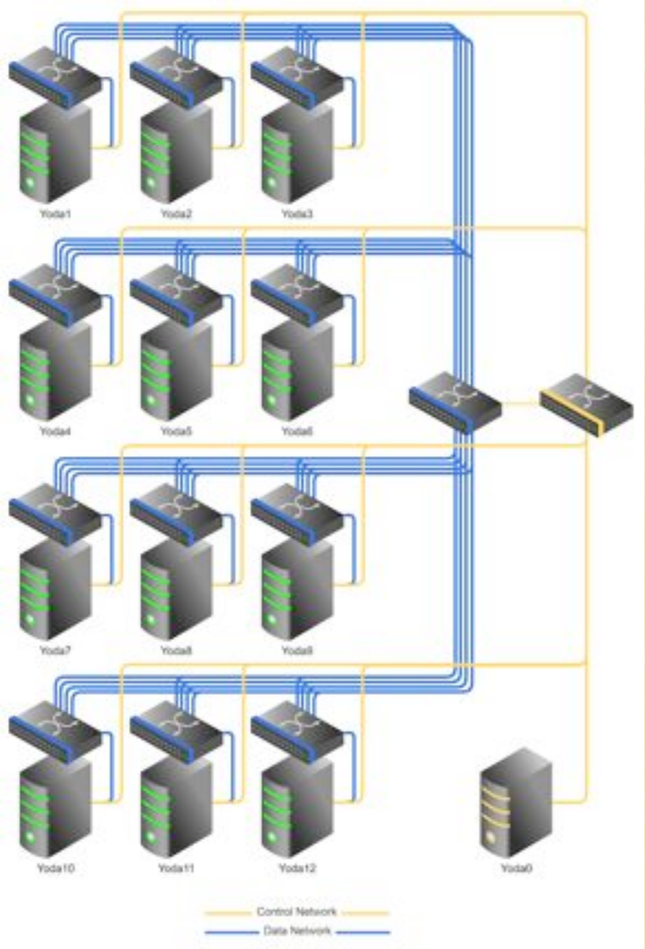
Yoda* Cluster

*The placement of these four letters in this specific order does not constitute an endorsement of any little green characters owned by LucasFilm and Disney. This is merely an term chosen to describe our little powerful cluster used by our lab whose informal name is DAGOBA (DAta GOing Beyond Analytics).

Architecture (hardware)

12 Client Workstations (with virtual switch)

Additional LAN ports



<u>Networks:</u>
control
access
data (reprogramable)

2 SDN switches

Primary Server

Architecture (software)

Open Source

ubuntu

OPEN VSWITCH

An Open Virtual Switch

OpenFlow

Floodlight

python™

Custom

FORCE

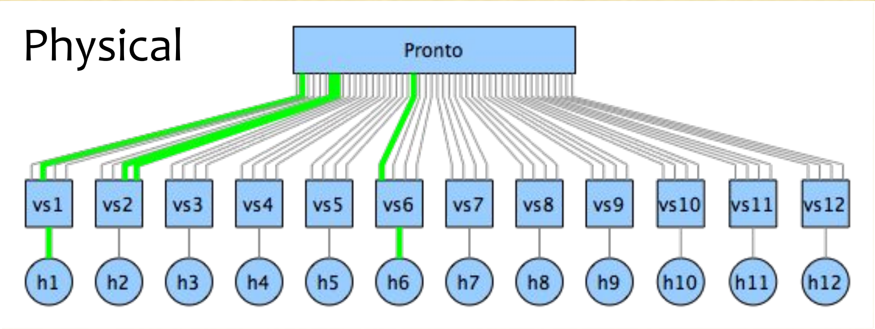
Flow Optimized Route Configuration Engine

Powered by NetworkX

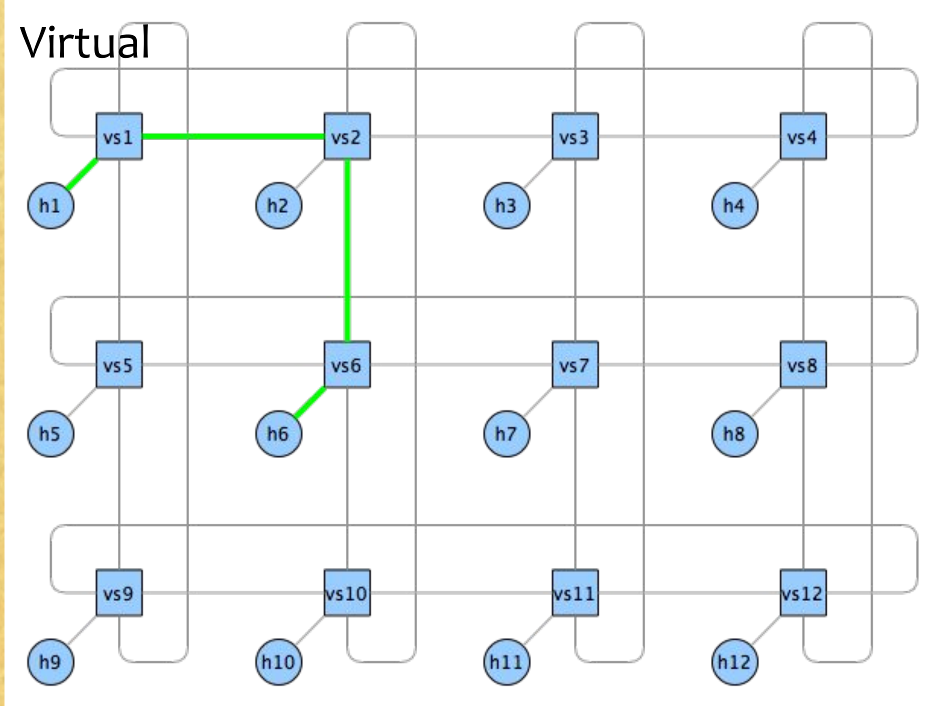
LSS

Link Sharing Score

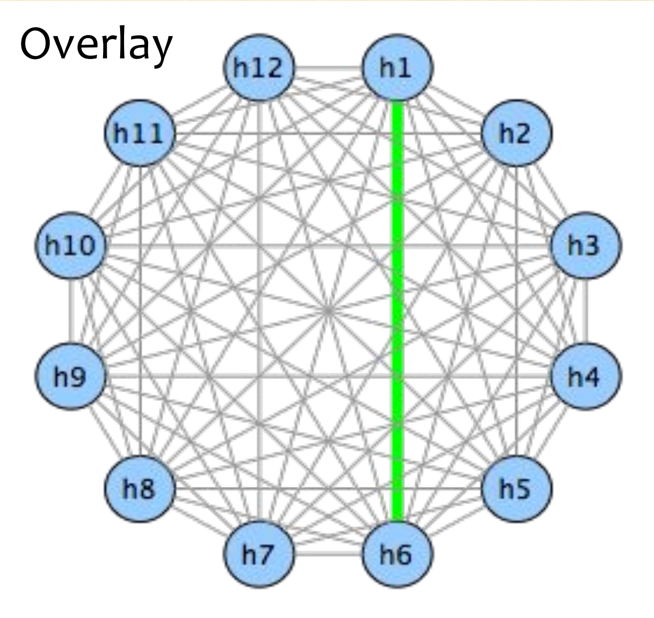
Virtual Topology Engine



Physical

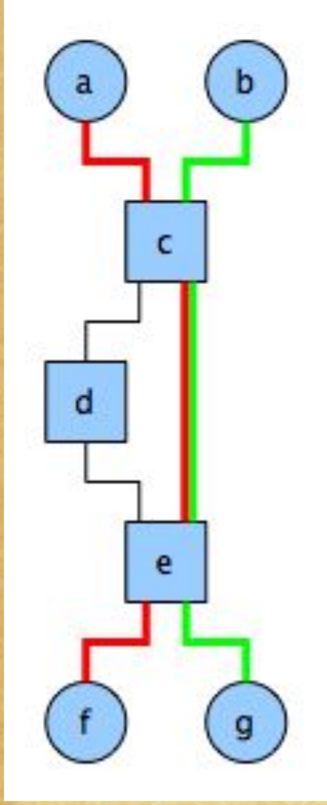


Virtual

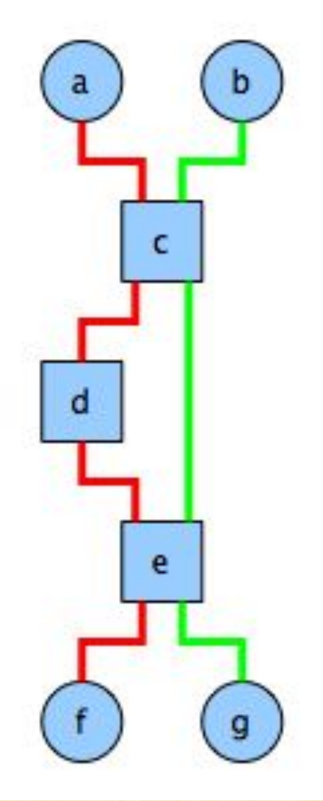


Overlay

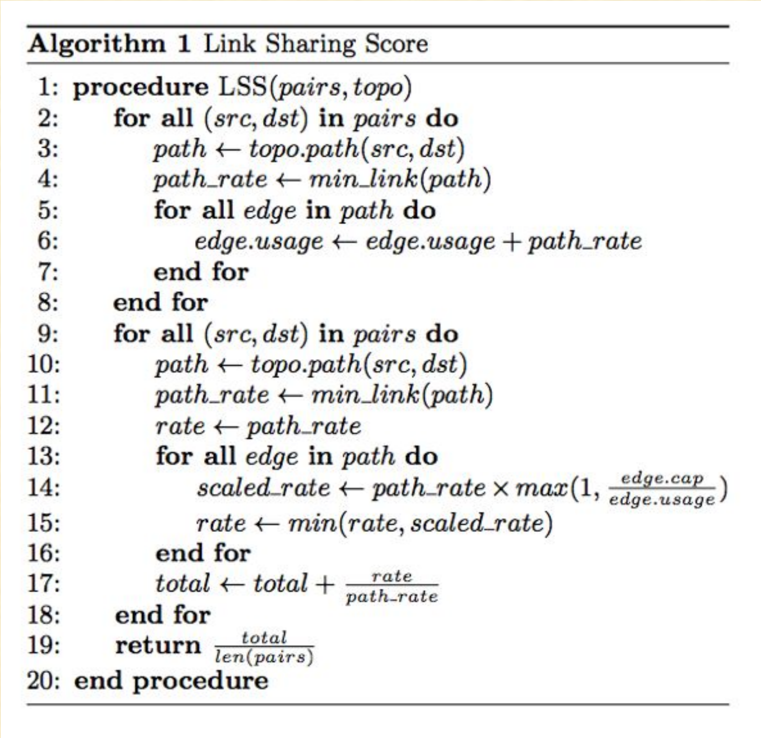
Congestion Matrix



Higher



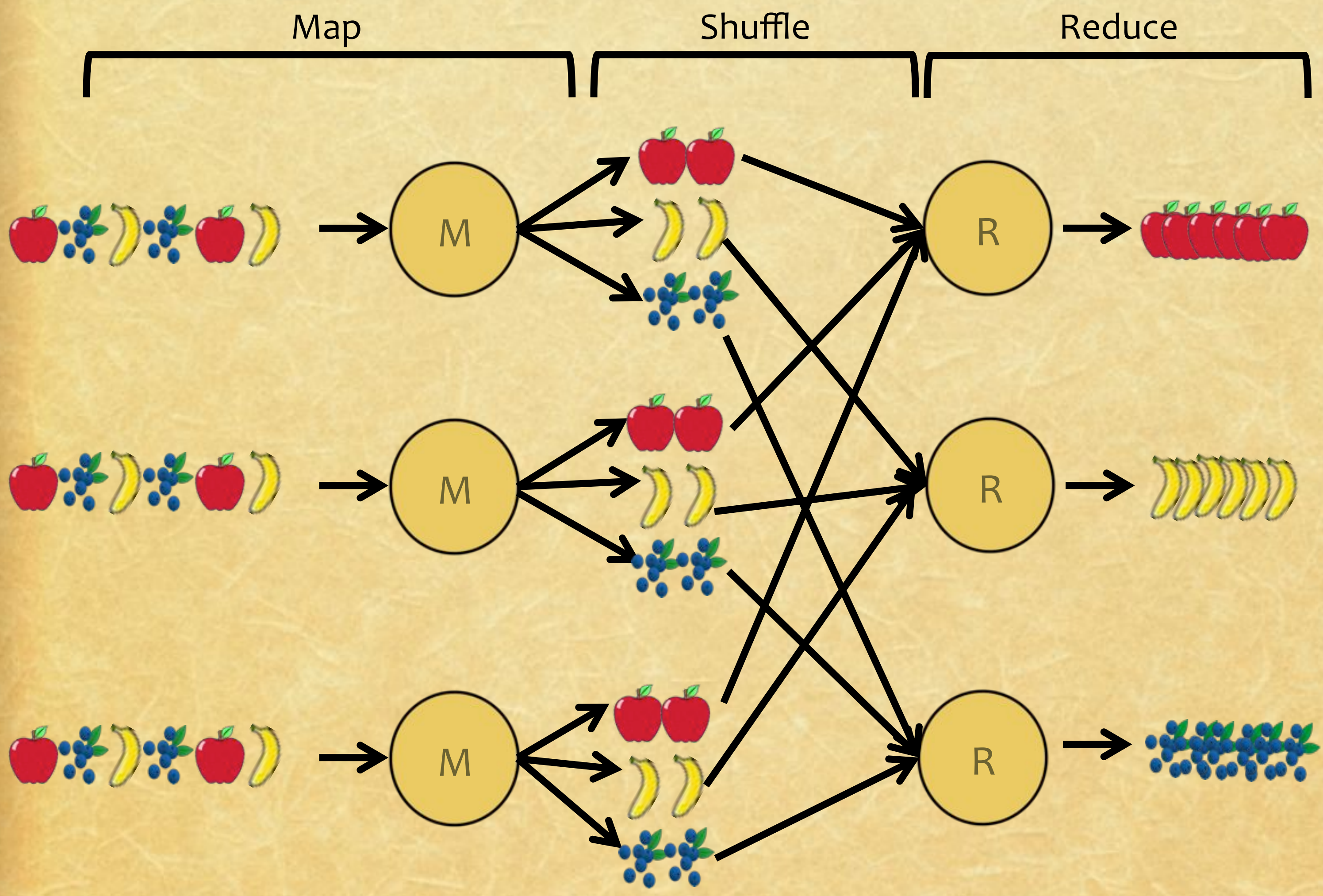
Lower



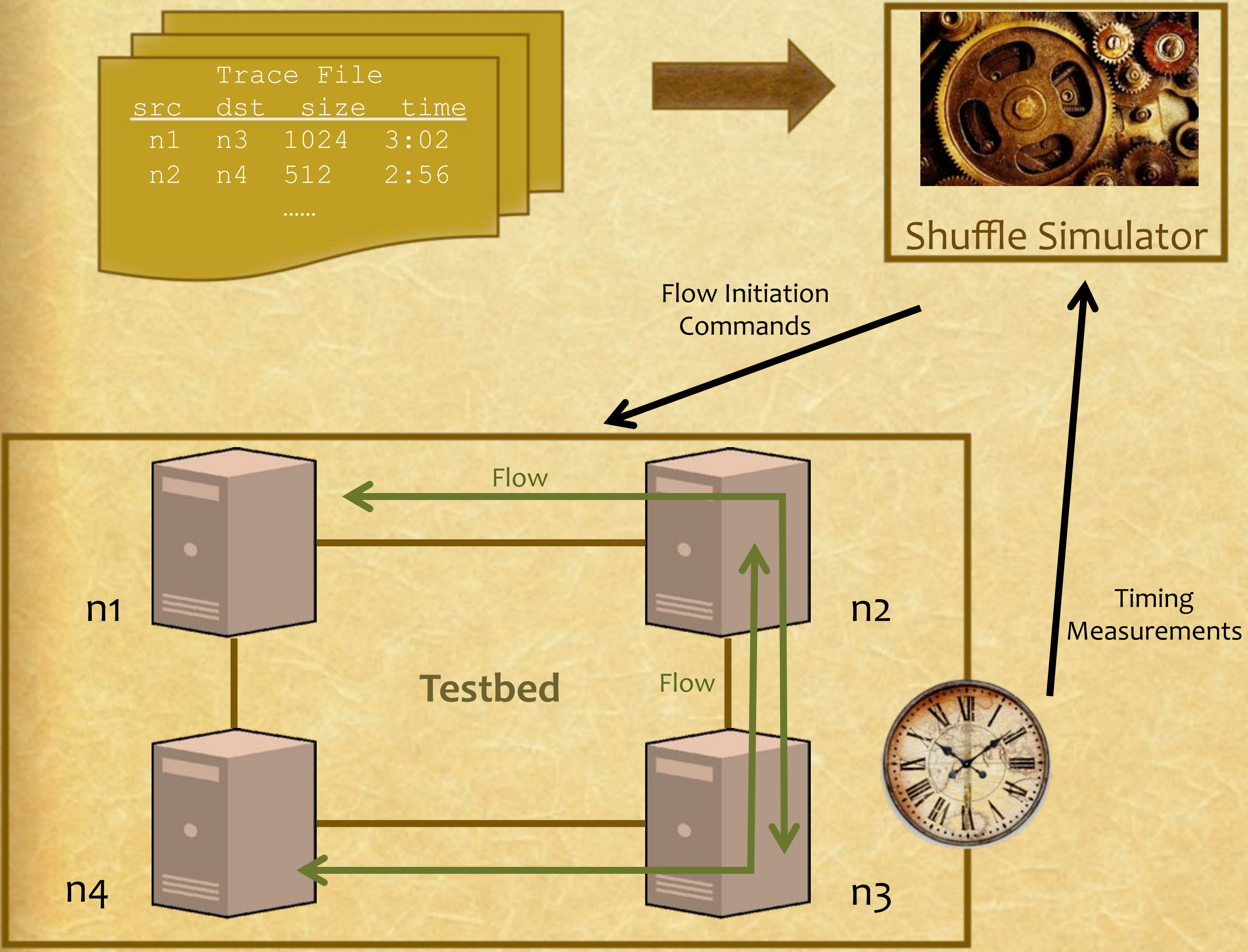
Algorithm 1 Link Sharing Score

```
procedure LSS($pairs, topo$)
    for all ($src, dst$) in $pairs$ do
        $path \leftarrow topo.path(src, dst)$
        $path\_rate \leftarrow min\_link(path)$
        for all $edge$ in $path$ do
            $edge.usage \leftarrow edge.usage + path\_rate$
        end for
    end for
    for all ($src, dst$) in $pairs$ do
        $path \leftarrow topo.path(src, dst)$
        $path\_rate \leftarrow min\_link(path)$
        $rate \leftarrow path\_rate$
        for all $edge$ in $path$ do
            $scaled\_rate \leftarrow path\_rate \times max(1, \frac{edge.cap}{edge.usage})$
            $rate \leftarrow min(rate, scaled\_rate)$
        end for
        $total \leftarrow total + \frac{rate}{path\_rate}$
    end for
    return $\frac{total}{len(pairs)}$
end procedure
```

Use Case: Hadoop



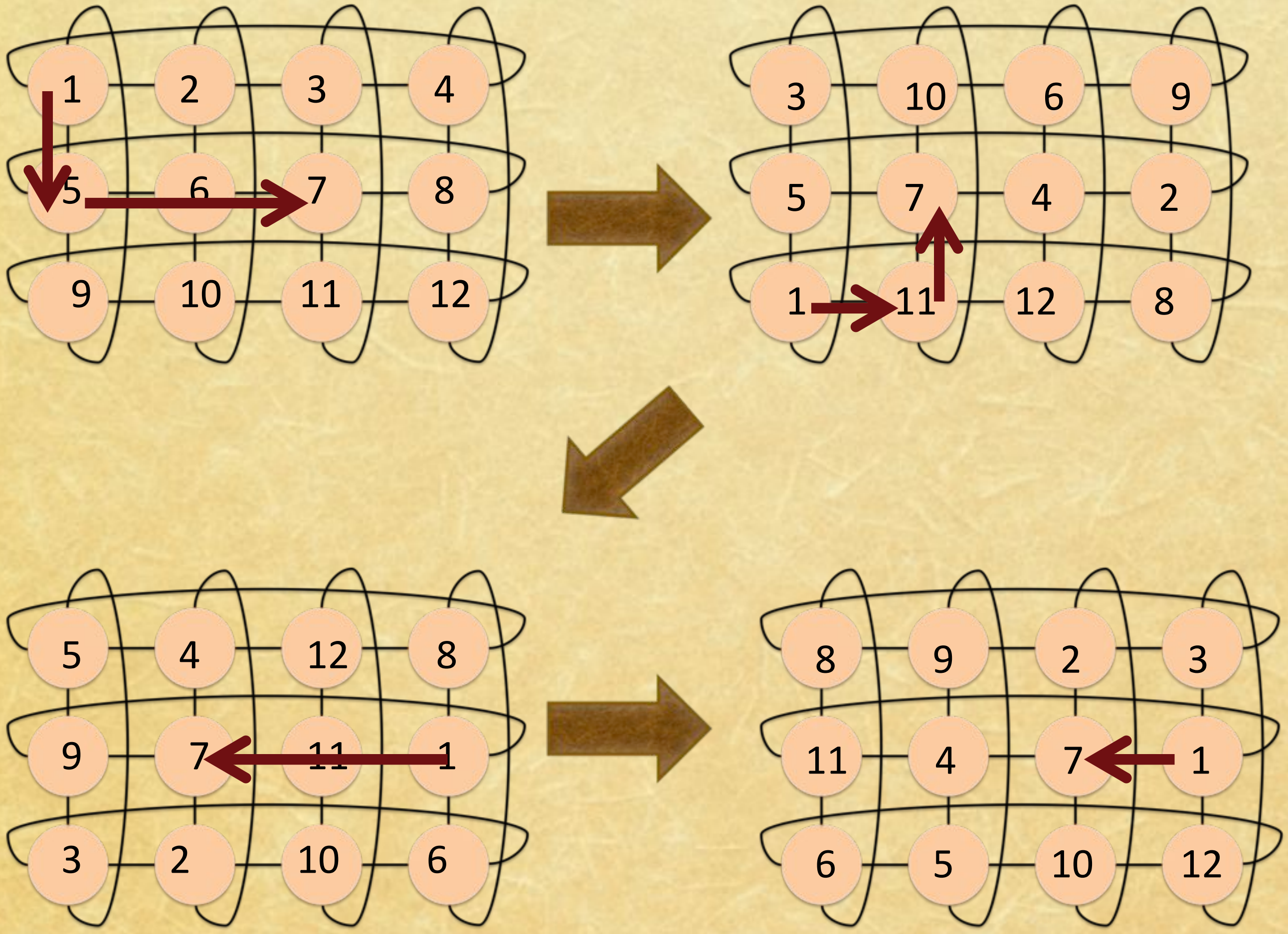
Hadoop Simulator



Trace File

src	dst	size	time
n1	n3	1024	3:02
n2	n4	512	2:56
……			

Experiment on Random Torus



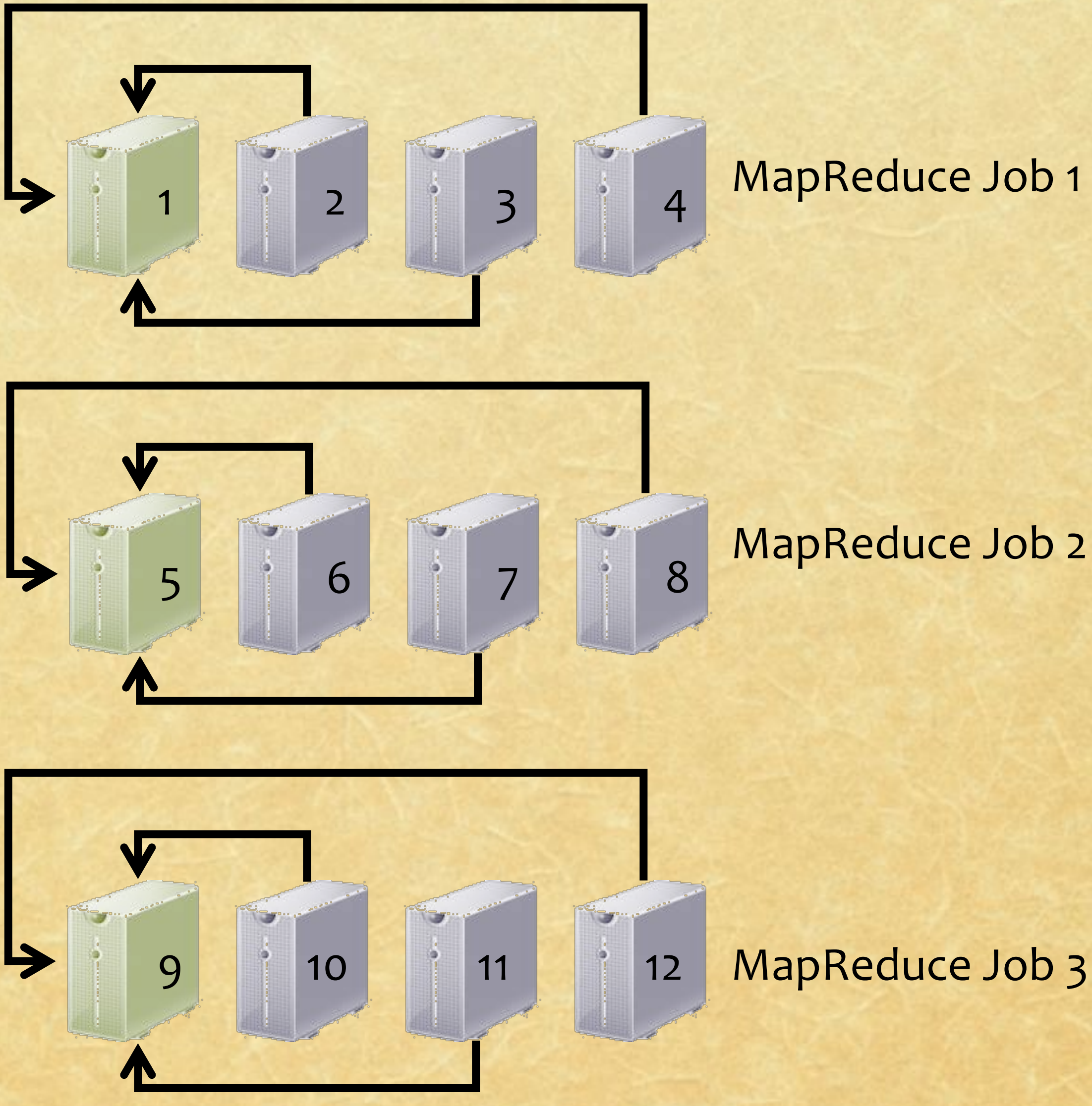
Some topologies place source and destination racks closer in the switching overlay

Shuffle Traffic Scenario

Experiment Design

- 3 simultaneous jobs
- 1 GB data transferred from each map to single reducer
- All flows concurrent
- 1000 runs each under a different placement of nodes in topology; record times, throughput and LSS for each

🟪 Mapper
🟩 Reducer

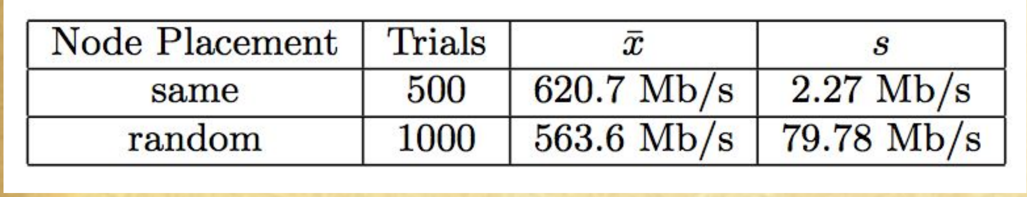


MapReduce Job 1

MapReduce Job 2

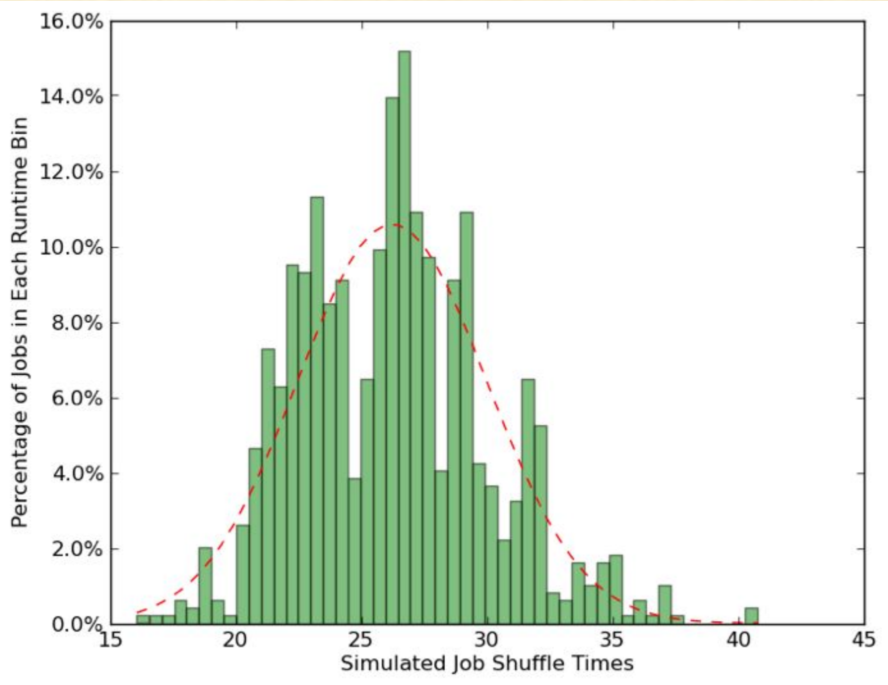
MapReduce Job 3

Shuffle Simulation Results



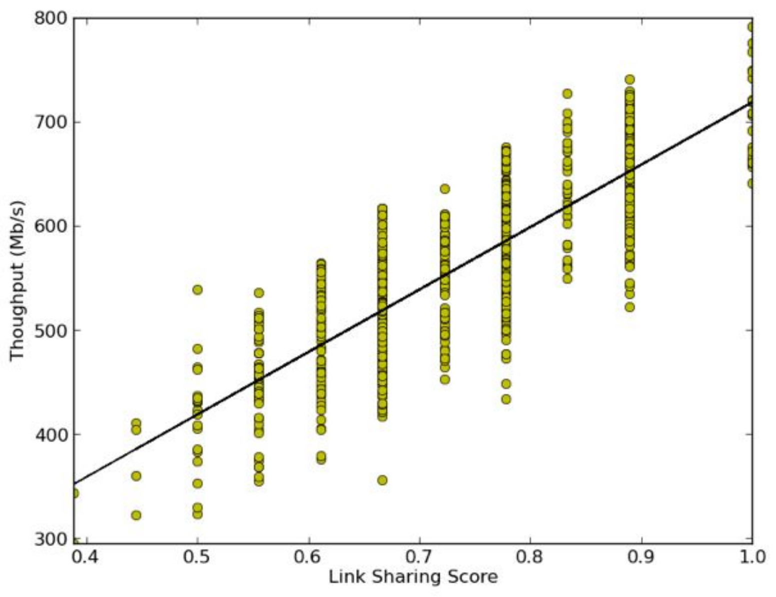
Node Placement	Trials	$\bar{x}$	s
same	500	620.7 Mb/s	2.27 Mb/s
random	1000	563.6 Mb/s	79.78 Mb/s

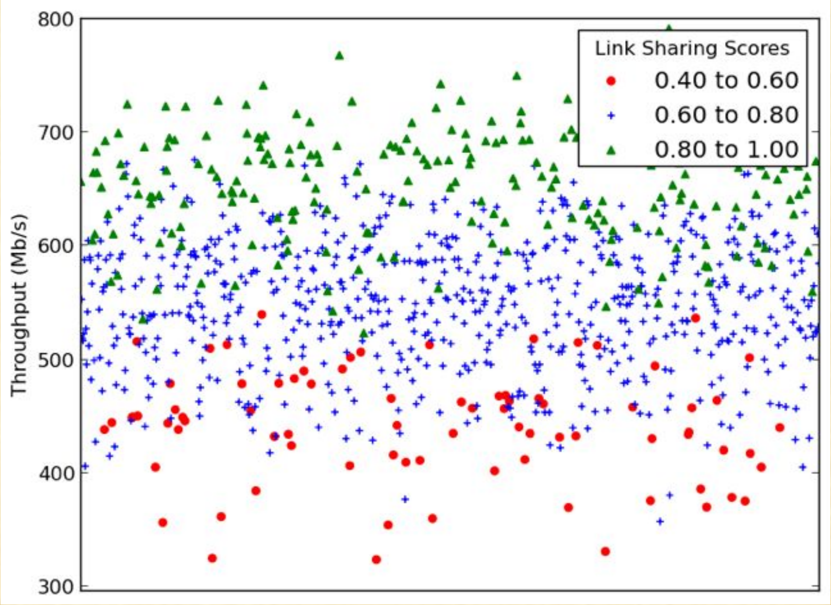
Throughput Mean and Standard Deviation (bigger is better)



Histogram of Shuffle Times (small is better)

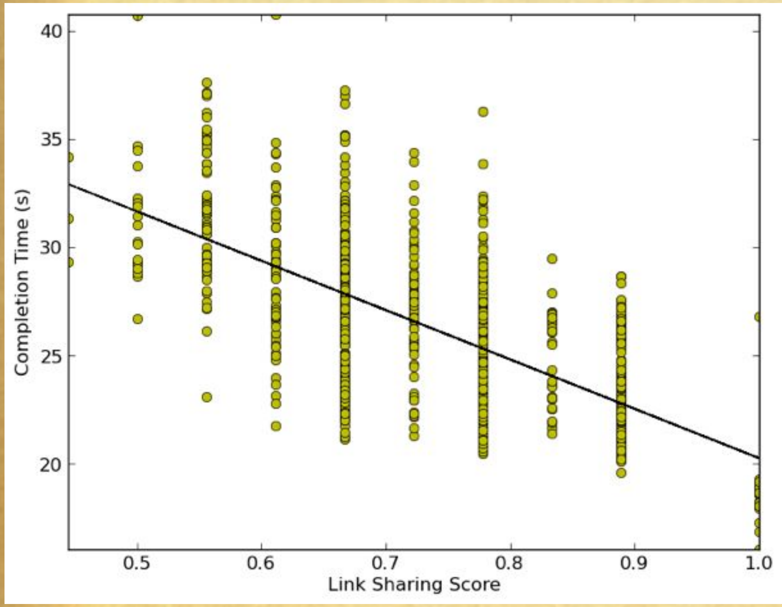
Throughput and LSS

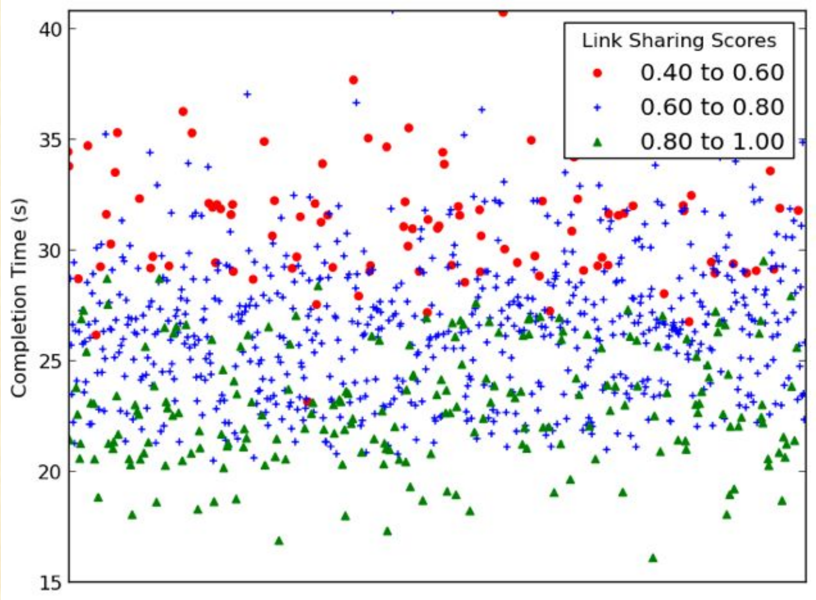




LSS correlates well with Throughput as is expected

Shuffle Time and LSS





LSS correlates well with Shuffle time as is expected.

The difference between optimal and suboptimal configurations can have significant effect on the overall time taken

Conclusion

- We have presented the Flow Optimized Route Configuration Engine (FORCE), a datacenter testbed emulator with a programmable interconnection controlled by an SDN controller. The FORCE allows researchers to get an early indication of the worthiness of data center topology hypothesis.
- These experimental results come without the cost in time or funding of building production level data centers.
- Additionally, the system features a Virtual Topology Engine, a Flow Network Evaluation System, and a Hadoop shuffle traffic simulator.
- We have presented initial experimental results to suggest that datacenter topology, specifically placement within a 4x3 2-D torus network, can impact the time to shuffle intermediate results from a MapReduce job.
- In the future, we plan to build a complete Hadoop traffic simulator, upgrade the emulated rack workstations, and develop a system that will provide execution time adaptability and maneuverability of datacenter topology to steer away from worst case scenarios. We also plan to deploy and validate our hypotheses in production data centers with SDN capabilities.