Cloud Networking

Network Virtualization

Case Study: VL2

P. Brighten Godfrey and Ankit Singla

Department of Computer Science

ILLINOIS

UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

Key Needs

Agility

Location independent addressing

Performance uniformity

Security

Network semantics

Case Study

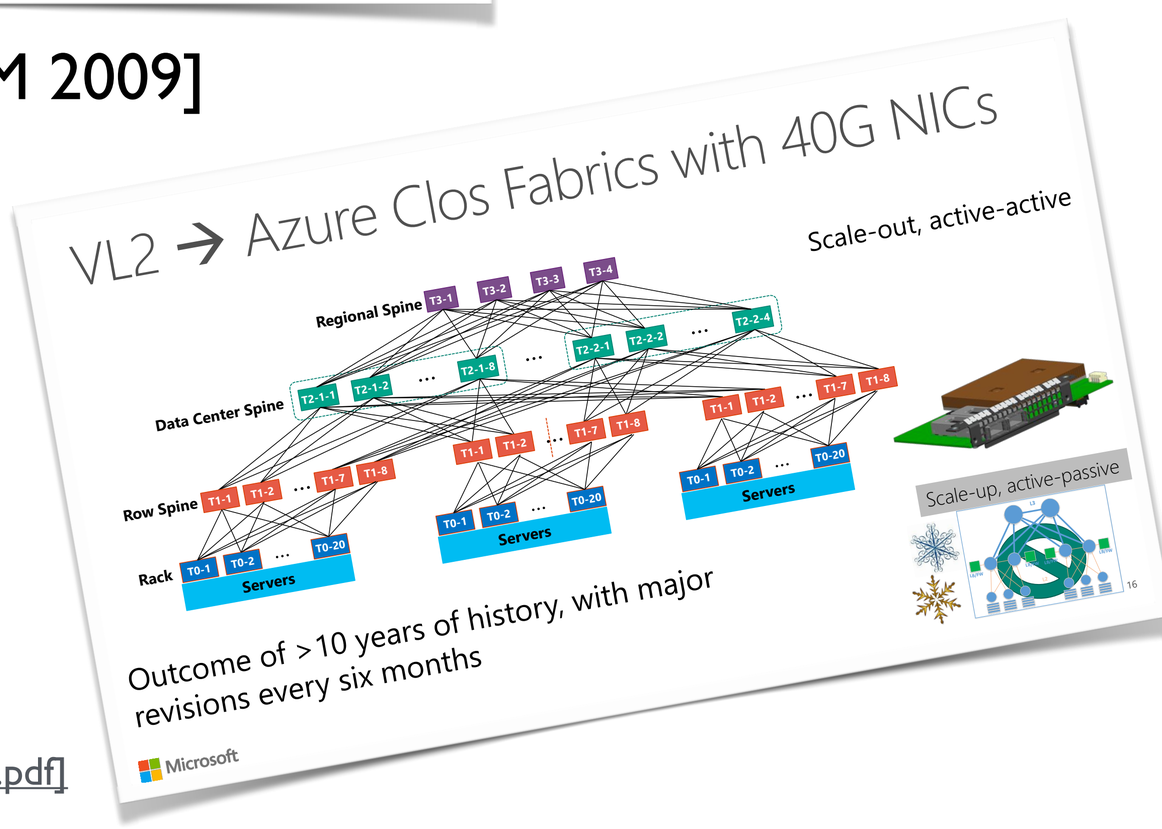
VL2: A Scalable and Flexible Data Center Network

Albert Greenberg, James R. Hamilton, Navendu Jain
Srikanth Kandula, Changhoon Kim, Parantap Lahiri
David A. Maltz, Parveen Patel, Sudipta Sengupta

Microsoft Research

[ACM SIGCOMM 2009]

Influenced architecture of Microsoft Azure



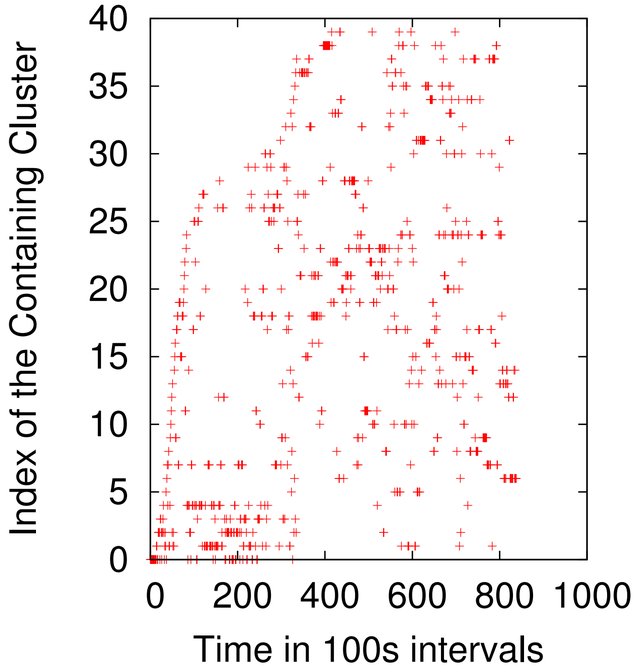
[From Albert Greenberg keynote at SIGCOMM 2015: http://conferences.sigcomm.org/sigcomm/2015/pdf/papers/keynote.pdf]

Motivating Environmental Characteristics

Increasing internal traffic is a bottleneck

- Traffic volume between servers is 4x external traffic

Unpredictable, rapidly-changing traffic matrices (TMs)



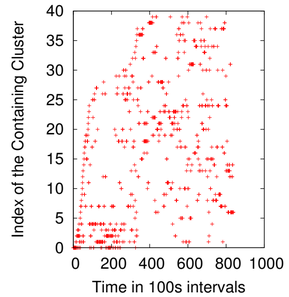
[Greenberg et al.]

Motivating Environmental Characteristics

Increasing internal traffic is a bottleneck

- Traffic volume between servers is 4x external traffic

Unpredictable, rapidly-changing traffic matrices (TMs)



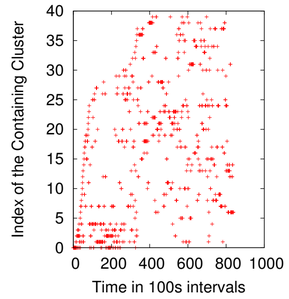
[Greenberg et al.]

Motivating Environmental Characteristics

Increasing internal traffic is a bottleneck

- Traffic volume between servers is 4x external traffic

Unpredictable, rapidly-changing traffic matrices (TMs)



[Greenberg et al.]

Design result: Nonblocking fabric

- High throughput for *any* TM that respects server NIC rates

Motivating Environmental Characteristics

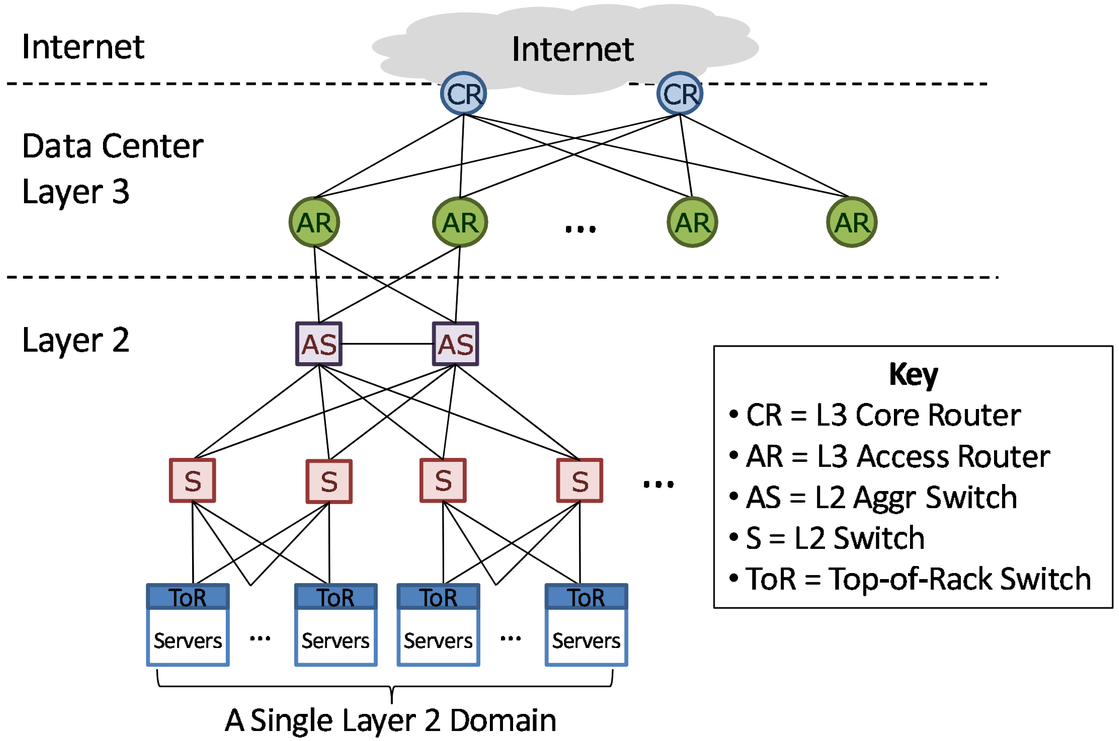
Failure characteristics

- Analyzed 300K alarm tickets, 36M error events
- 0.4% of failures were resolved in over one day
- 0.3% of failures eliminated all redundancy in a device group (e.g. both uplinks)

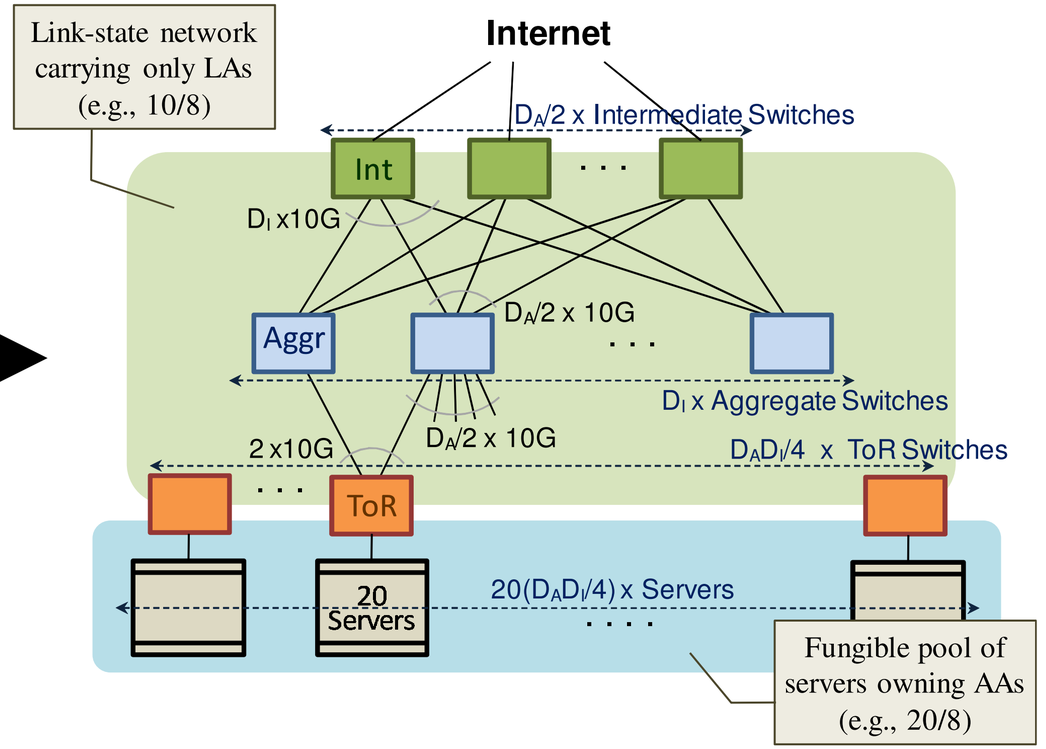
Design result: Clos topology

- “Scale out” instead of “scale up”

VL2 physical topology



Traditional



VL2

[Figures from Greenberg et al.]

Routing in VL2

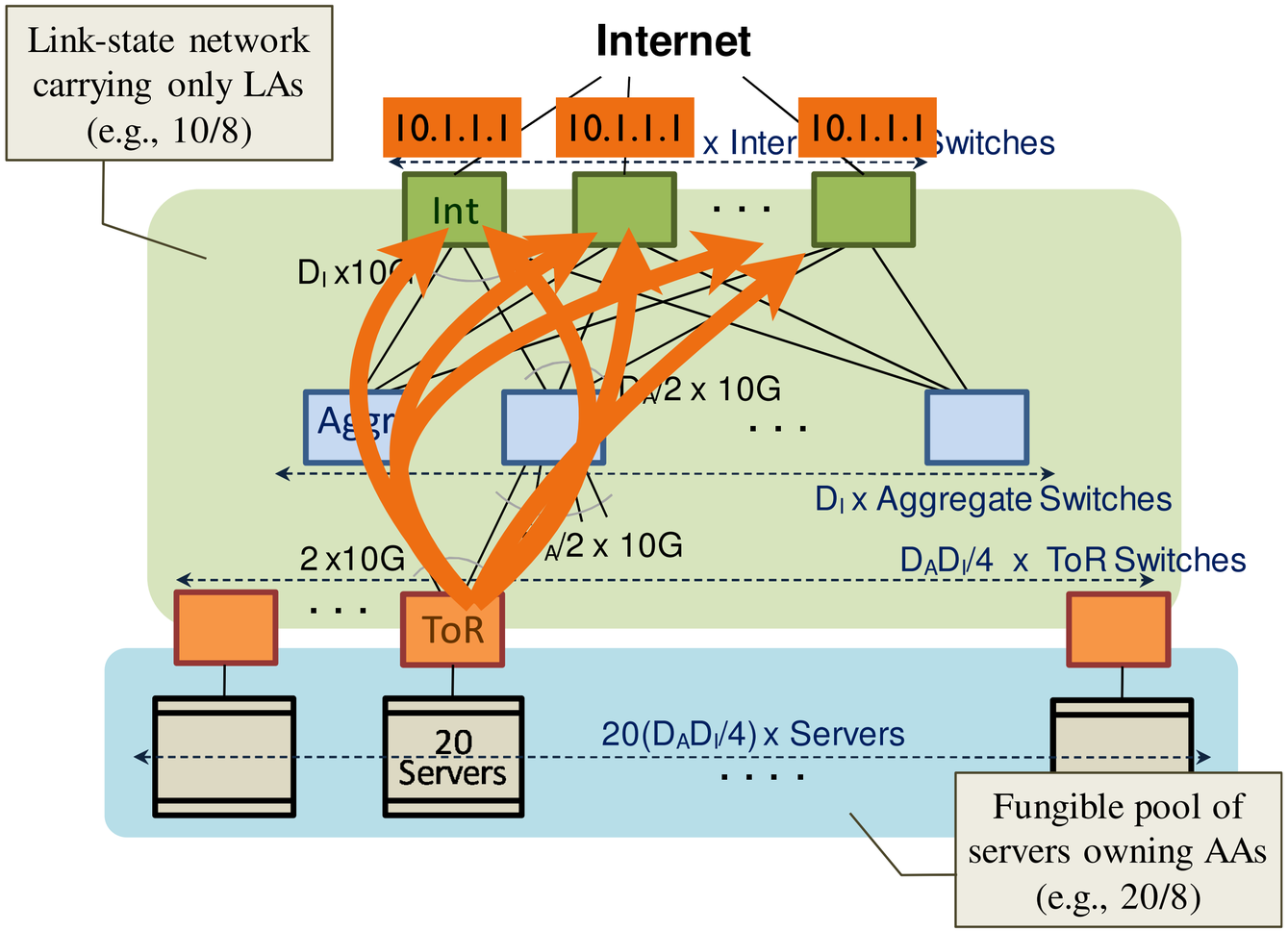
Unpredictable traffic

- Difficult to adapt

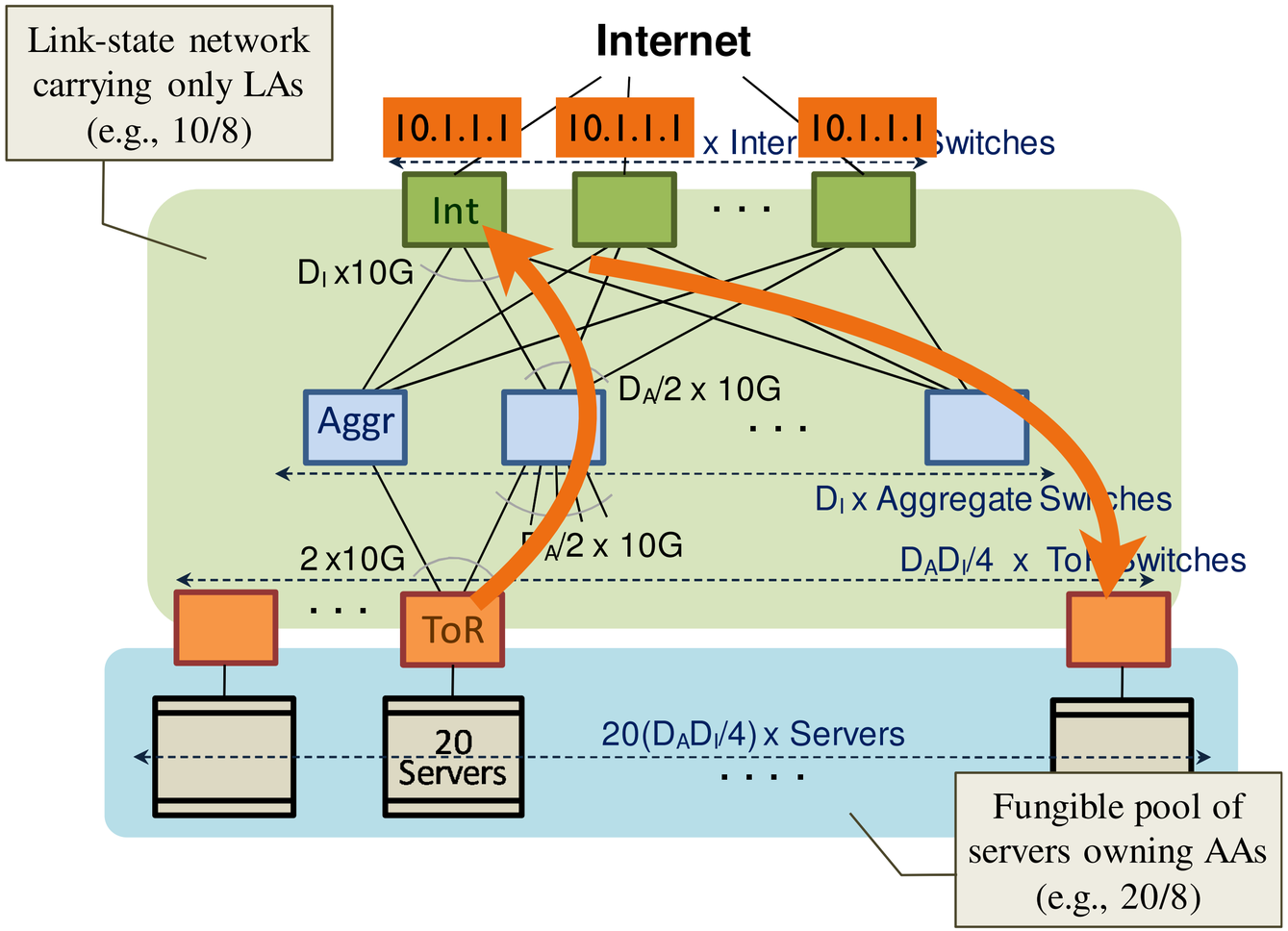
Design result: "Valiant Load Balancing"

- Route traffic independent of current traffic matrix
- Spreads arbitrary traffic pattern so it's uniform among top layer switches

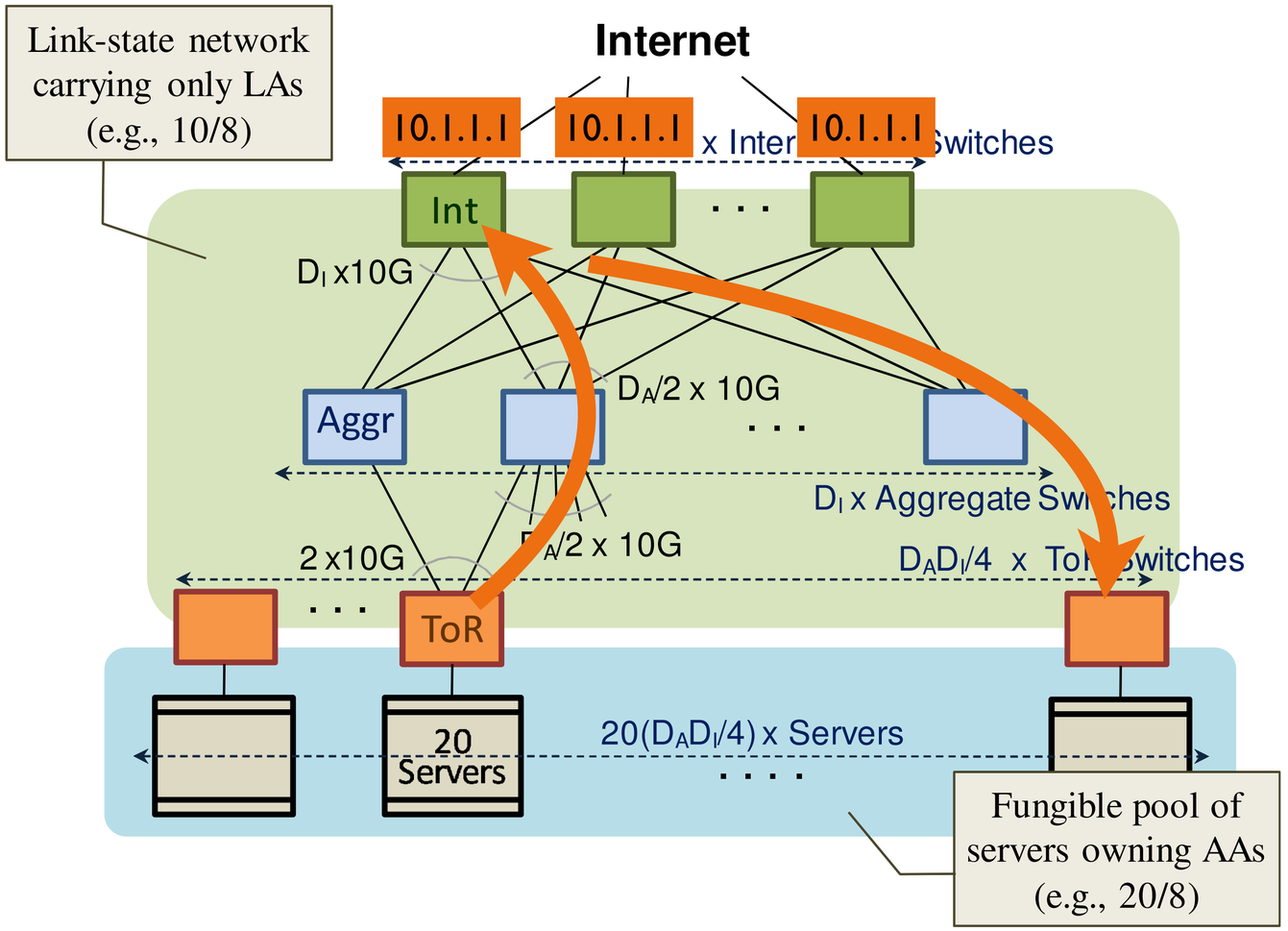
Routing Implementation



Routing Implementation



Routing Implementation



Similar effect to ECMP to each rack

Smaller forwarding tables at most switches

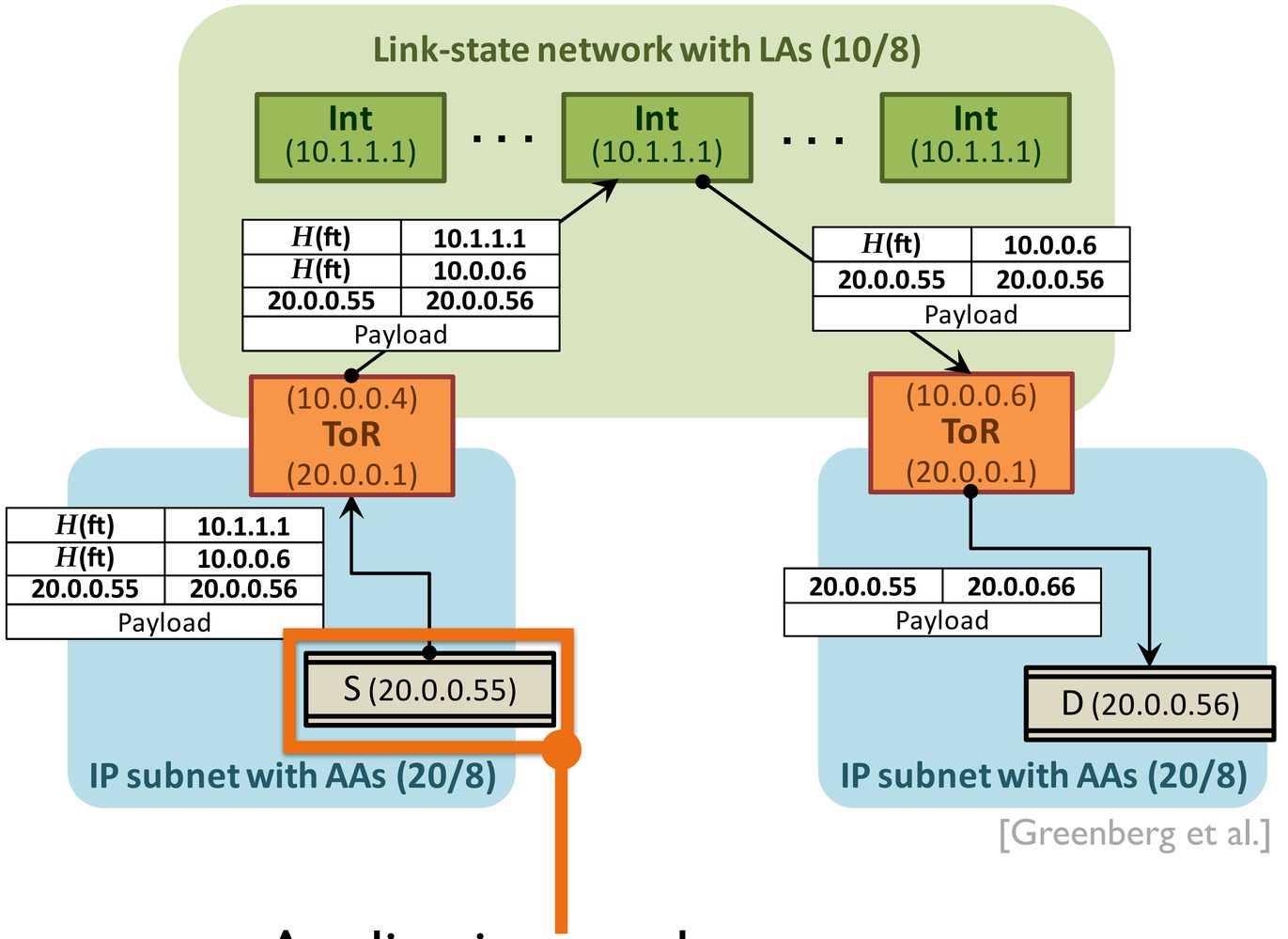
Virtualization

"All problems in computer science can be solved by another level of indirection."

– *David Wheeler*

App / Tenant layer	• Application Addresses (AAs): Location independent • Illusion of a single big Layer 2 switch connecting the app
Virtualization layer	• Directory server: Maintain AA to LA mapping • Server agent: Query server, wrap AAs in outer LA header
Physical network layer	• Locator Addresses (LAs): Tied to topology, used to route • Layer 3 routing via OSPF

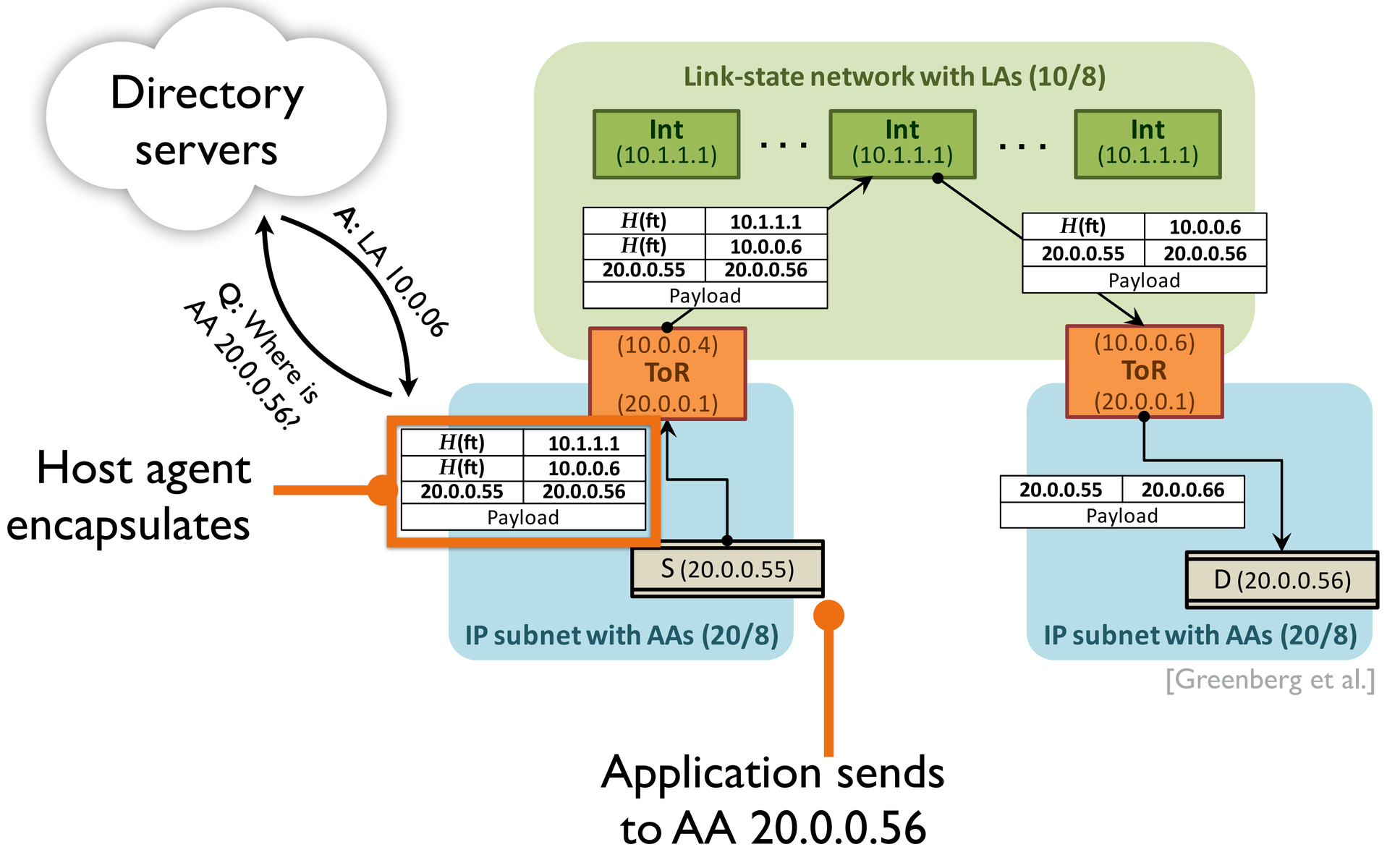
End-to-end example



[Greenberg et al.]

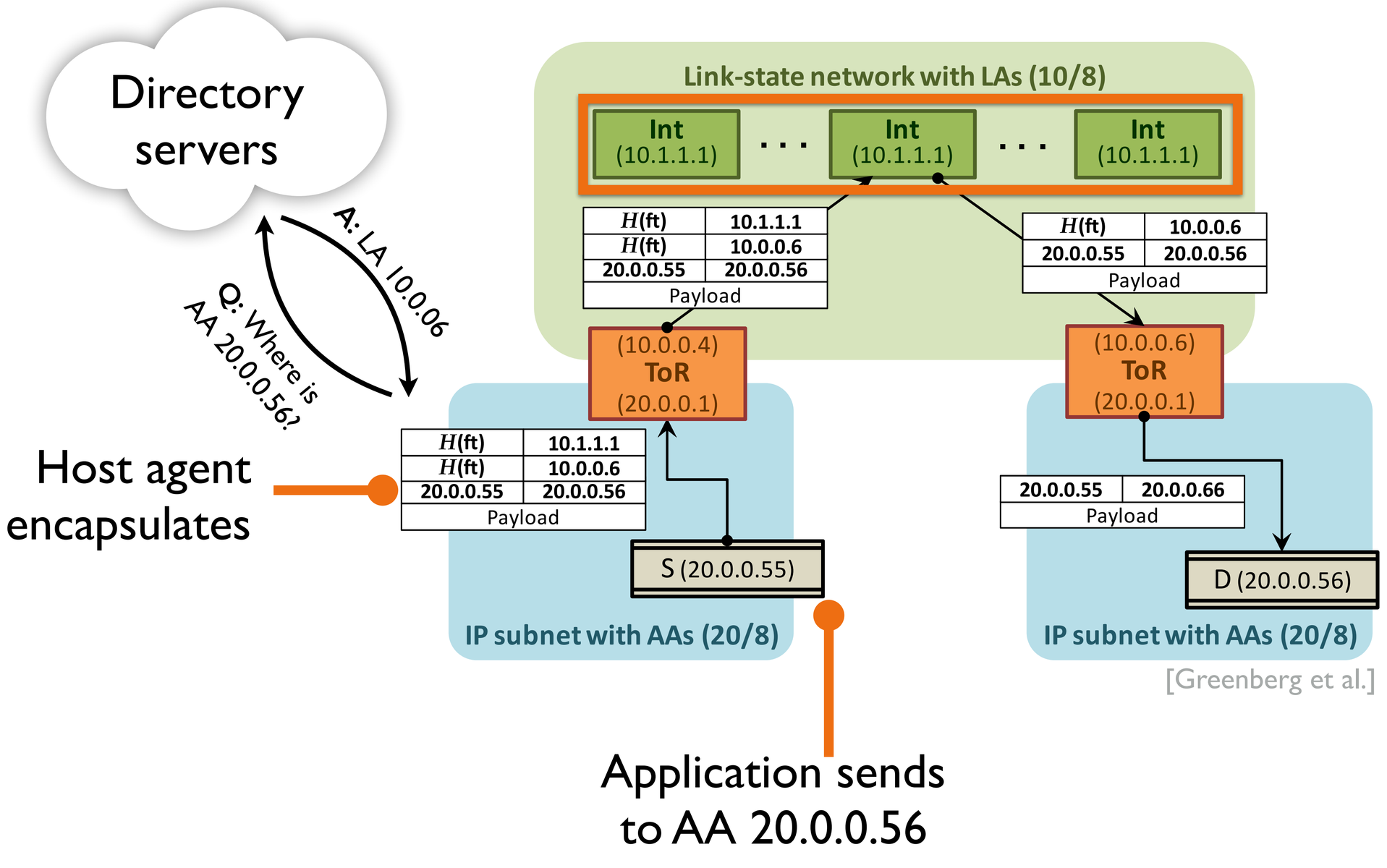
Application sends to AA 20.0.0.56

End-to-end example



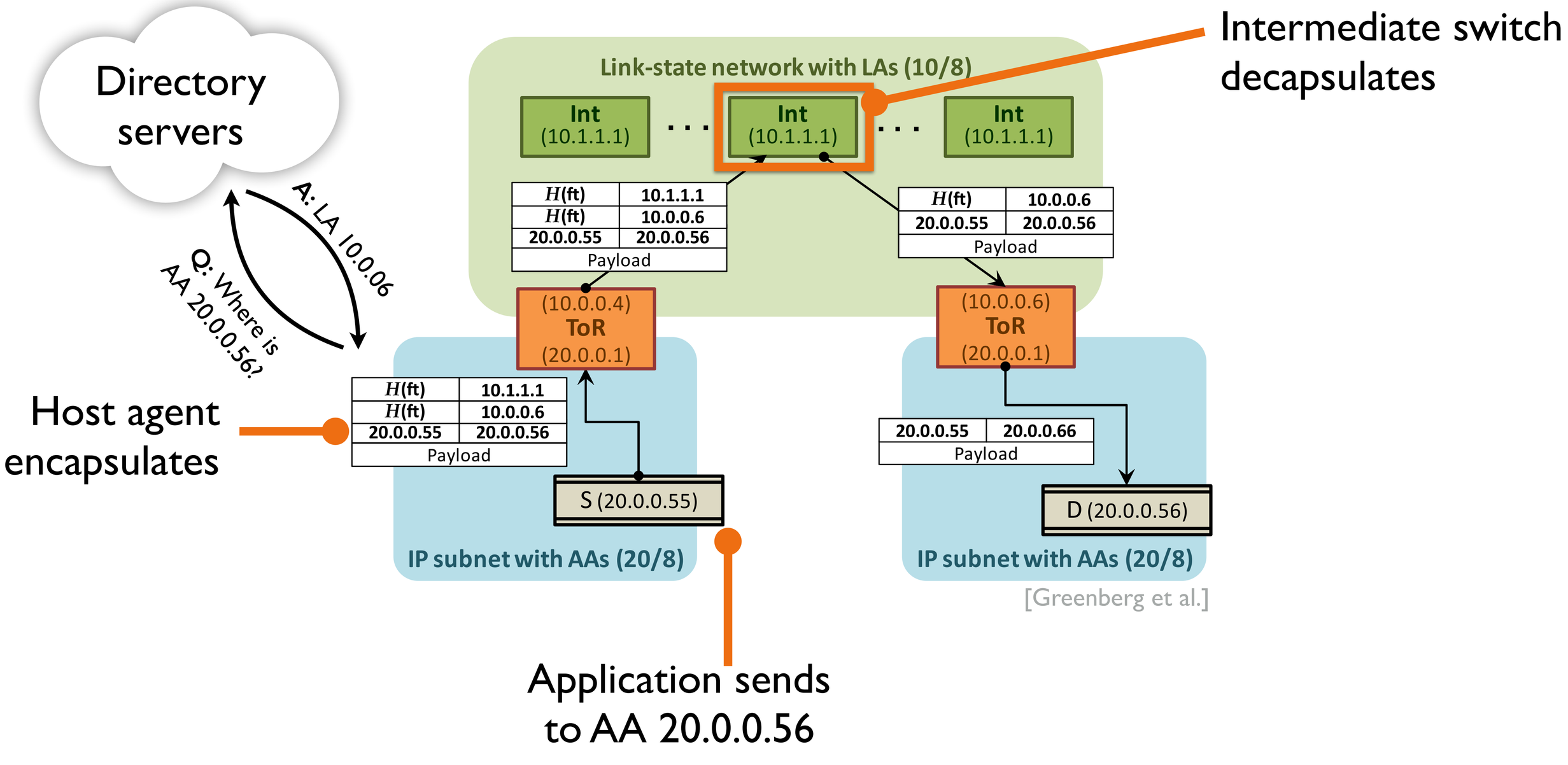
[Greenberg et al.]

End-to-end example



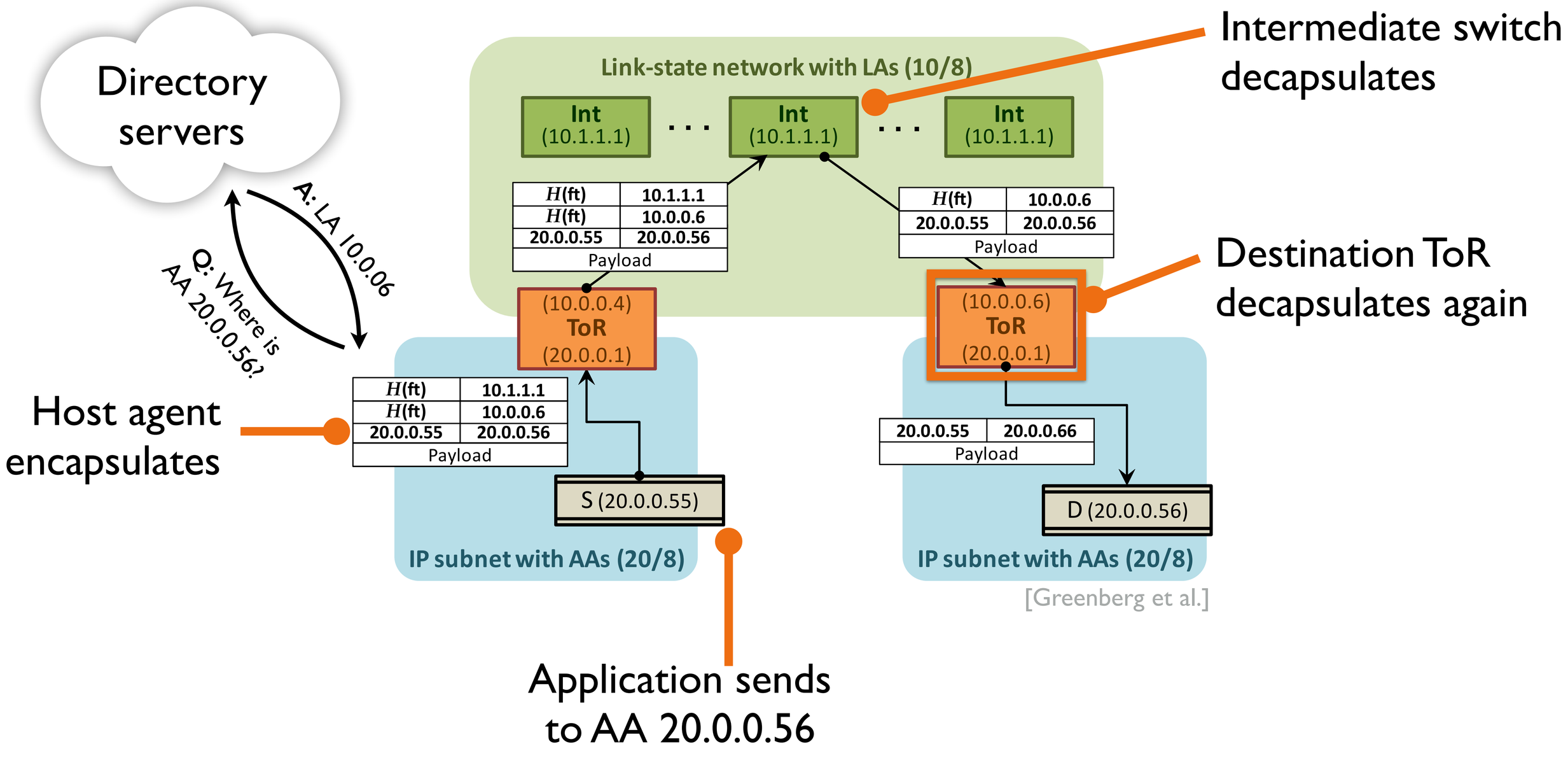
[Greenberg et al.]

End-to-end example



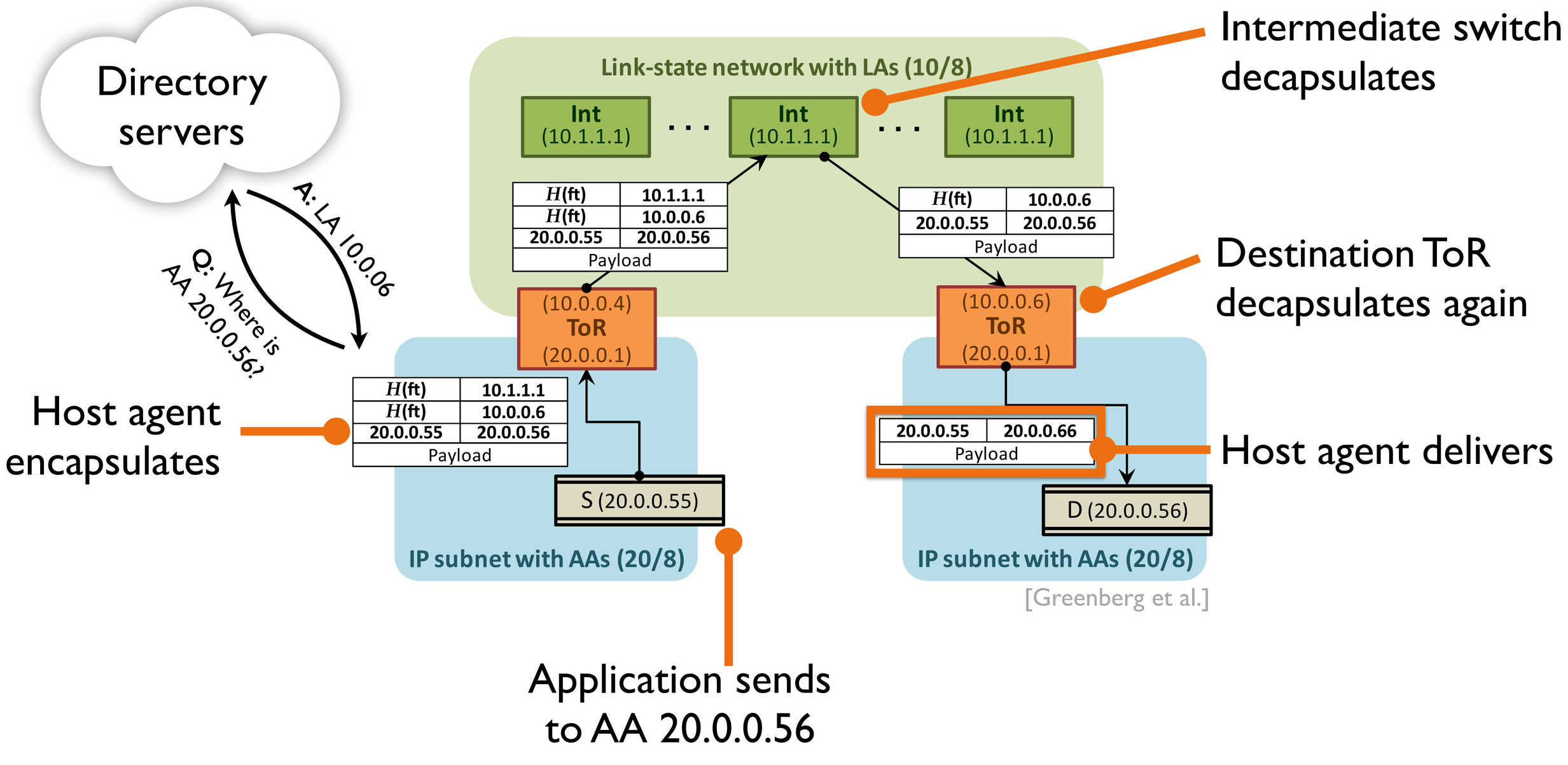
[Greenberg et al.]

End-to-end example



[Greenberg et al.]

End-to-end example



[Greenberg et al.]

Did we achieve agility?

Location independent addressing

- AAs are location independent

L2 network semantics

- Agent intercepts and handles L2 broadcast, multicast

- Both of the above require "layer 2.5" shim agent running on host; but, concept transfers to hypervisor-based virtual switch

Did we achieve agility?

Performance uniformity

- Clos network is nonblocking (non-oversubscribed)
- Uniform capacity everywhere
- ECMP provides good (though not perfect) load balancing
- But, performance isolation among tenants depends on TCP backing off to rate destination can receive
- Leaves open the possibility of fast load balancing

Security

- Directory system can allow/deny connections by choosing whether to resolve an AA to a LA
- But, segmentation not explicitly enforced at hosts

Where's the SDN?

Directory servers: Logically centralized control

- Orchestrate application locations
- Control communication policy

Host agents: dynamic "programming" of data path