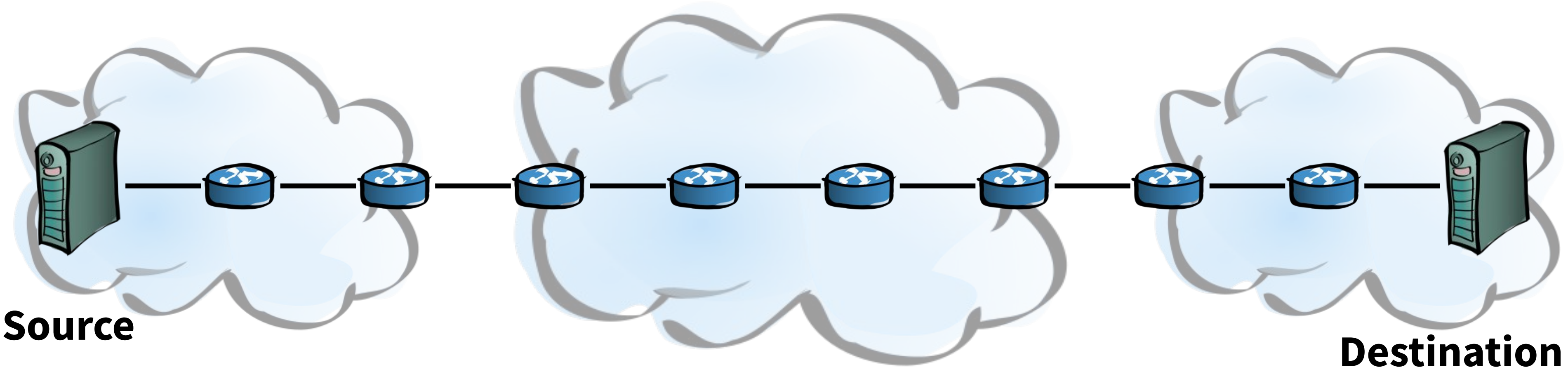
Classification of Load Balancing in the Internet

Rafael Almeida, **Ítalo Cunha**, Darryl Veitch,
Renata Teixeira, Christophe Diot

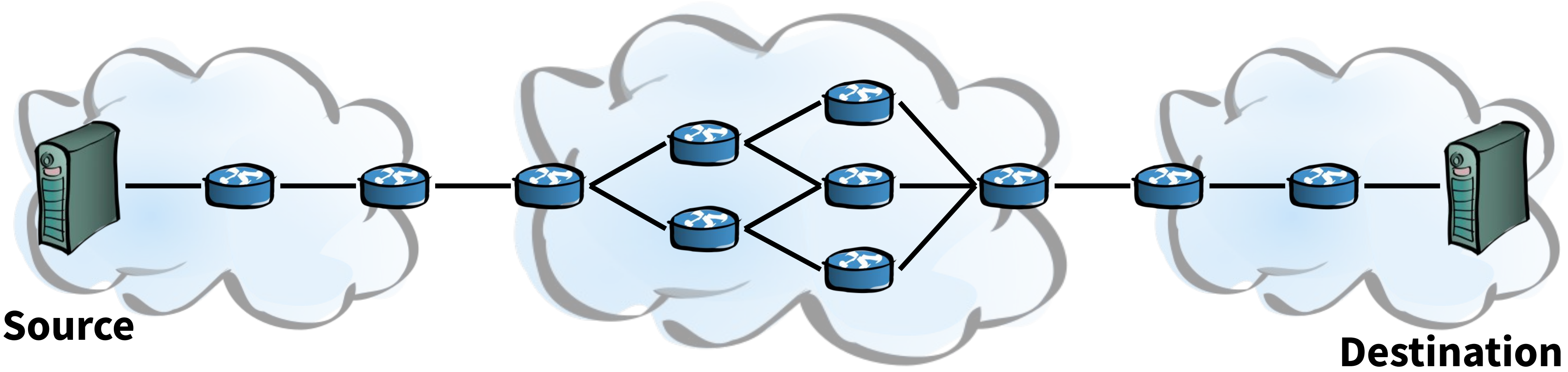
UFMG
UTS
Inria
Google

Traffic Load Balancing in Internet Routes



Routes traverse multiple networks and routers

Traffic Load Balancing in Internet Routes



Most routes traverse multiple branches of routers

Traffic Load Balancing

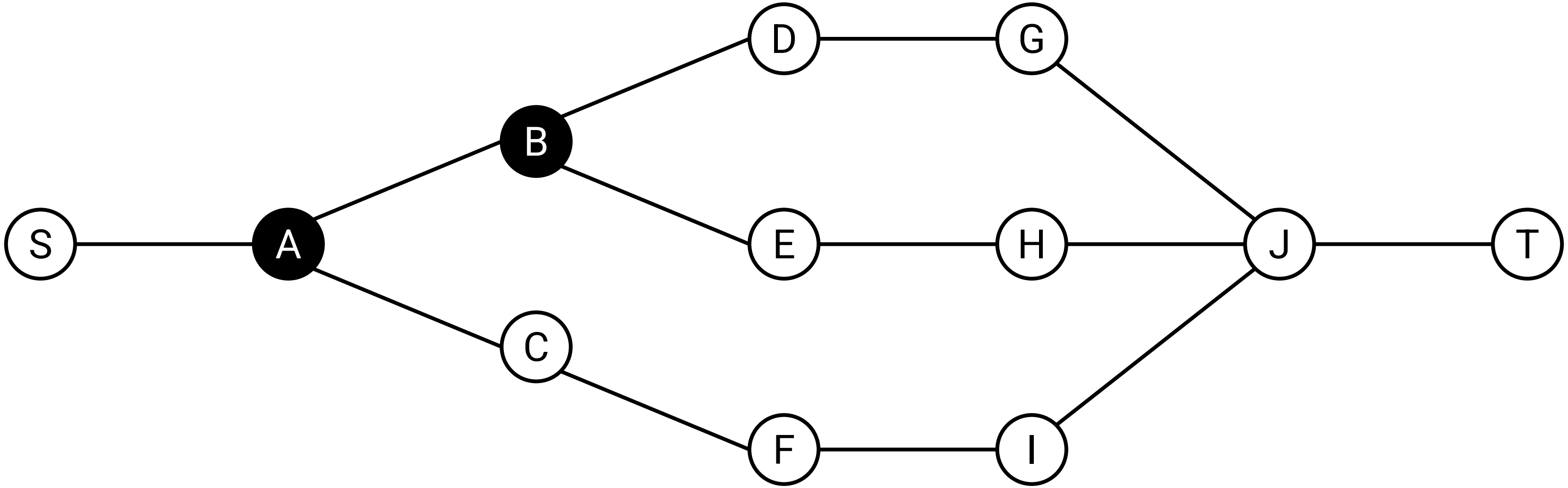
Widely deployed on Internet routes

- Increase bandwidth
- Increase reliability
- Improve link utilization

Important for researchers and operators

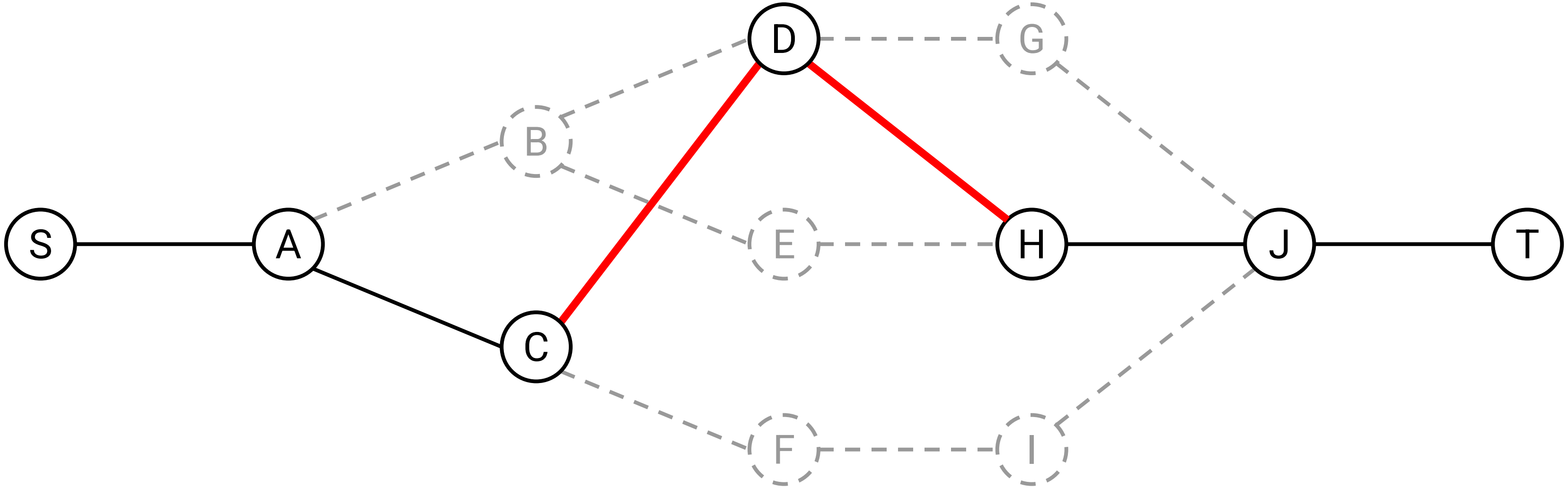
- Impacts Internet characterization and modeling
- May disrupt production traffic

Traceroute on Load Balanced Routes



Routers A and B perform load balancing

Classic Traceroute on Load Balanced Routes



Different probes may take different branches, leading to inference of false links

How Load Balancers Work

TTL	8
Protocol	TCP
Source addr	150.164.4.12
Dest addr	157.240.222.33
Source port	35151
Dest port	80
Length	1442
Checksum	0xaffe

IP packet

How Load Balancers Work

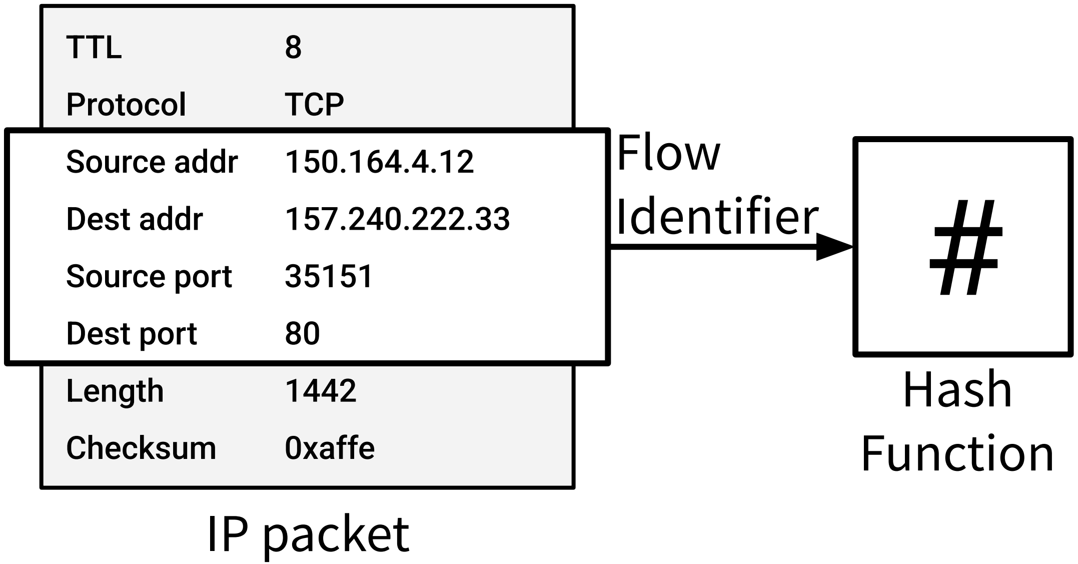
TTL	8
Protocol	TCP
Source addr	150.164.4.12
Dest addr	157.240.222.33
Source port	35151
Dest port	80
Length	1442
Checksum	0xaffe

Flow Identifier

IP packet

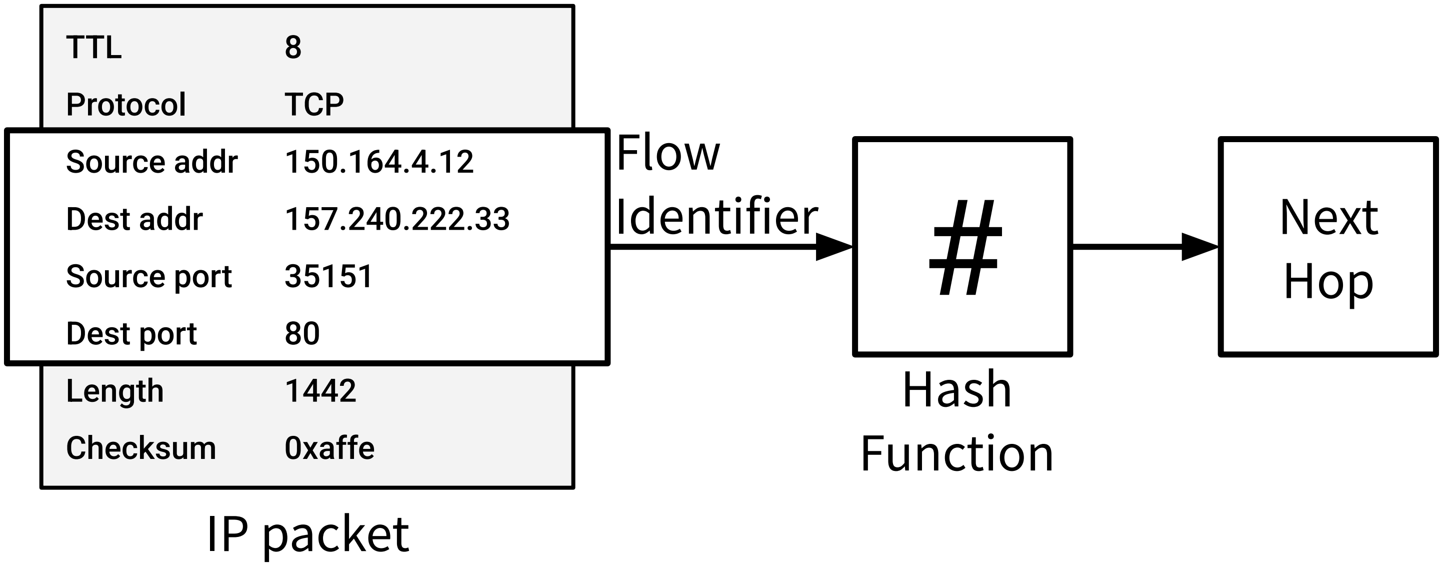
The load balancer extracts a set of bits called the flow identifier from the packet

How Load Balancers Work



The flow identifier is hashed

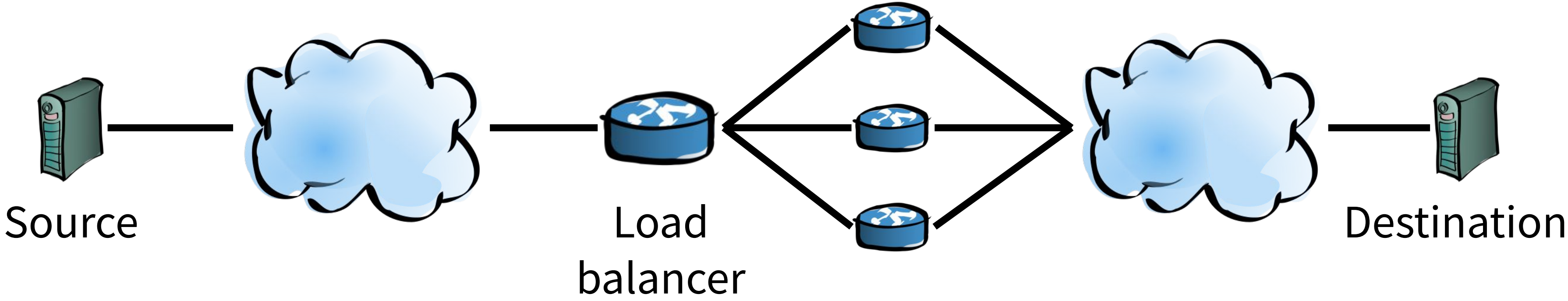
How Load Balancers Work



The flow identifier is hashed and the result defines the next hop for forwarding

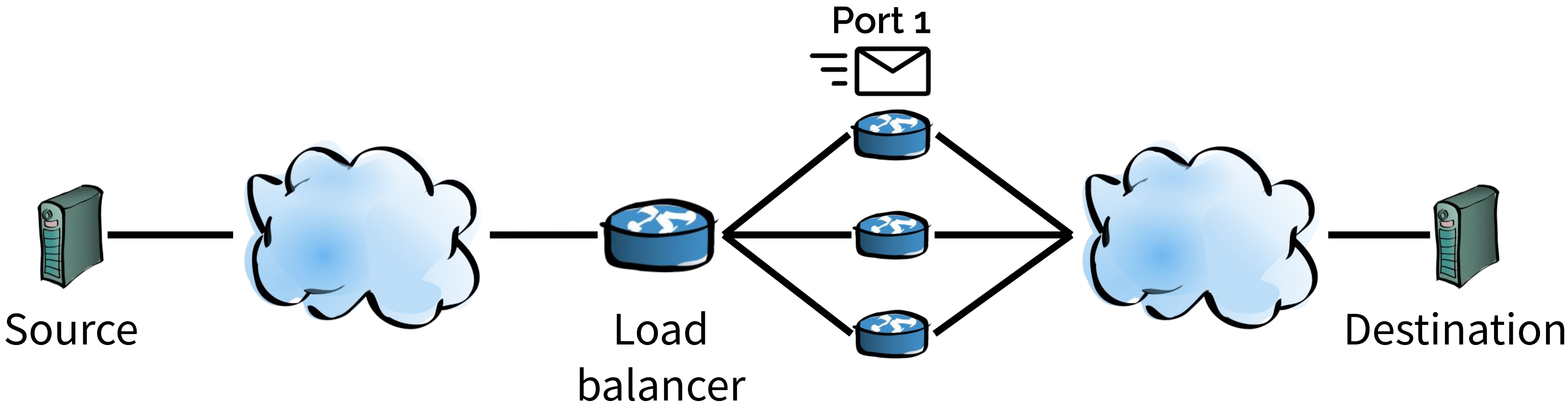
Load Balancers and Hash Domains

Packets with different flow identifiers may take different branches



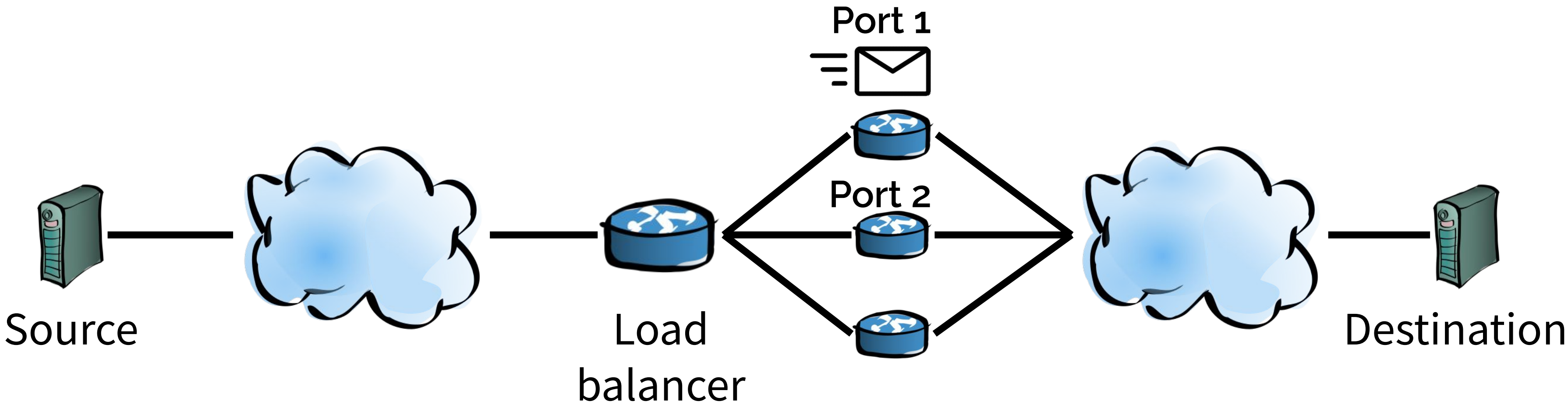
Load Balancers and Hash Domains

Packets with different flow identifiers may take different branches



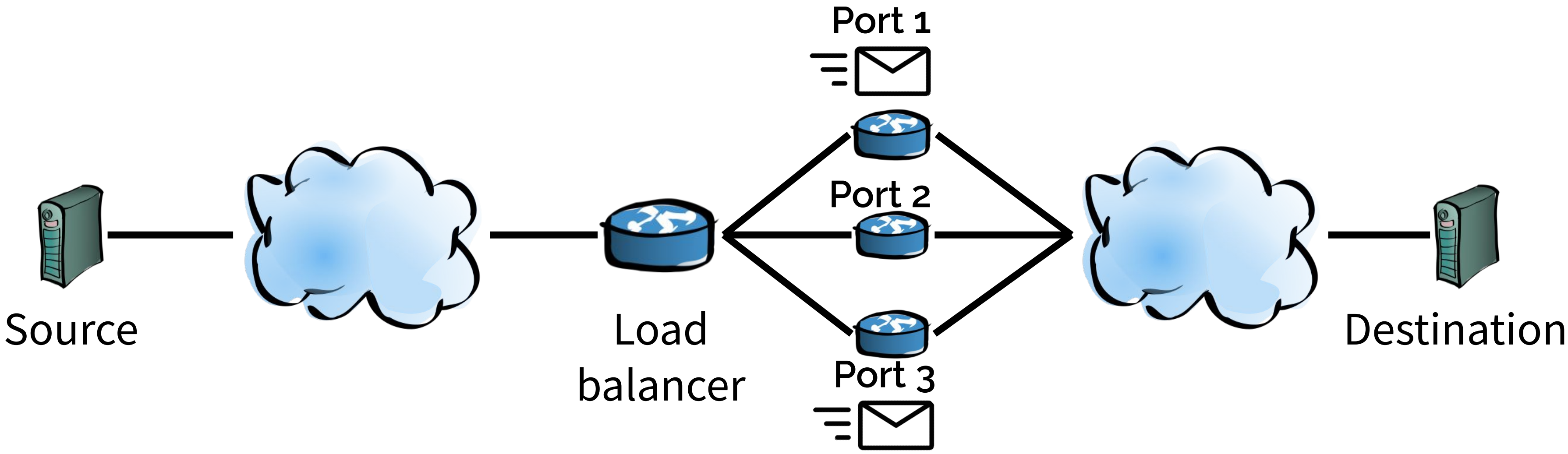
Load Balancers and Hash Domains

Packets with different flow identifiers may take different branches



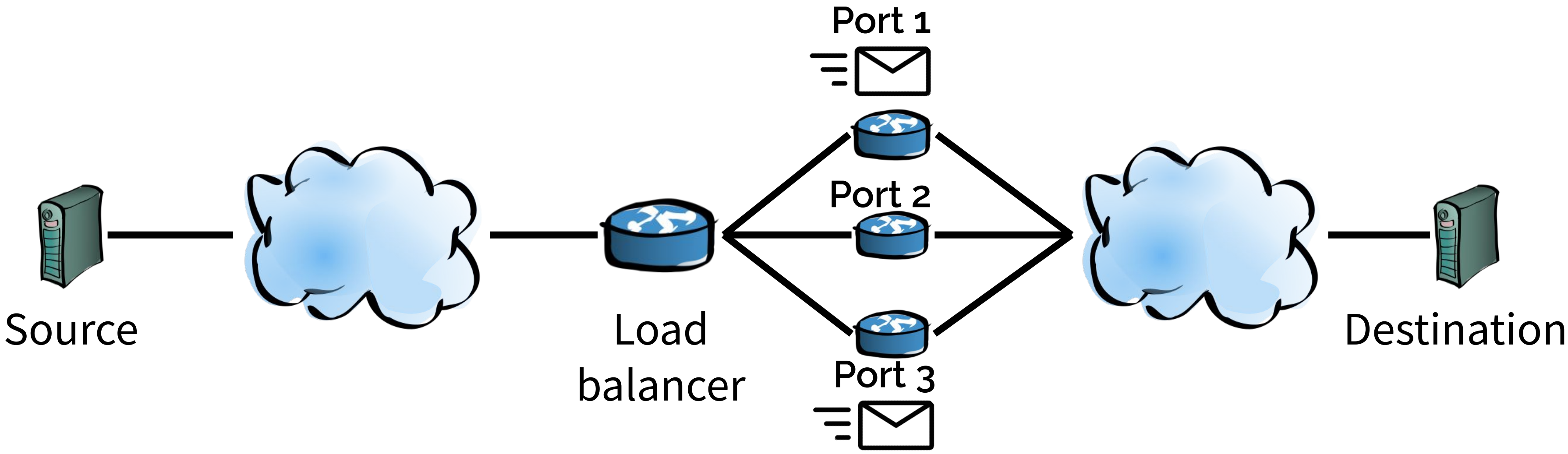
Load Balancers and Hash Domains

Packets with different flow identifiers may take different branches



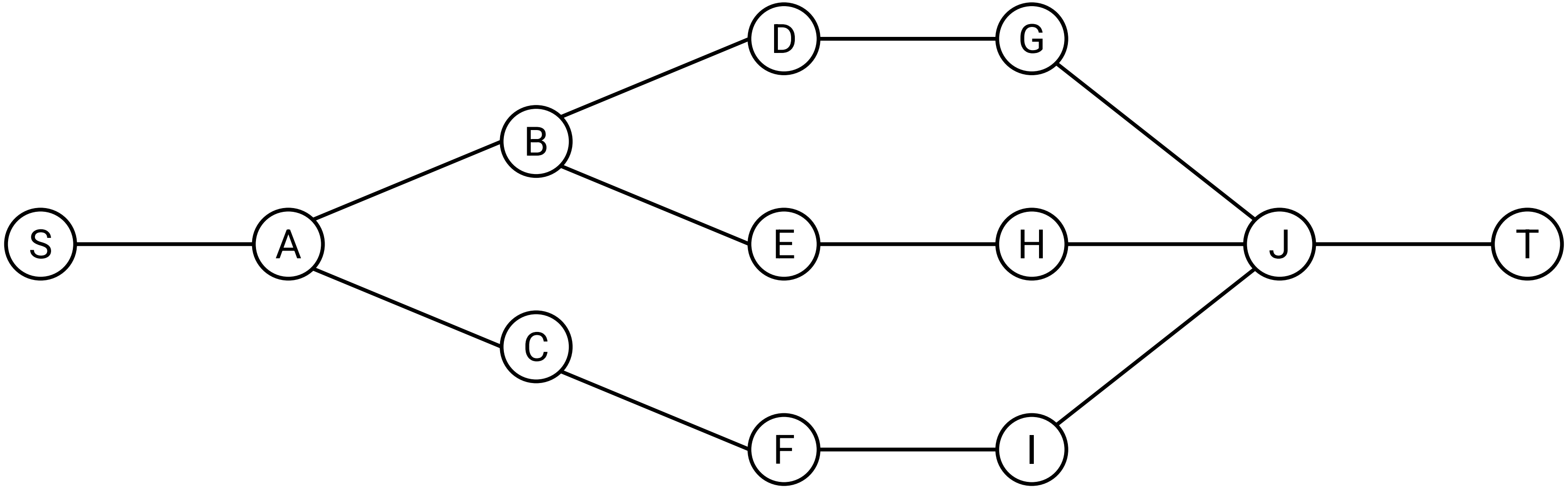
Load Balancers and Hash Domains

Packets with different flow identifiers may take different branches



Good for TCP connections, bad for classic traceroute

The Multipath Detection Algorithm



The MDA systematically varies and tracks flows identifiers to identify all branches

Our Contribution: The Multipath Classification Algorithm

Extensions to the Multipath Detection Algorithm

Our Contribution: The Multipath Classification Algorithm

Extensions to the Multipath Detection Algorithm

MDA

- Detects the three most common classes of load balancing

MCA

- Detects any class of load balancing

Our Contribution:
The Multipath Classification Algorithm

Extensions to the Multipath Detection Algorithm

MDA

- Detects the three most common classes of load balancing
- Ad-hoc identification of three classes of load balancing

MCA

- Detects any class of load balancing
- Principled identification of any class of load balancing

Our Contribution:
The Multipath Classification Algorithm

Extensions to the Multipath Detection Algorithm

MDA

- Detects the three most common classes of load balancing
- Ad-hoc identification of three classes of load balancing

MCA

- Detects any class of load balancing
- Principled identification of any class of load balancing
- Optimizations to reduce probing cost

Outline

Detecting and classifying any class of load balancer

Optimizations for reducing probing cost

Characterization of load balancing in the Internet

Multipath Classification Algorithm

Detect load balancers and links

- Similar to MDA, but varying more bits and ensuring high entropy

Classify load balancers

- Additional probing phase to test each header field at a time

Detecting and Classifying any Load Balancer

No assumption on load balancer behavior

- Hash function need not balance traffic perfectly
  - Increased chance of missing links, but no false links

Detecting and Classifying any Load Balancer

No assumption on load balancer behavior

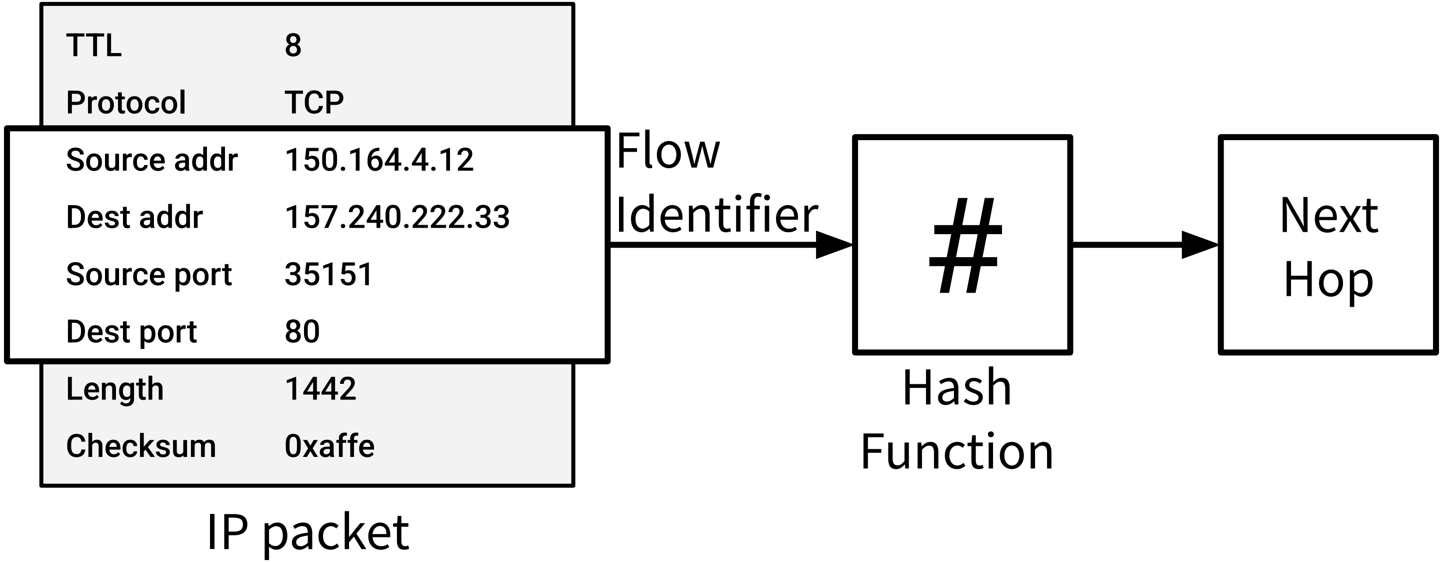
- Hash function need not balance traffic perfectly
  - Increased chance of missing links, but no false links
- Routers (and middleboxes) can modify packet headers
  - Keep track of which hops modify header fields

Detecting and Classifying any Load Balancer

No assumption on load balancer behavior

- Hash function need not balance traffic perfectly
  - Increased chance of missing links, but no false links
- Routers (and middleboxes) can modify packet headers
  - Keep track of which hops modify header fields
- Hash function can use any set of bits in packet headers as the flow id

Varying Flow Identifiers



MCA allows varying most bits in the IP, TCP, UDP, and ICMP headers

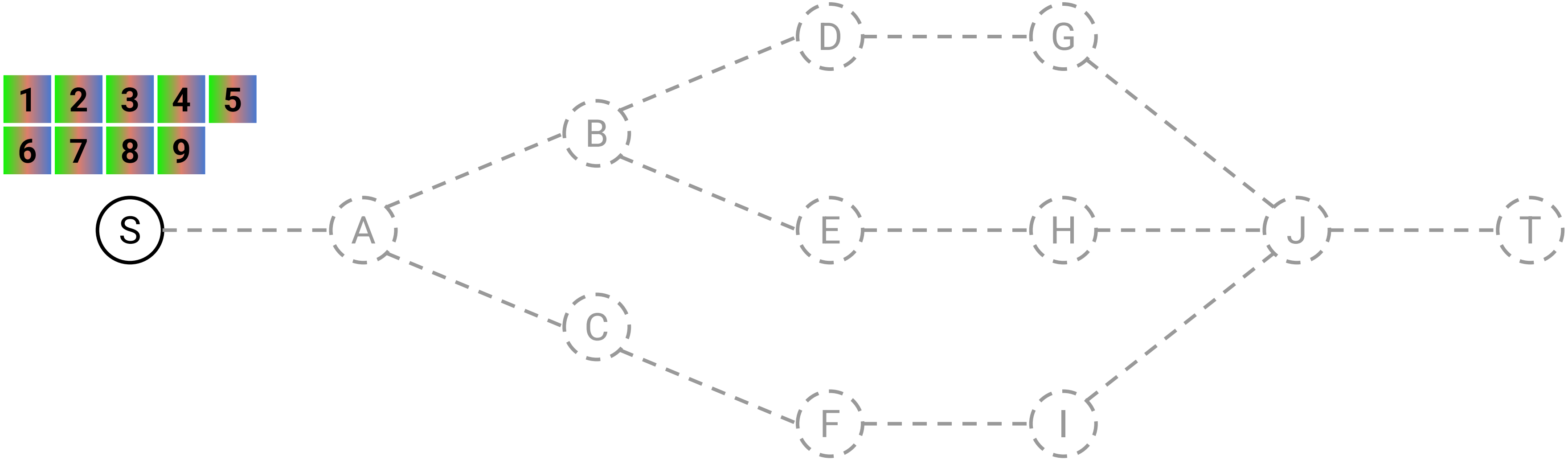
Probe #	Sequential
1	0 0 0 0 1
2	0 0 0 1 0
3	0 0 0 1 1
4	0 0 1 0 0
5	0 0 1 0 1
6	0 0 1 1 0

Probe #	Sequential
1	0 0 0 0 1
2	0 0 0 1 0
3	0 0 0 1 1
4	0 0 1 0 0
5	0 0 1 0 1
6	0 0 1 1 0
Total	0 0 3 3 3

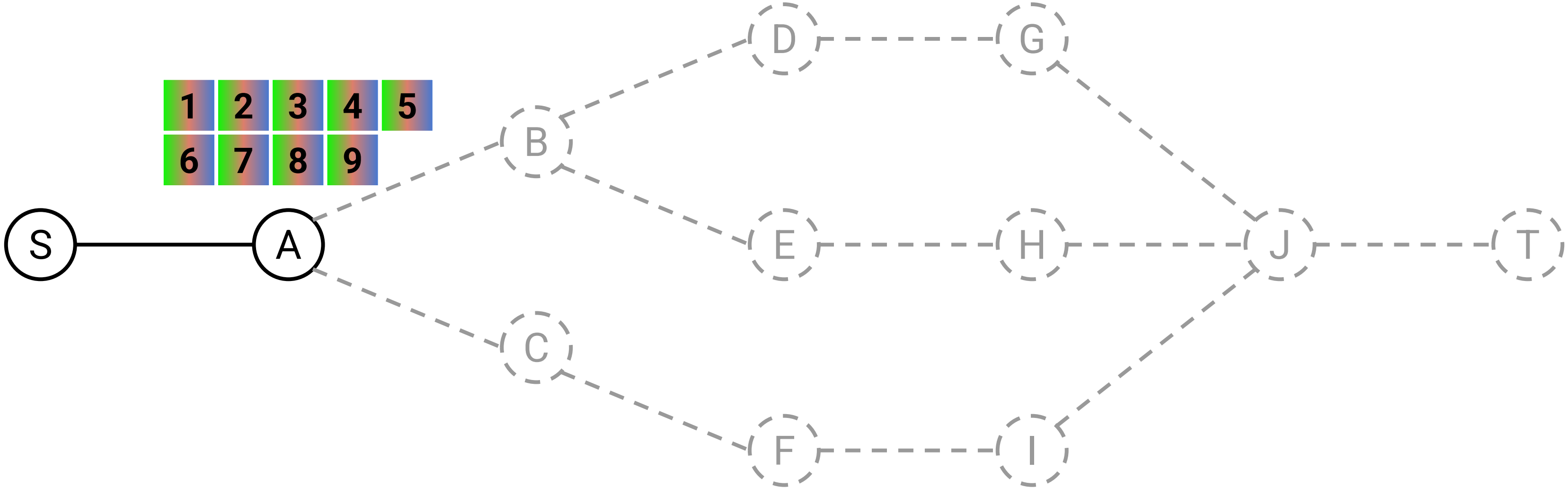
Probe #	Sequential	High entropy
1	0 0 0 0 1	0 0 0 0 1
2	0 0 0 1 0	0 0 1 1 0
3	0 0 0 1 1	0 1 0 1 1
4	0 0 1 0 0	1 0 1 0 1
5	0 0 1 0 1	1 1 0 0 0
6	0 0 1 1 0	1 1 1 1 0
Total	0 0 3 3 3	3 3 3 3 3

Probe #	Sequential	High entropy
1	0 0 0 0 1	0 0 0 0 1
2	0 0 0 1 0	0 0 1 1 0
3	0 0 0 1 1	0 1 0 1 1
4	0 0 1 0 0	1 0 1 0 1
5	0 0 1 0 1	1 1 0 0 0
6	0 0 1 1 0	1 1 1 1 0

How MCA Works - Detection

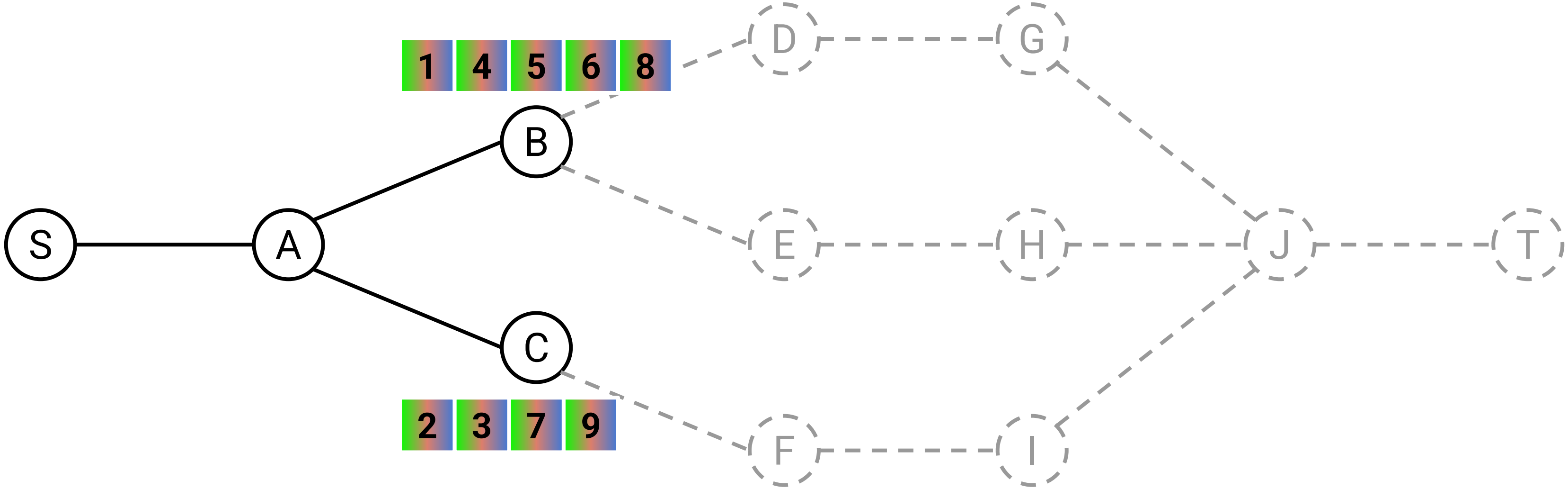


How MCA Works - Detection



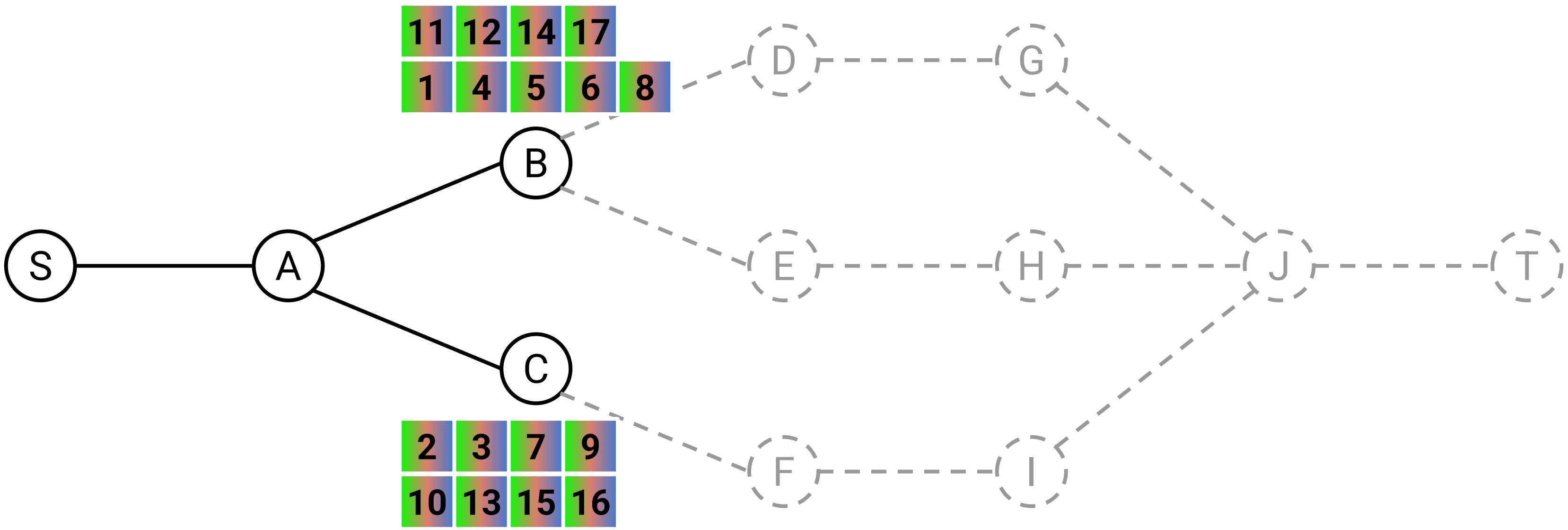
MCA sends 9 probes with TTL = 1 varying all bits, does not find any load balancing, and proceeds to the next hop

How MCA Works - Detection



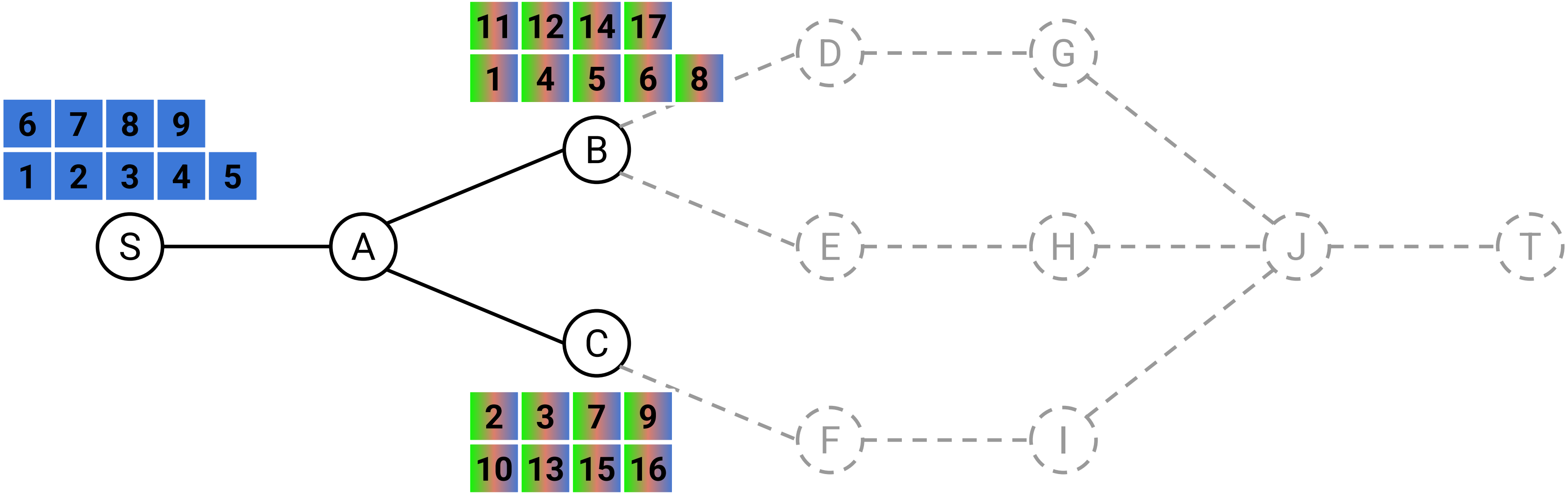
MCA sends 9 probes with TTL = 2, detects load balancing, and sends additional probes to check for more links

How MCA Works - Detection



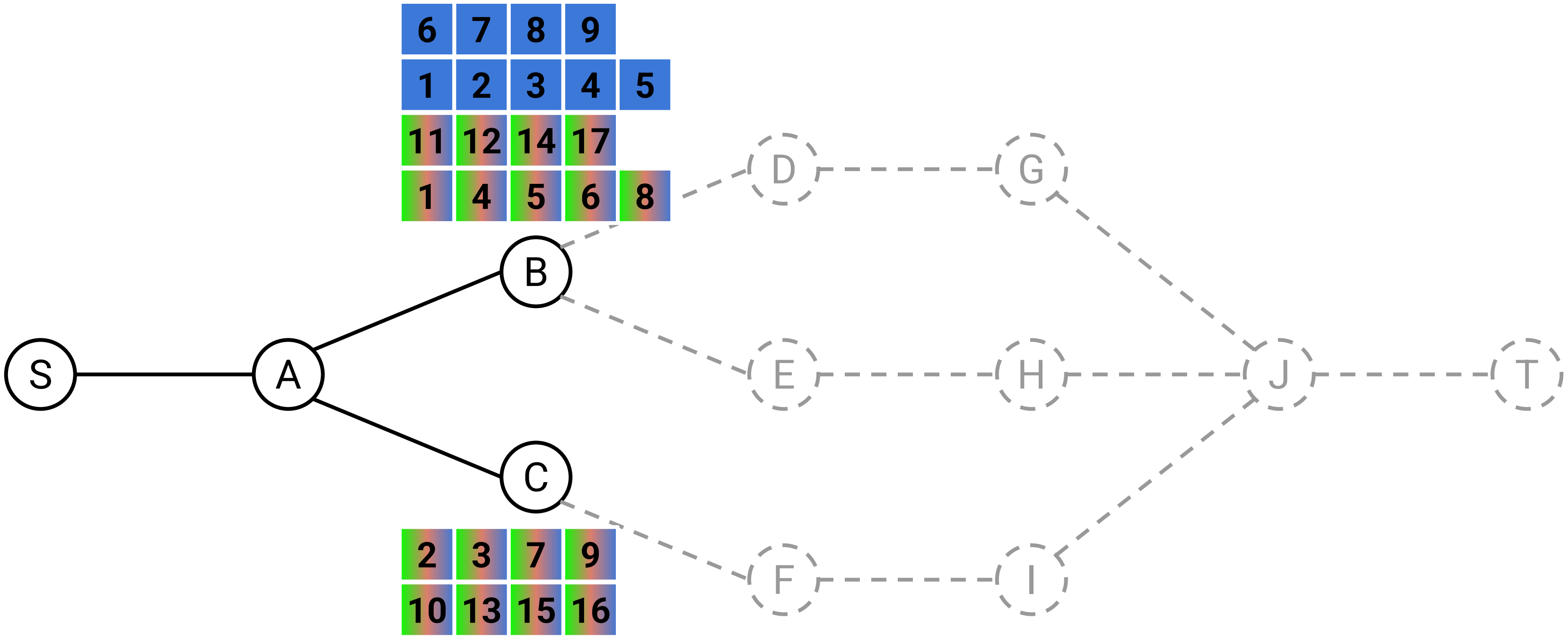
No new links identified with the additional probes: stop detection and perform classification

How MCA Works - Classification



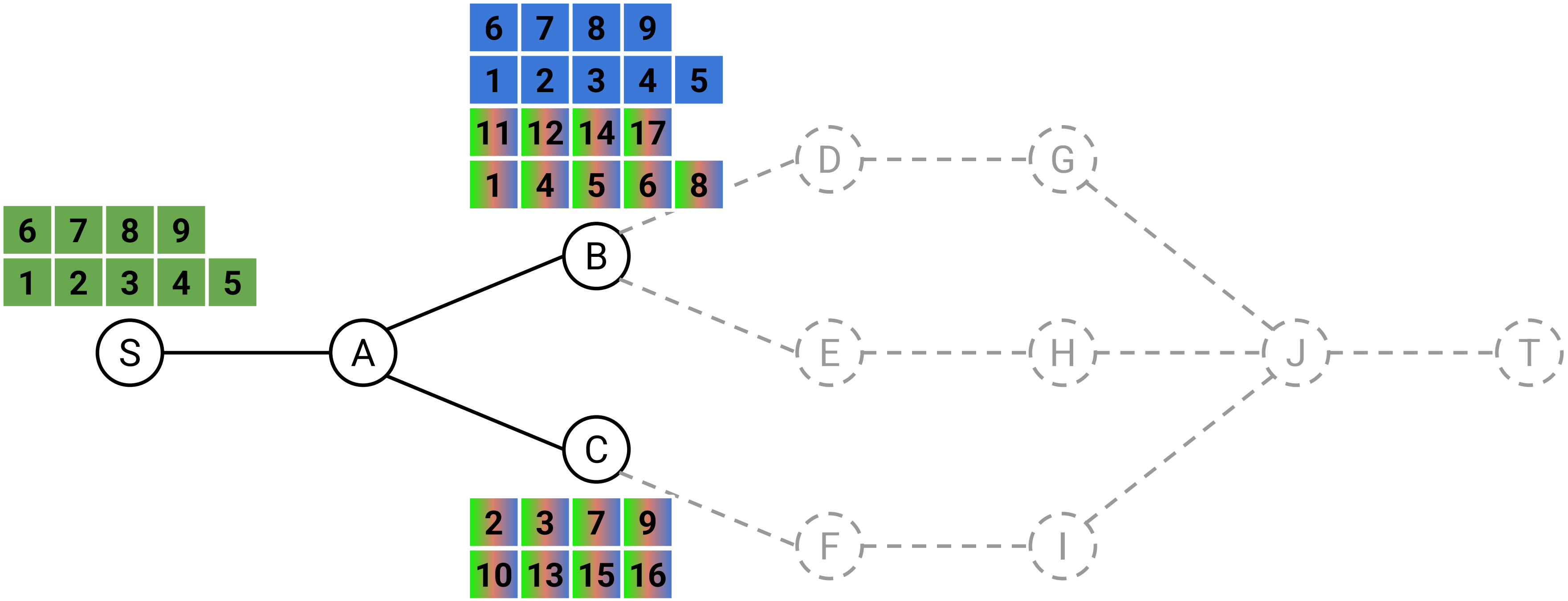
MCA sends 9 probes varying a subset of bits in the packet header, like the port numbers

How MCA Works - Classification



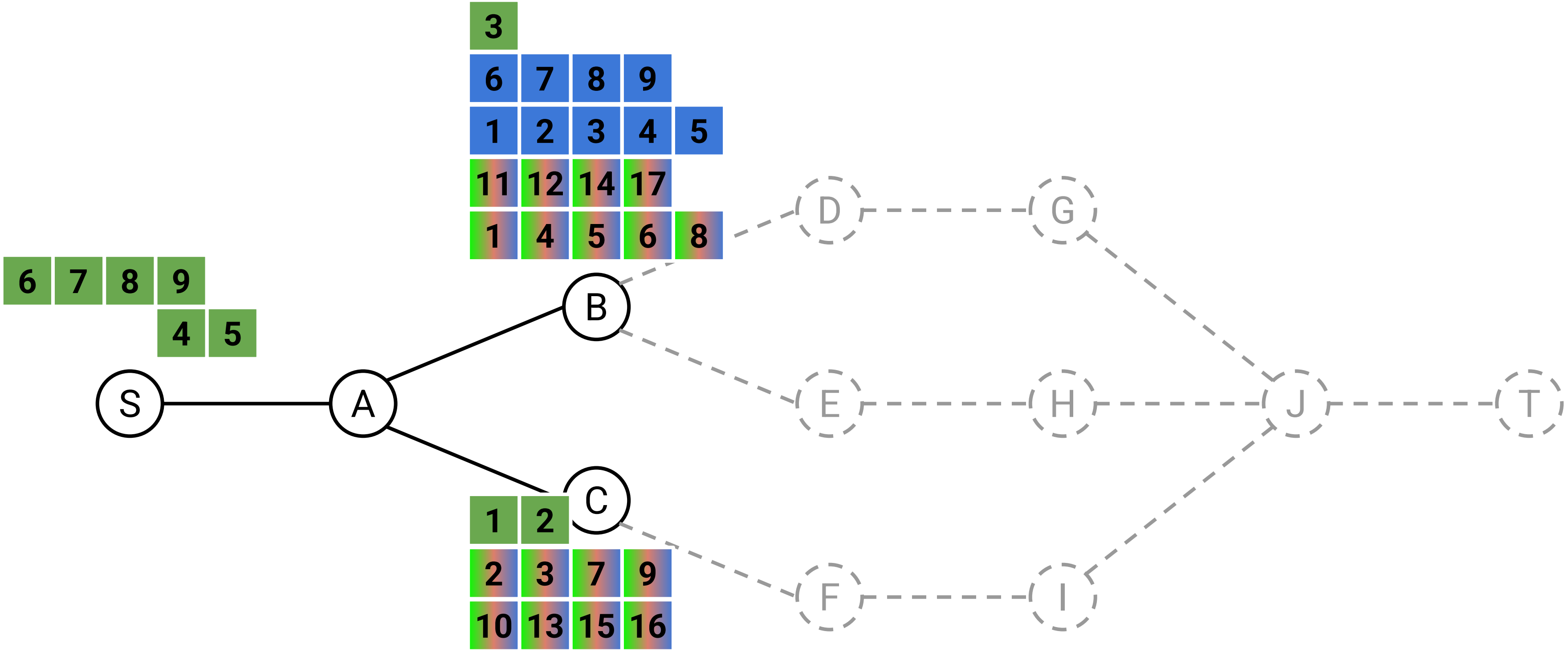
All packets traverse the link A-B, MCA infers that A's hash function does not consider ports

How MCA Works - Classification



Process is repeated for other bits, e.g., the destination address

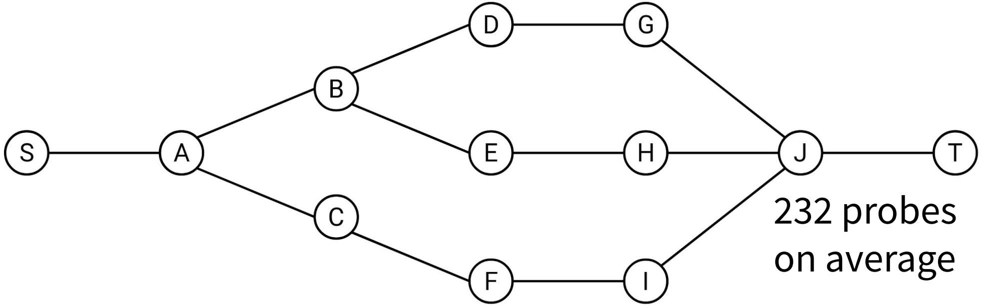
How MCA Works - Classification



As soon as MCA identifies load balancing, it infers A’s hash function includes the destination address and proceeds

Optimizations to Reduce Probing Cost

Detection and classification require many probes



Optimizations reduce probing with no impact on precision

- Probing cost reduction of 8% for the default configuration
- With our optimizations, classification is ≈35% of the probing cost
  - Reasonable increase on top of detection (MDA)

Outline

Detecting and classifying any class of load balancer

Optimizations for reducing classification cost

Characterization of Load Balancing in the Internet

Dataset Coverage

Deployed MCA on 4 different cloud providers

- 31 vantage points in 5 continents

Dataset Coverage

Deployed MCA on 4 different cloud providers

- 31 vantage points in 5 continents
- MCA measurements to
  - 19866 IPv4 destinations: coverage of 4388 ASes
  - 16674 IPv6 destinations: coverage of 8103 ASes
- 2.7 million MCA measurements total

MCA Configuration

Vary many bits in packet headers

- Cover known load balancer classes and allow discovery of new classes

Measurements for IPv4 and IPv6

- Vary last bits of destination address, DSCP, traffic class, flow label

Measurements for TCP, UDP, and ICMP

- Vary ports and checksum

Load Balancing is Prevalent

75% of IPv4 routes traverse at least one load balancer, and 10% traverse more than 10.

56% of IPv6 routes traverse at least one load balancer, and 3% traverse more than 10.

Classes of Load Balancing

Class	Header Fields
Per-flow	IP addresses and ports
Per-dest	IP addresses
Per-packet	Contents of packet header ignored, forwarded to a random next hop

Classes of Load Balancing

Class	Header Fields
Per-flow	IP addresses and ports
Per-dest	IP addresses
Per-packet	Contents of packet header ignored, forwarded to a random next hop
Per-app	TCP and UDP ports
v6 flow label	Any class (other than per-packet), but also using the IPv6 flow label field
Other	Load balancers considering other flow identifiers

Classes of Load Balancing

	IPv4			IPv6		
	UDP	TCP	ICMP	UDP	TCP	ICMP
Per-flow						
Per-dest						
Per-packet						
Per-app						
v6 flow label						
Other						

Classes of Load Balancing

	IPv4			IPv6		
	UDP	TCP	ICMP	UDP	TCP	ICMP
Per-flow	69.6%	69.8%	1.5%	77.6%	78.5%	0.3%
Per-dest	24.4%	24.1%	94.2%	13.3%	13.5%	90.1%
Per-packet						
Per-app						
v6 flow label						
Other						

Classes of Load Balancing

	IPv4			IPv6		
	UDP	**TCP**	**ICMP**	**UDP**	**TCP**	**ICMP**
Per-flow	69.6%	69.8%	1.5%	77.6%	78.5%	0.3%
Per-dest	24.4%	24.1%	94.2%	13.3%	13.5%	90.1%
Per-packet						
Per-app						
v6 flow label						
Other						

Classes of Load Balancing

	IPv4			IPv6		
	UDP	**TCP**	**ICMP**	**UDP**	**TCP**	**ICMP**
Per-flow	69.6%	69.8%	1.5%	77.6%	78.5%	0.3%
Per-dest	24.4%	24.1%	94.2%	13.3%	13.5%	90.1%
Per-packet	0.1%	0.1%	0.1%	0.1%	0.1%	0.3%
Per-app						
v6 flow label						
Other						

Classes of Load Balancing

	IPv4			IPv6		
	UDP	**TCP**	**ICMP**	**UDP**	**TCP**	**ICMP**
Per-flow	69.6%	69.8%	1.5%	77.6%	78.5%	0.3%
Per-dest	24.4%	24.1%	94.2%	13.3%	13.5%	90.1%
Per-packet	0.1%	0.1%	0.1%	0.1%	0.1%	0.3%
Per-app	1.8%	2.1%	0.0%	0.9%	0.8%	0.0%
v6 flow label						
Other						

Classes of Load Balancing

	IPv4			IPv6		
	UDP	TCP	ICMP	UDP	TCP	ICMP
Per-flow	69.6%	69.8%	1.5%	77.6%	78.5%	0.3%
Per-dest	24.4%	24.1%	94.2%	13.3%	13.5%	90.1%
Per-packet	0.1%	0.1%	0.1%	0.1%	0.1%	0.3%
Per-app	1.8%	2.1%	0.0%	0.9%	0.8%	0.0%
v6 flow label	-	-	-	3.1%	2.7%	3.2%
Other	2.3%	2.6%	2.7%	3.2%	2.8%	3.9%

Classes of Load Balancing

	IPv4			IPv6		
	UDP	**TCP**	**ICMP**	**UDP**	**TCP**	**ICMP**
Per-flow	69.6%	69.8%	1.5%	77.6%	78.5%	0.3%
Per-dest	24.4%	24.1%	94.2%	13.3%	13.5%	90.1%
Per-packet	0.1%	0.1%	0.1%	0.1%	0.1%	0.3%
Per-app	1.8%	2.1%	0.0%	0.9%	0.8%	0.0%
v6 flow label	-	-	-	3.1%	2.7%	3.2%
Other	2.3%	2.6%	2.7%	3.2%	2.8%	3.9%

Classes of Load Balancing

	IPv4			IPv6		
	UDP	TCP	ICMP	UDP	TCP	ICMP
Per-flow	69.6%	69.8%	1.5%	77.6%	78.5%	0.3%
Per-dest	24.4%	24.1%	94.2%	13.3%	13.5%	90.1%
Per-packet	0.1%	0.1%	0.1%	0.1%	0.1%	0.3%
Per-app	1.8%	2.1%	0.0%	0.9%	0.8%	0.0%
v6 flow label	-	-	-	3.1%	2.7%	3.2%
Other	2.3%	2.6%	2.7%	3.2%	2.8%	3.9%

Conclusion

Multipath Classification Algorithm

- No assumptions on load balancer behavior
- Detects 5% more load balancers than previous solutions
- Reasonable probing cost

Characterization of Load Balancing in the Internet

- Revisited results from previous characterizations
- Load balancing remains prevalent, and behavior has improved

Software and Dataset Available

Python/Scapy implementation of MCA:

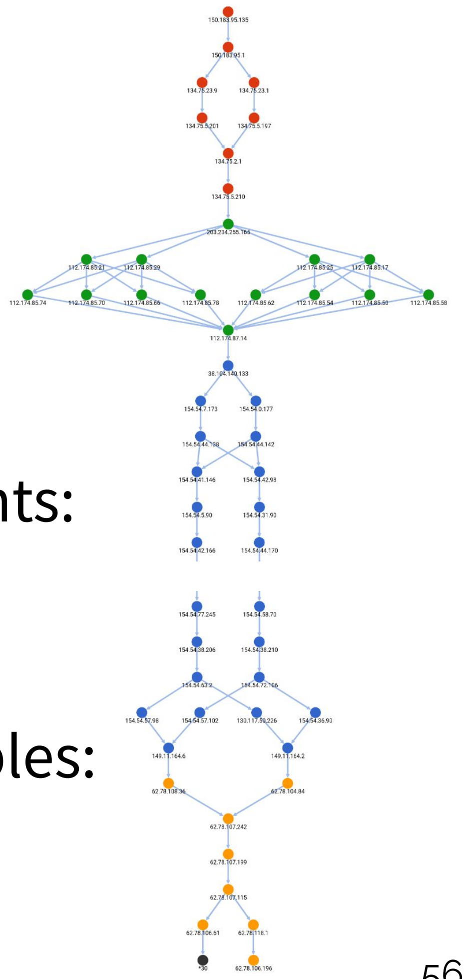
- pip3 install mca-traceroute

Route Explorer is a front-end for MCA measurements:

- https://github.com/rlcalmeida/route-explorer

Dataset, along with interesting handpicked examples:

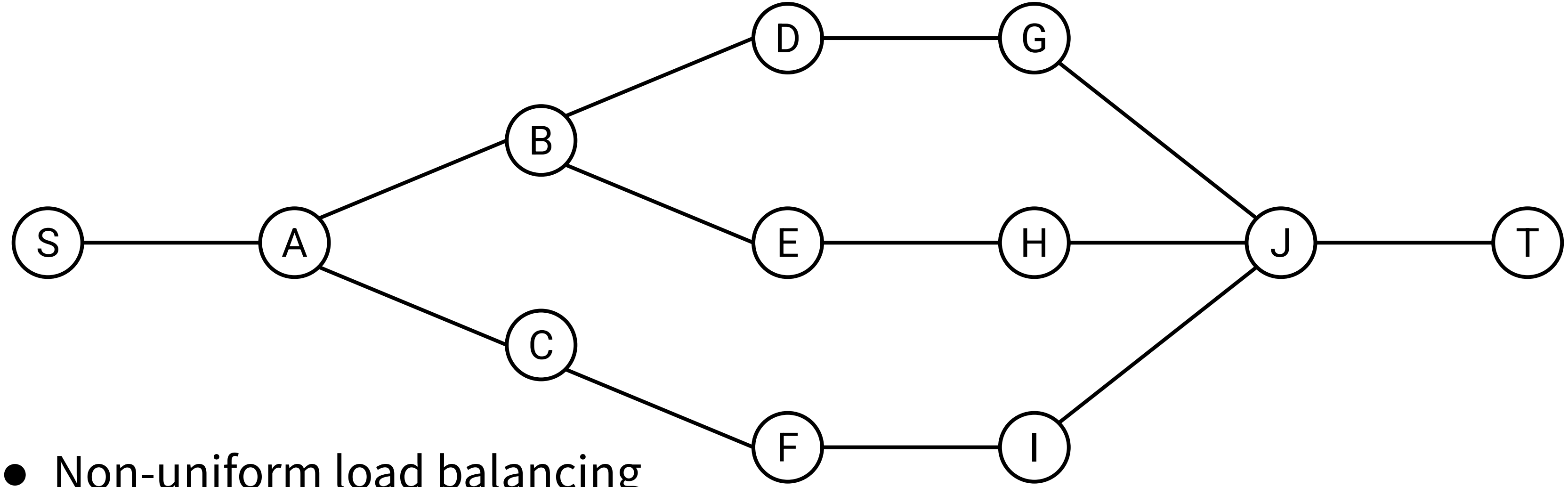
- https://www.dcc.ufmg.br/~rlca/mca



Classification of Load Balancing in the Internet

Rafael Almeida, **Ítalo Cunha**, Darryl Veitch,
Renata Teixeira, Christophe Diot

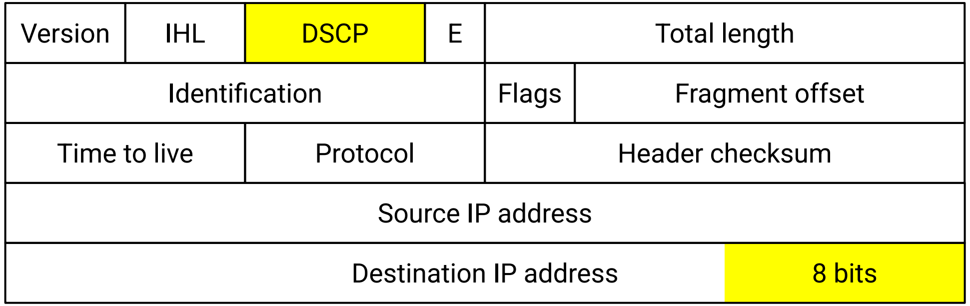
Operational Considerations



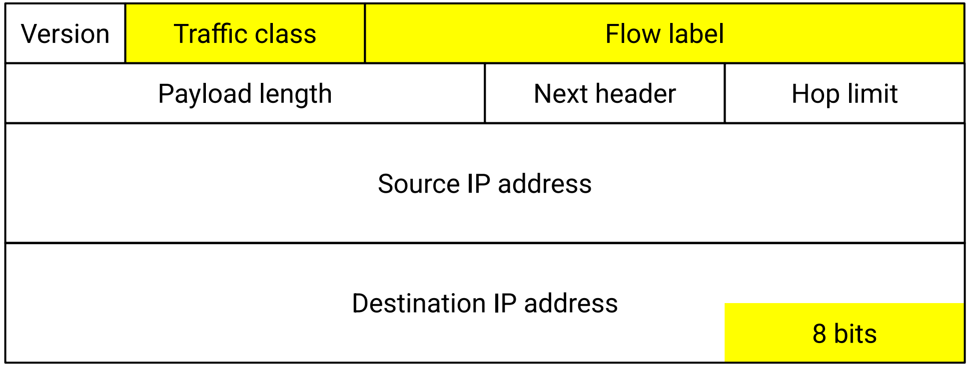
- Non-uniform load balancing
- TTL in the flow identifier
- Polarization

Measurements Setup

IPv4



IPv6



Measurements Setup

TCP

Source port		Destination port
Sequence number		
Acknowledge number		
Len	Reserved + flags	Window size
Checksum		Urgent pointer

UDP

Source port	Destination port
Length	Checksum

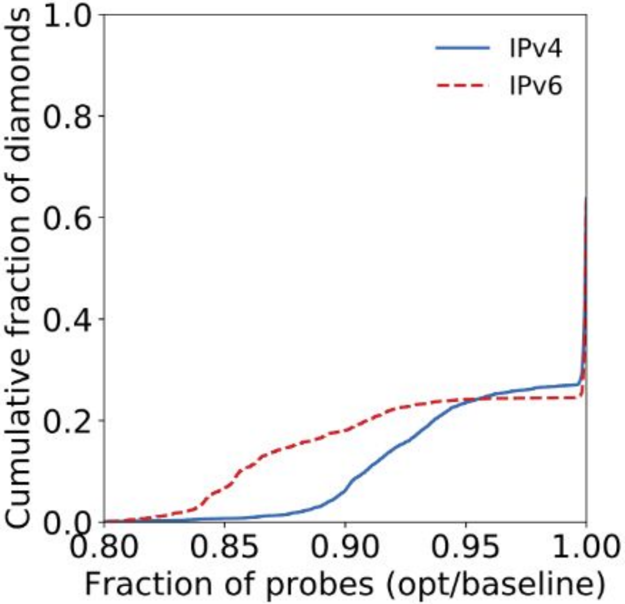
ICMP

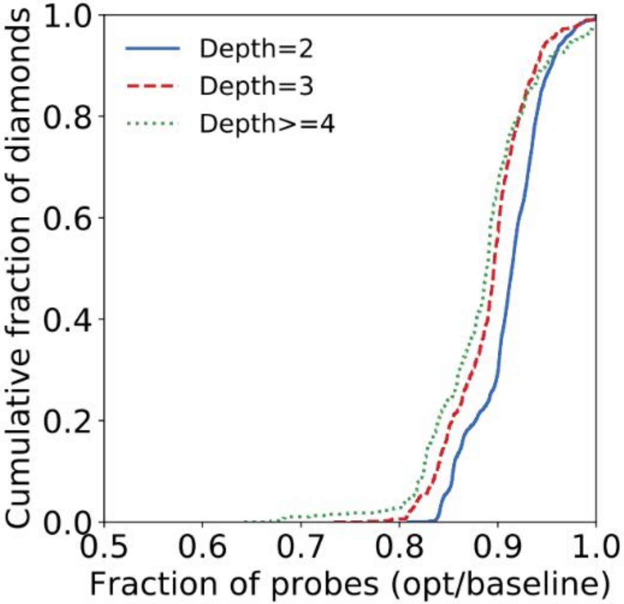
Type	Code	Checksum
Length		Sequence number

Dataset

			Number of traces		
Platform	**# VP**	**Period**	**IPv4**	**IPv6**	**# AS**
UFMG	1	2018-08-21–2018-09-06	16,272	18,684	1,540
Linode	6	2018-08-21–2019-03-01	262,752	242,088	6,787
Vultr	6	2018-08-21–2019-03-01	305,628	263,136	7,586
DigitalOcean	7	2018-08-21–2019-03-01	356,808	321,180	7,587
CAIDA Ark	11	2018-08-21–2019-04-27	571,104	469,464	8,939
All	**31**	**2018-08-21–2019-04-27**	**1,512,564**	**1,314,552**	**10,454**

Fraction of probes sent when using optimizations

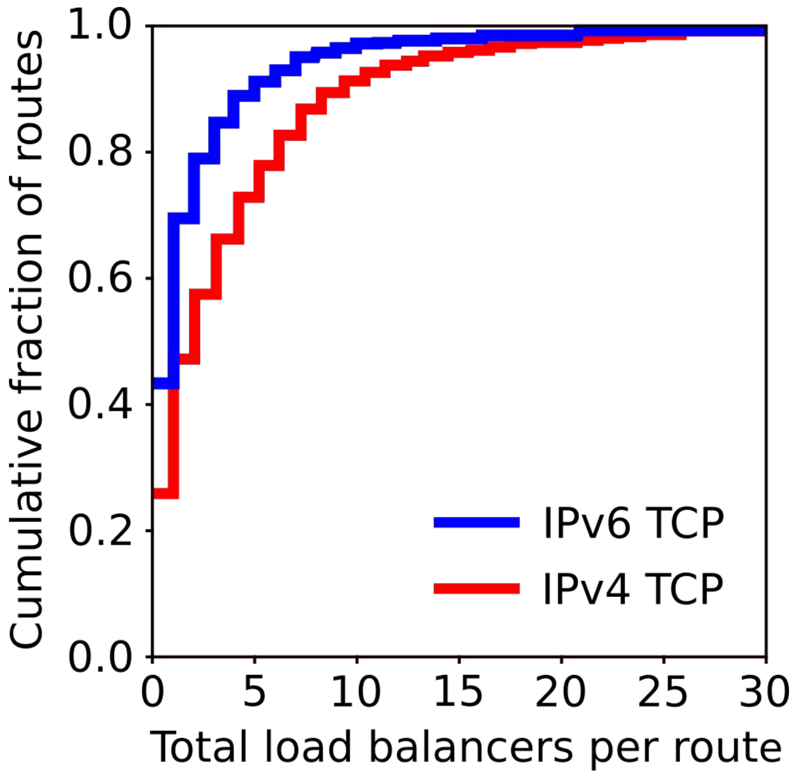




Fraction of probes sent when using optimizations

- Reduction in the detection step
    - 2.8% for IPv4 and 0.7% for IPv6
- Reduction in the classification step
    - 11% for IPv4 and 18% for IPv6
- Overall reduction
    - 6% for IPv4 and 8% for IPv6

Load Balancing is Prevalent



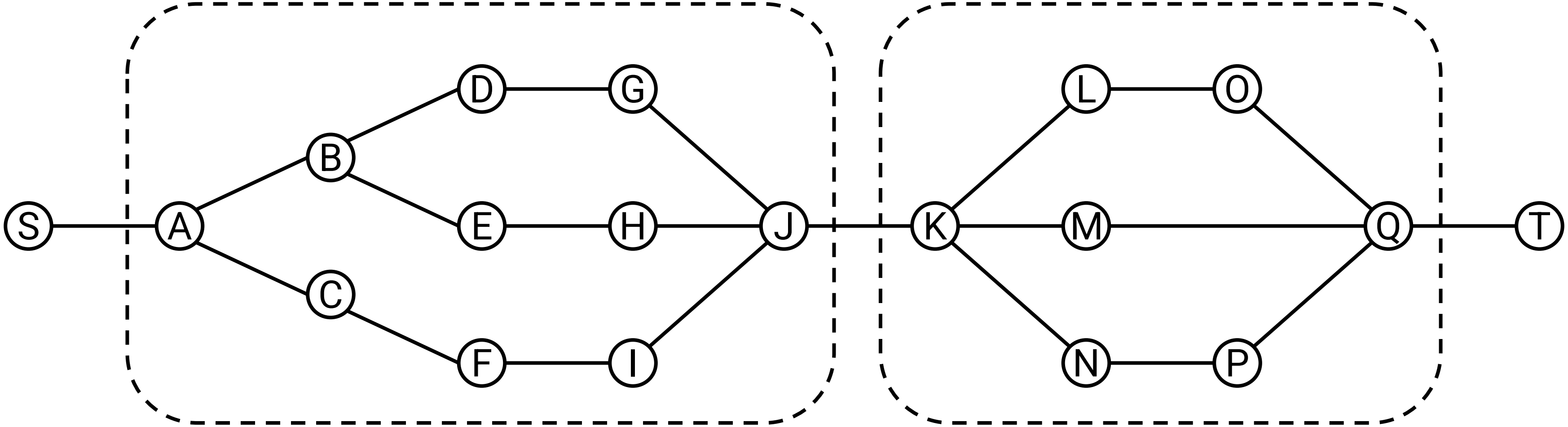
75% of the IPv4 routes traverse at least one load balancer (56% of IPv6 routes).

Some routes have 10+ load balancers.

Load Balancer Diamonds

Sequences of hops between branch and join points

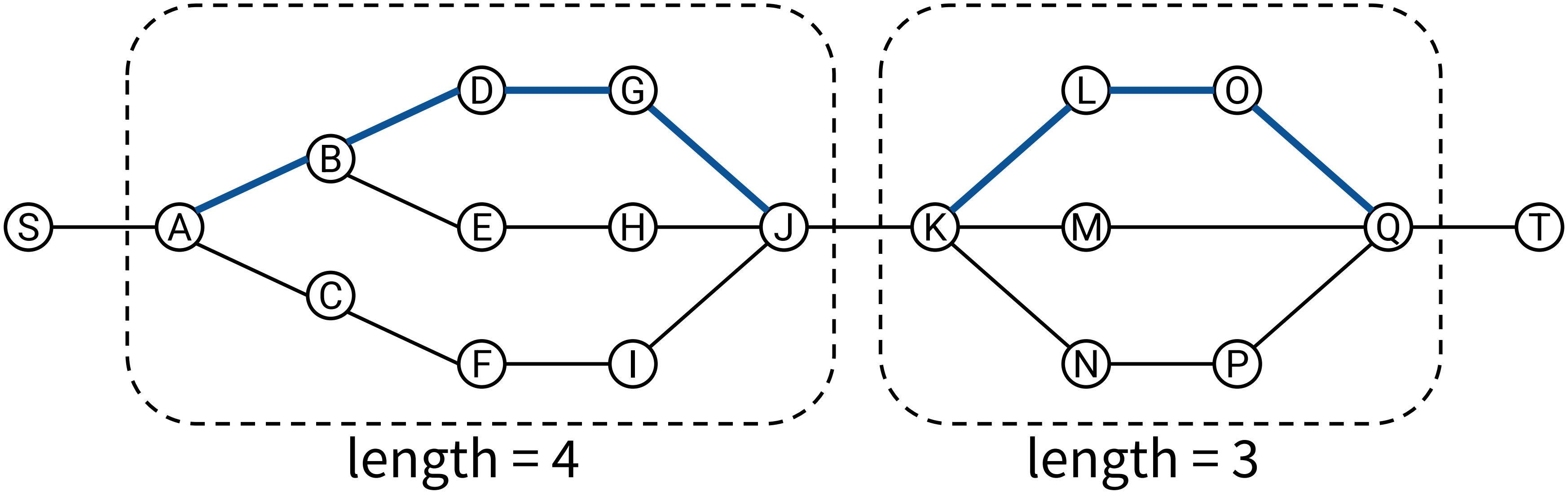
- IPv4 diamonds more complex than IPv6 diamonds
- Similar characteristics across transport protocols



Diamond Length

Maximum number of hops between branch and join points

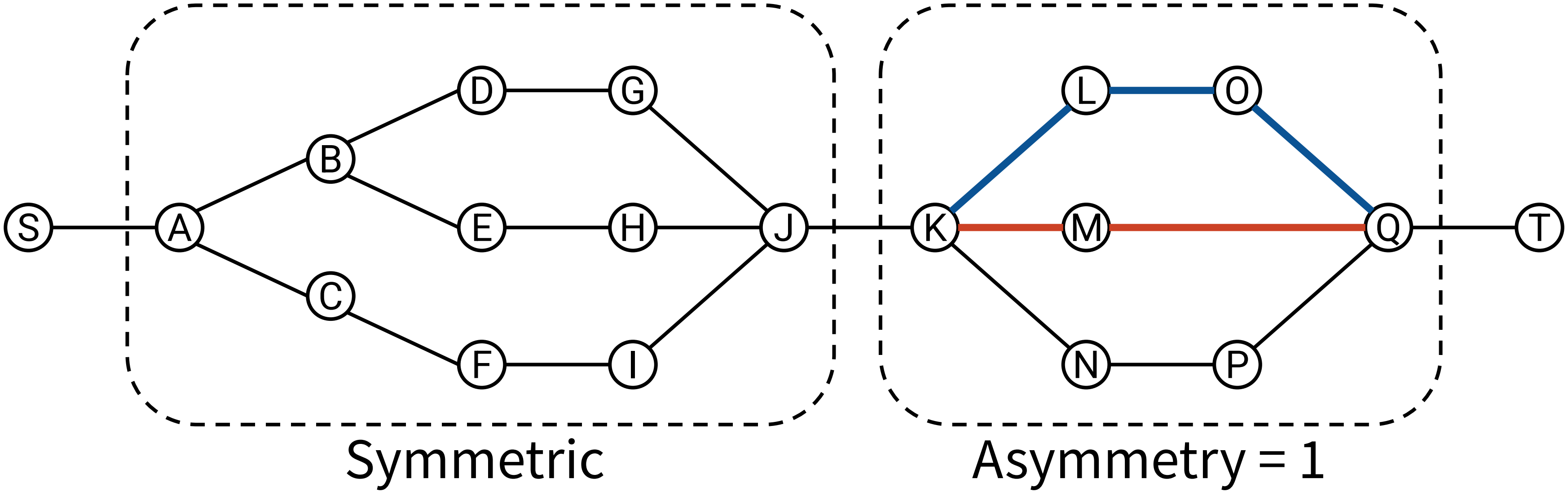
- 80% of IPv4 diamonds shorter than 5 hops



Diamond Asymmetry

Maximum difference in branch length in diamond

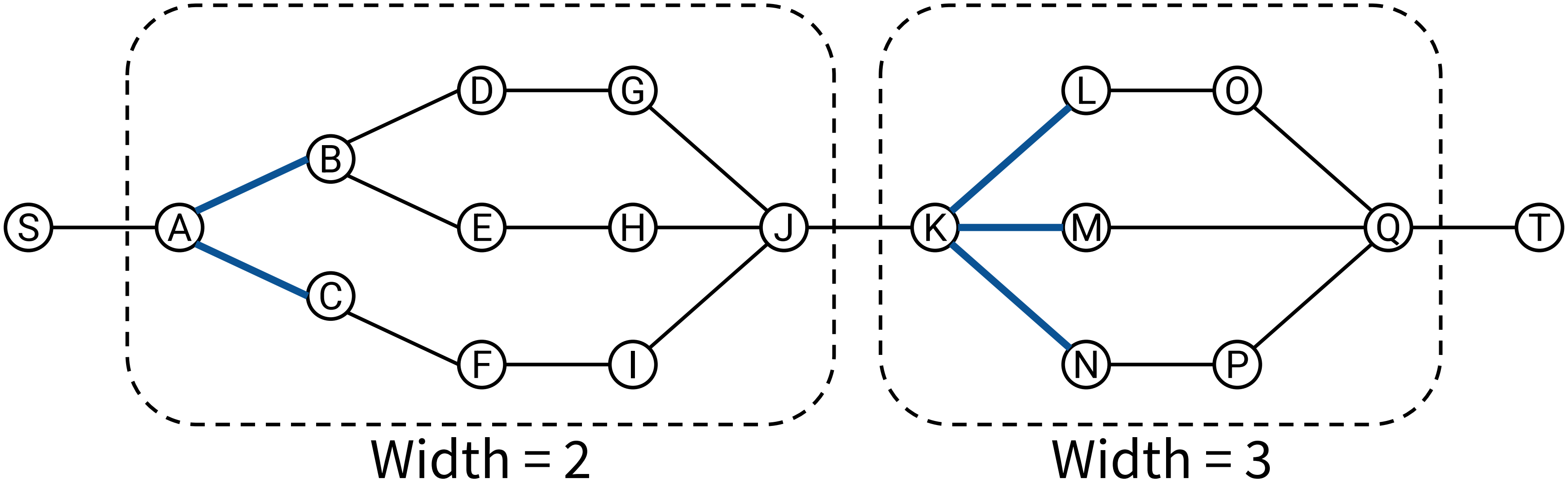
- 80% of IPv4 diamonds are symmetric



Diamond Width

Number of link-disjoint branches in diamond

- 75% of IPv4 diamonds have 2 link-disjoint branches



Diamonds Depth

IPv4: 38% have depth >= 2

IPv6: 30% have depth >= 2