

# CSE 550: *Systems for all*

Au 2022

Ratul Mahajan

**June 15, 2020 T-Mobile  
Network Outage Report**

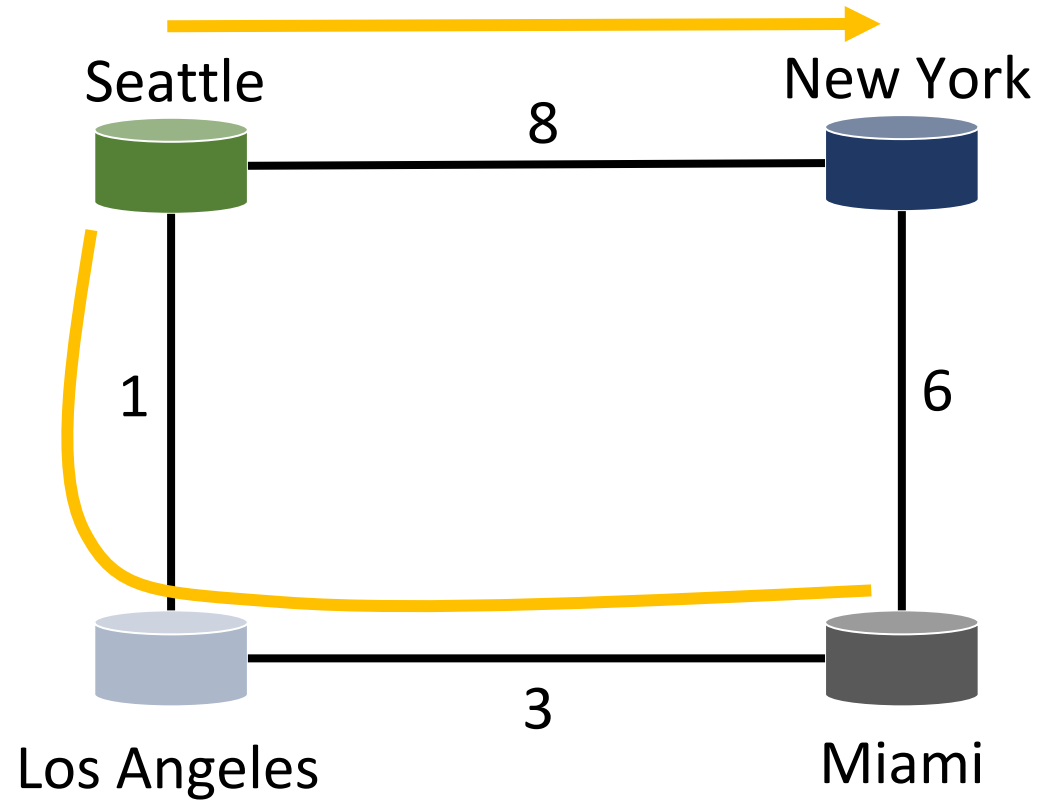
**PS Docket No. 20-183**

A Report of the Public Safety and Homeland Security Bureau  
Federal Communications Commission  
October 22, 2020

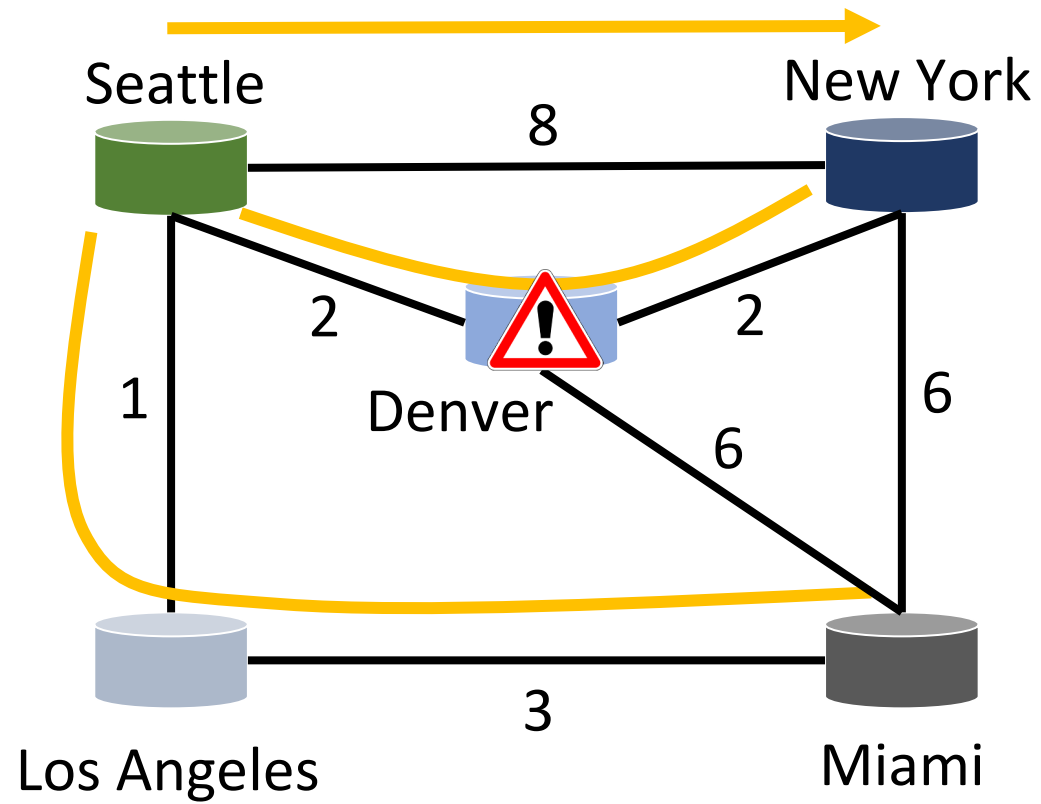
“At least 41% of all calls that attempted to use T-Mobile’s network during the outage failed, including at least 23,621 failed calls to 911.”

“[An old woman] who has dementia, could not reach [her son] after her car would not start and her roadside-assistance provider could not call her to clarify her location; she was stranded for seven hours”

# Anatomy of the outage (illustration)

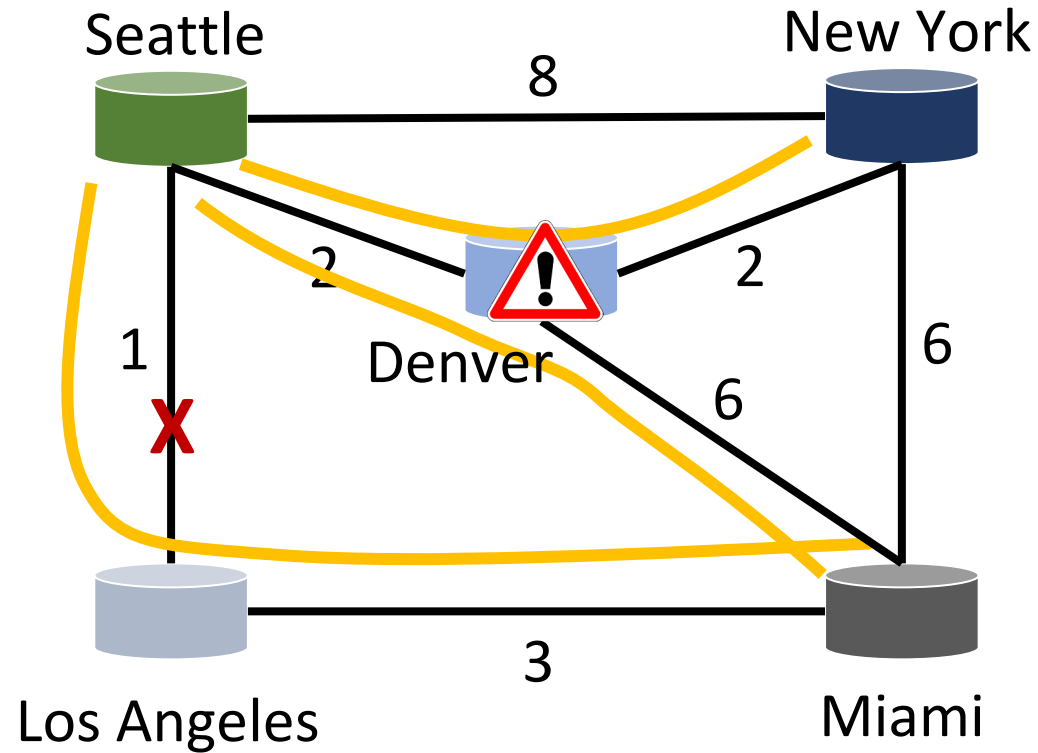


# Anatomy of the outage (illustration)



# Anatomy of the outage (illustration)

What if T-Mobile could guarantee that no traffic will transit Denver?



What if T-Mobile could predict the impact of link failure?

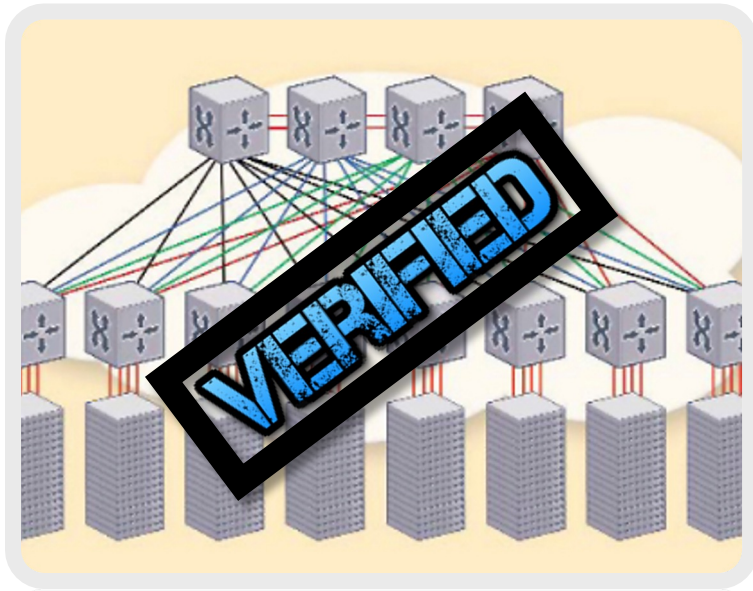
# Network verification

**Guarantee network behavior<sup>\*†</sup>**

*\* Some aspect of behavior*

*† Under some assumptions*

# A horizontal slice of the problem



Configuration, state

Verify

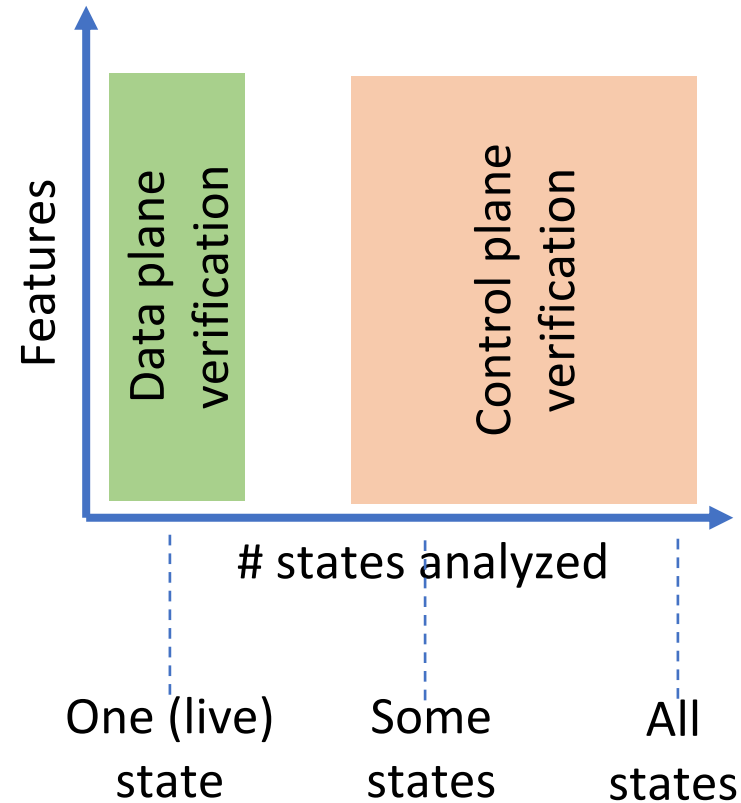
Software (OS, protocols)

Trust

Hardware

# The space of network verification tools

Shortest-path or policy routing?  
Are packet transformed?  
Stateless or stateful forwarding?  
...

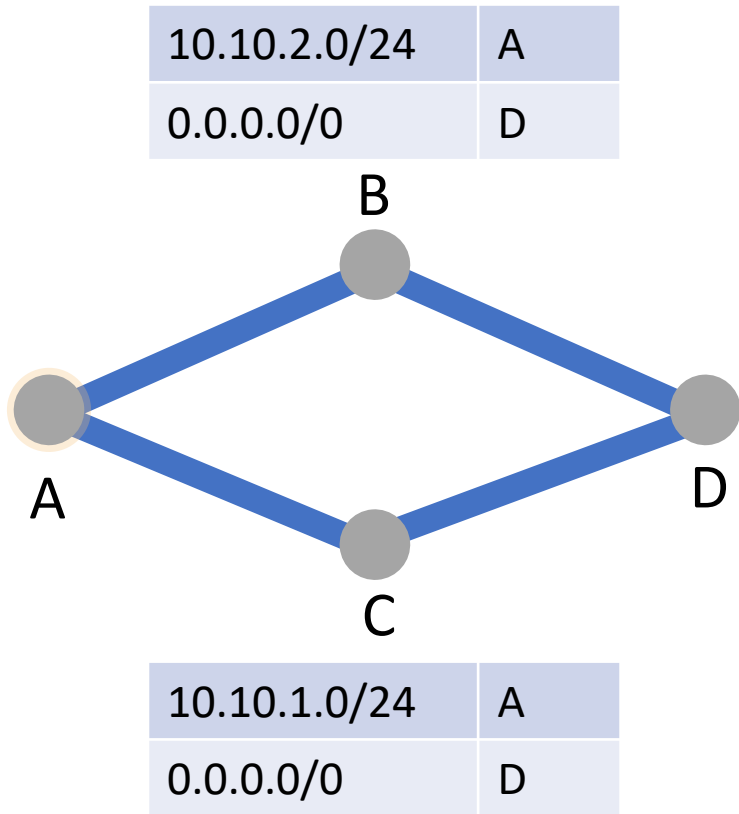




# Data plane verification

***Who*** can talk to whom using  
which ***packets*** and ***paths*** in  
**one state** of the network?

10.10.1.0/24	B
10.10.0.0/16	C
0.0.0.0/0	B

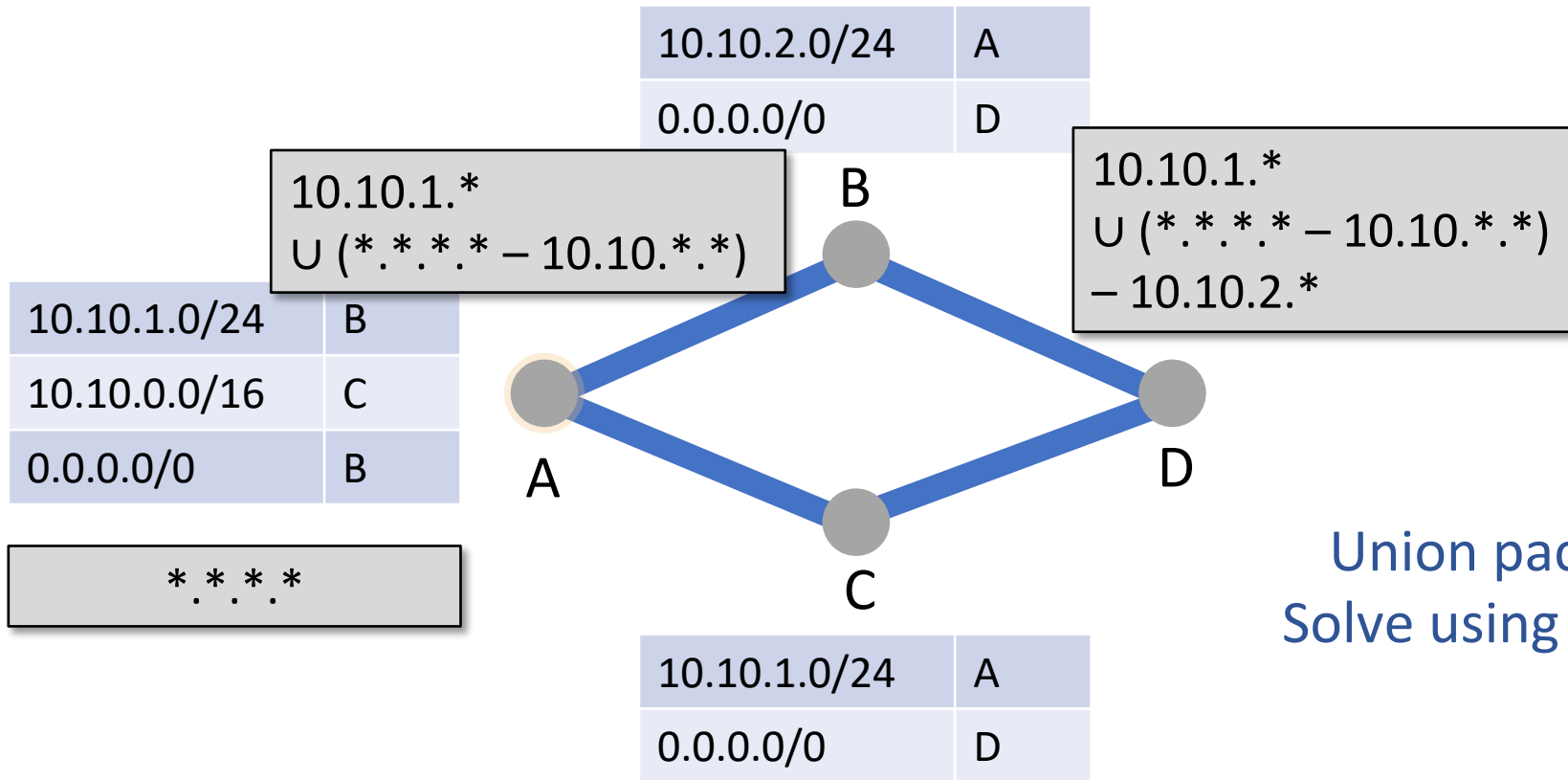


10.10.2.0/24	A
0.0.0.0/0	D

10.10.1.0/24	A
0.0.0.0/0	D

Can A talk to D and using which packets?

# 💡 DPV idea: Ternary simulation



Union packet sets along possible paths  
Solve using custom data structure or BDDs

# Control plane verification

**Who** can talk to whom using which *packets* and **paths** in **many states** of the network?



Finds bugs proactively  
Enables what if analysis

# Verifying distributed control planes

Routers generate and process messages per low-level directives

```
OSPF INTERFACE INT2_I METRIC I
```

```
OSPF INTERFACE INT2_I METRIC I
```

```
OSPF REDISTRIBUTED CONNECTED METRIC IO
```

```
IP PREFIX-LIST PLI DENY 192.168.0.0/16 LE 32
```

```
IP PREFIX-LIST PLI ALLOW
```

```
ROUTE-MAP FROMR2 IO
```

```
MATCH IP ADDRESS PREFIX-LIST PLI
```

```
SET LOCAL-PREFERENCE I20
```

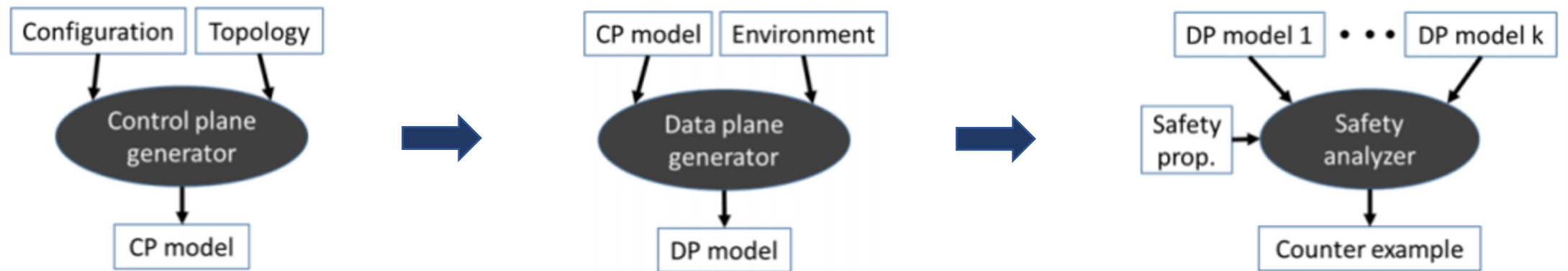
Goal

Reason about states that emerge when many such programs run concurrently



# CPV idea #1: Simulate the control plane

1. Simulate the control plane to generate data plane states
2. Use DPV to analyze the states



Can analyze *any* data plane but not *all* data planes?



## CPV idea #2: Encode the fixed point

1. Valid network states are fixed points of the control plane
2. Fixed points can be formally encoded

ARC [2016] use a graph encoding (not general)

Minesweeper [2017] uses SMT encoding

Over to Aleksei and Yuan-Mao



So, what did we learn this quarter?