Replication:
A Two Decade review of Policy Atoms Tracing the Evolution of AS Path Sharing Prefixes

Weili Wu, Zachary Bischof, Cecilia Testart, Alberto Dainotti

Georgia Tech

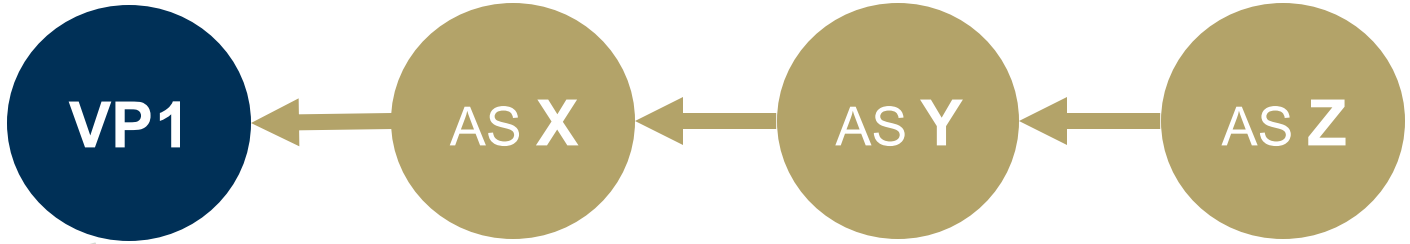
Replicated a 2002 paper

- “Policy atoms” capture prefixes experiencing the same routing policies
  - Andre Broido, kc Claffy, “*Analysis of RouteViews BGP data: policy atoms*”. NRDM Workshop 2001
  - Yehuda Afek, Omer Ben-Shalom, and Anat Bremler-Barr,
    “***On the structure and application of BGP policy atoms”***. Internet Measurement Workshop 2002

What we replicated

What is a policy atom

A policy atom is a set of prefixes that share the same AS paths within each vantage point



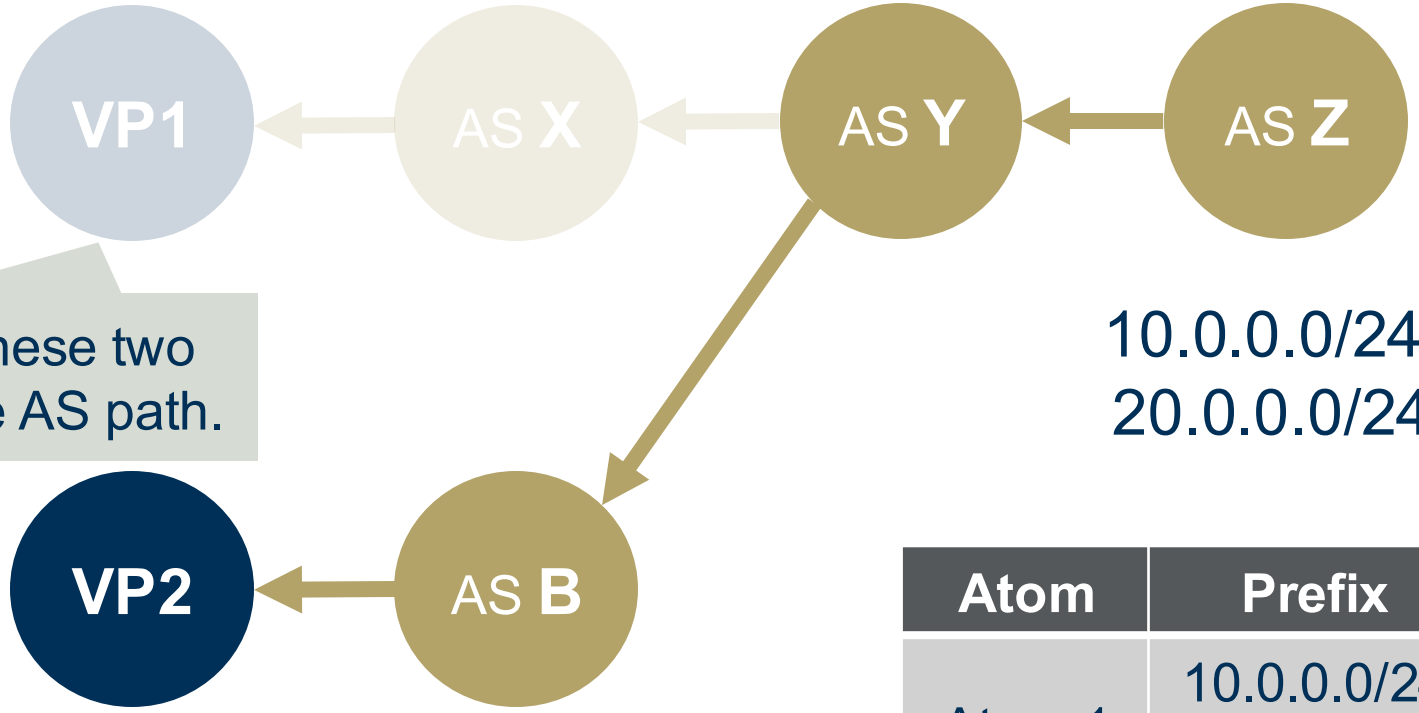
This **VP1** observes these two prefixes with the same AS path.

10.0.0.0/24
20.0.0.0/24

Georgia Tech

What is a policy atom

A policy atom is a set of prefixes that share the same AS paths within each vantage point



This **VP1** observes these two prefixes with the same AS path.

This **VP2** observes these two prefixes with the same AS path.

Atom	Prefix
Atom 1	10.0.0.0/24
	20.0.0.0/24

Georgia Tech

Why policy atoms matter

- Policy atoms is prevalent [IMW 2002]
- Prefixes within atoms stick together across BGP updates [IMW 2002]

- Applications
  - Traffic engineering detection [PAM 2025, Darwich et al.]
  - Routing behavior modeling [MACS 2021, Singh et al.; FGCS 2019, Tian et al.]
  - Measurement targets selection [NSDI 2008, Mahajan et al.; OSDI 2006, Madhyastha et al.]

Our replication study

- Reproduced the 2002 analysis to validate our methodology
- Replicated for the past two decades (2004-2024)
- Investigated where in the topology do atoms split
- Extended this study to IPv6

Prefixes within atoms stick together across BGP updates

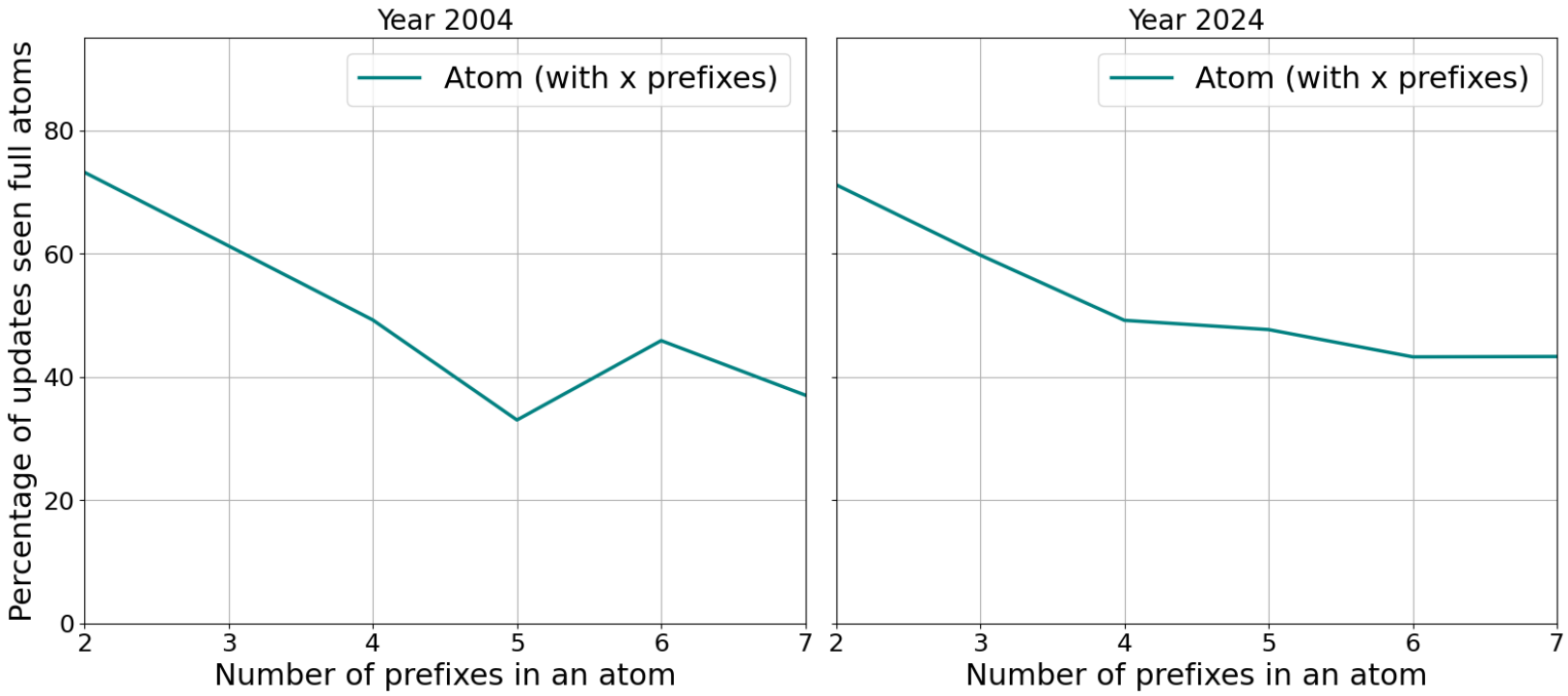
We calculate the fraction of related updates that contain all the prefixes of that atom:

For atom **a** of size **k**

$$\frac{\sum_{a:|a|=k} \#\ \text{UPDATES containing all prefixes in } \mathbf{a}}{\sum_{a:|a|=k} \#\ \text{UPDATES containing at least one prefix in } \mathbf{a}}$$

Prefixes within atoms stick together across BGP updates

- Most updates contain all prefixes of an atom



For atoms of size [2, 3, 4, 5, 6, 7, …]

$$\frac{\sum_{a:|a|=k} \#\ \text{UPDATES containing all prefixes in } \mathbf{a}}{\sum_{a:|a|=k} \#\ \text{UPDATES containing at least one prefix in } \mathbf{a}}$$

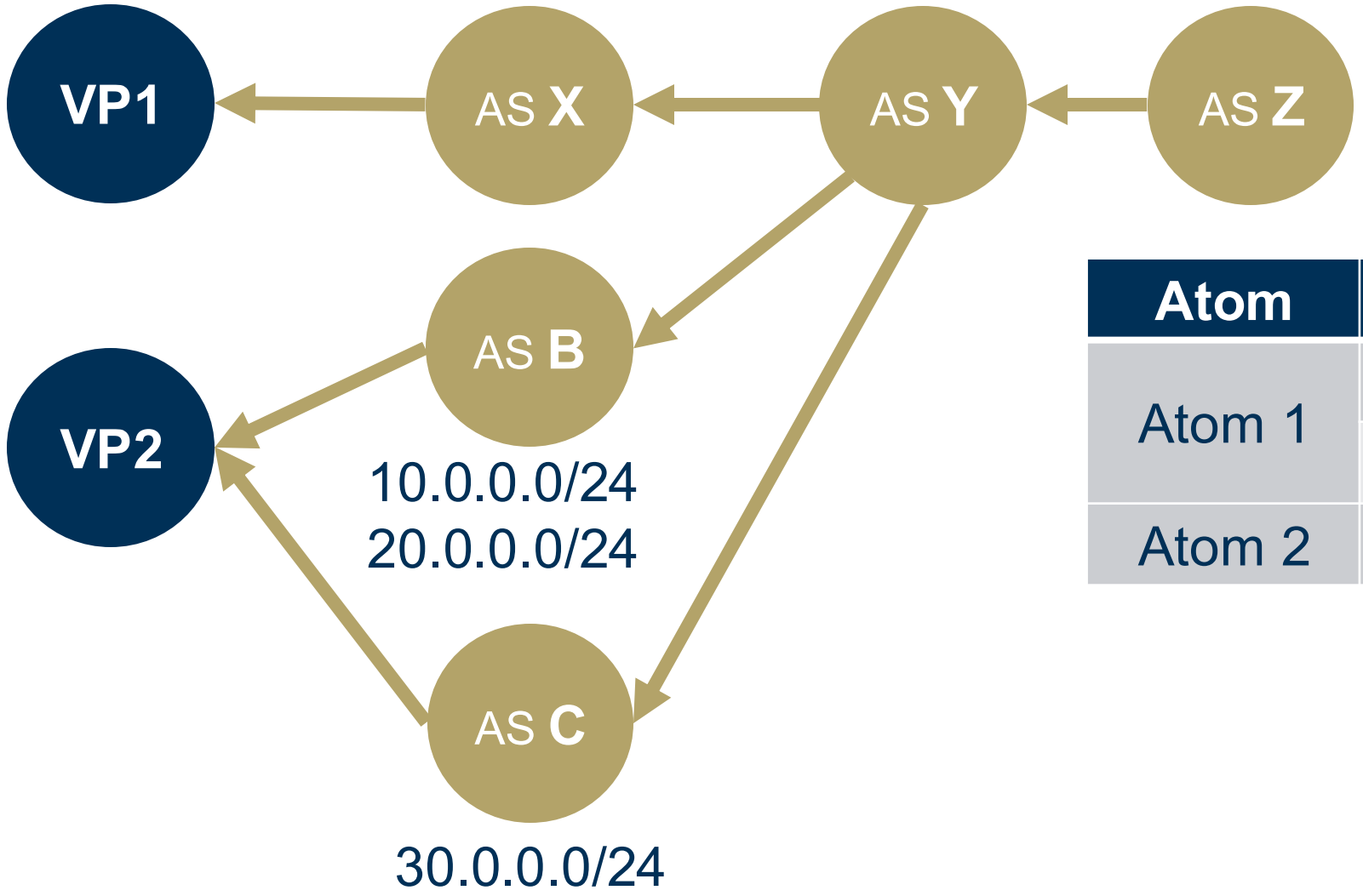
Georgia Tech

How far from the origin prefixes split into separate atoms

- Are policy atoms created due to routing policies of **origin** or **transit** ASes?

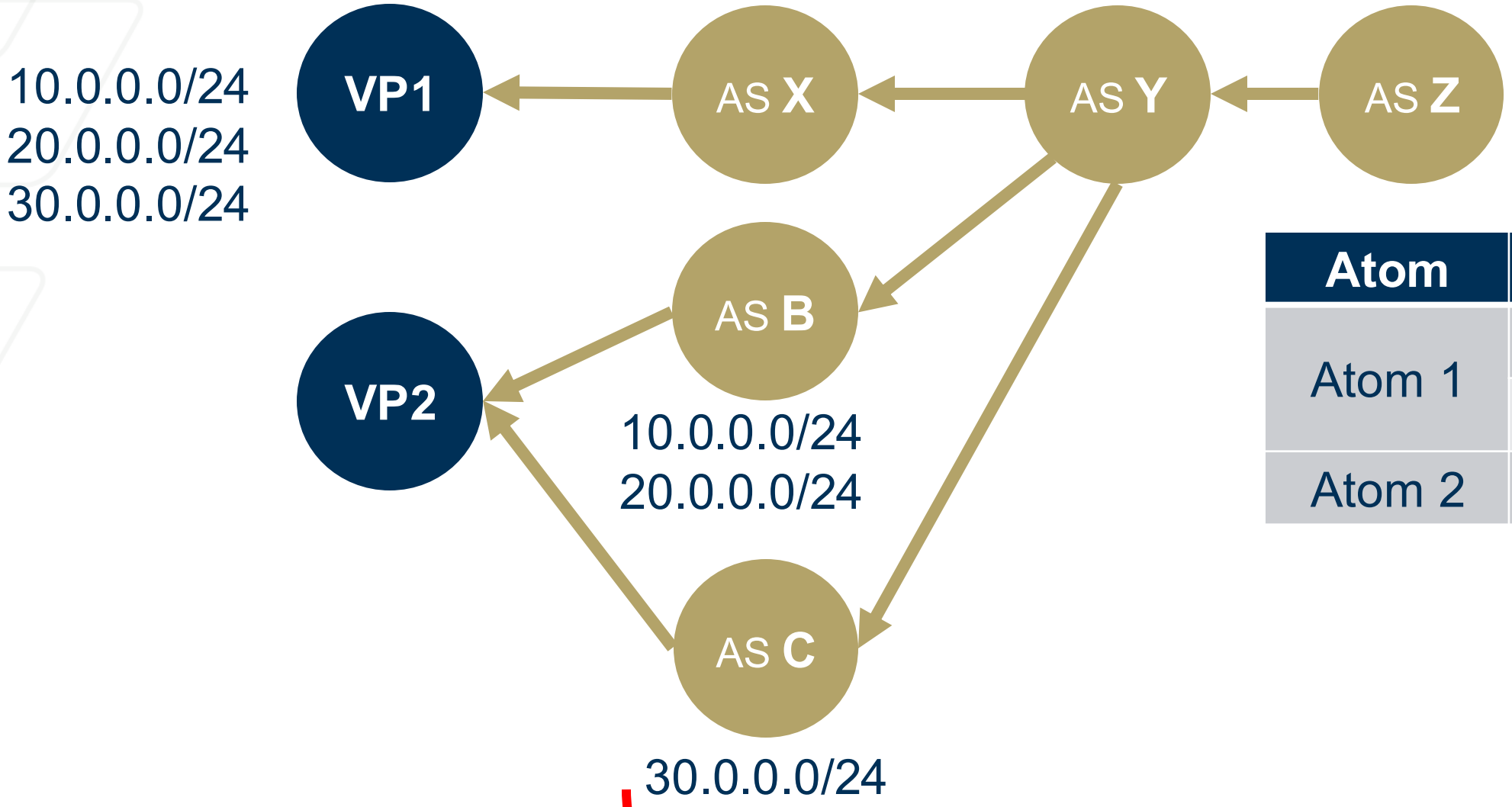
Example of prefixes splitting into 2 atoms

10.0.0.0/24
20.0.0.0/24
30.0.0.0/24



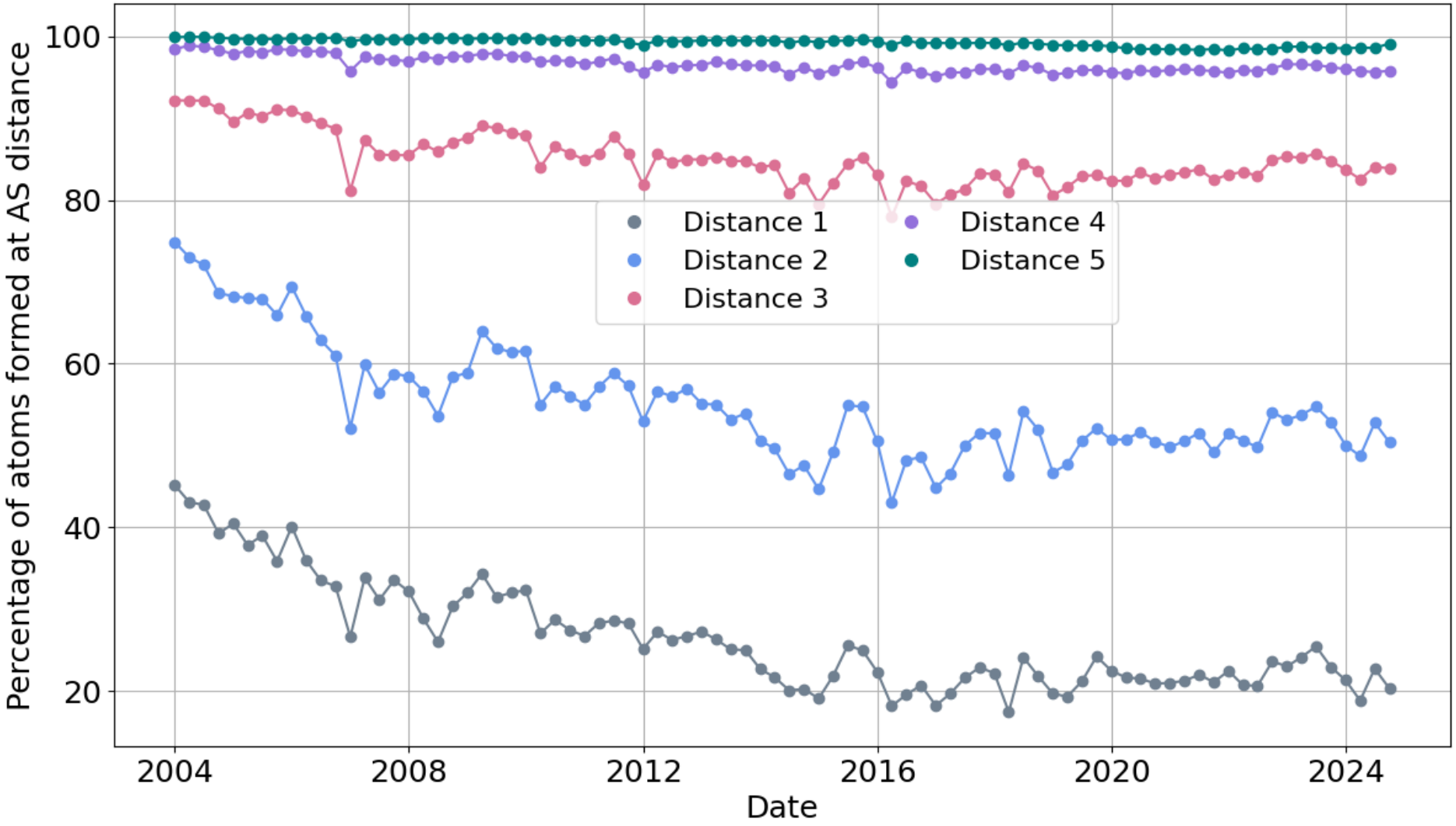
Atom	Prefix
Atom 1	10.0.0.0/24
	20.0.0.0/24
Atom 2	30.0.0.0/24

Example of prefixes splitting into 2 atoms



Atom	Prefix
Atom 1	10.0.0.0/24
	20.0.0.0/24
Atom 2	30.0.0.0/24

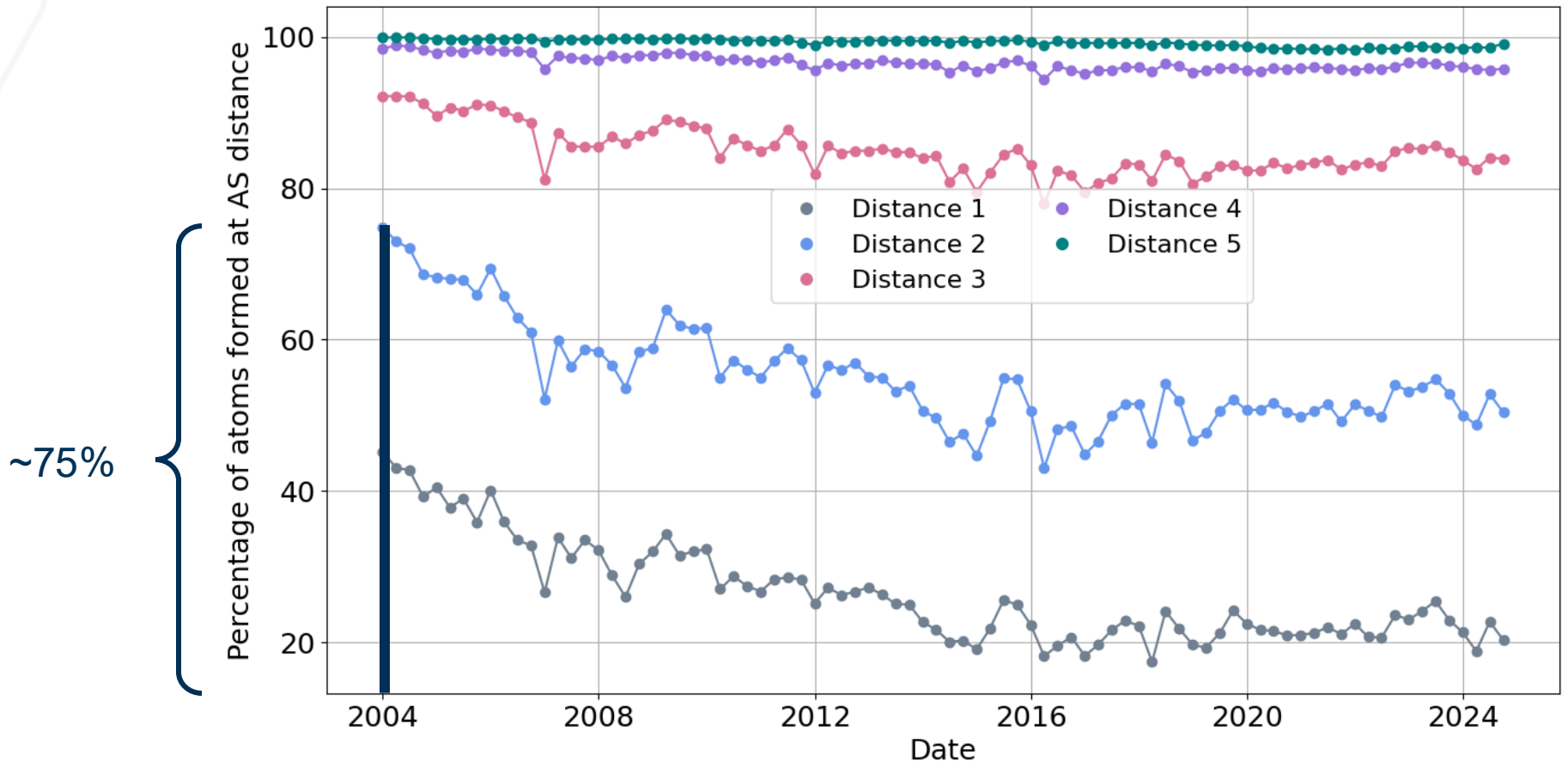
In 2024 policy atoms split further away from the origin



New

Georgia Tech

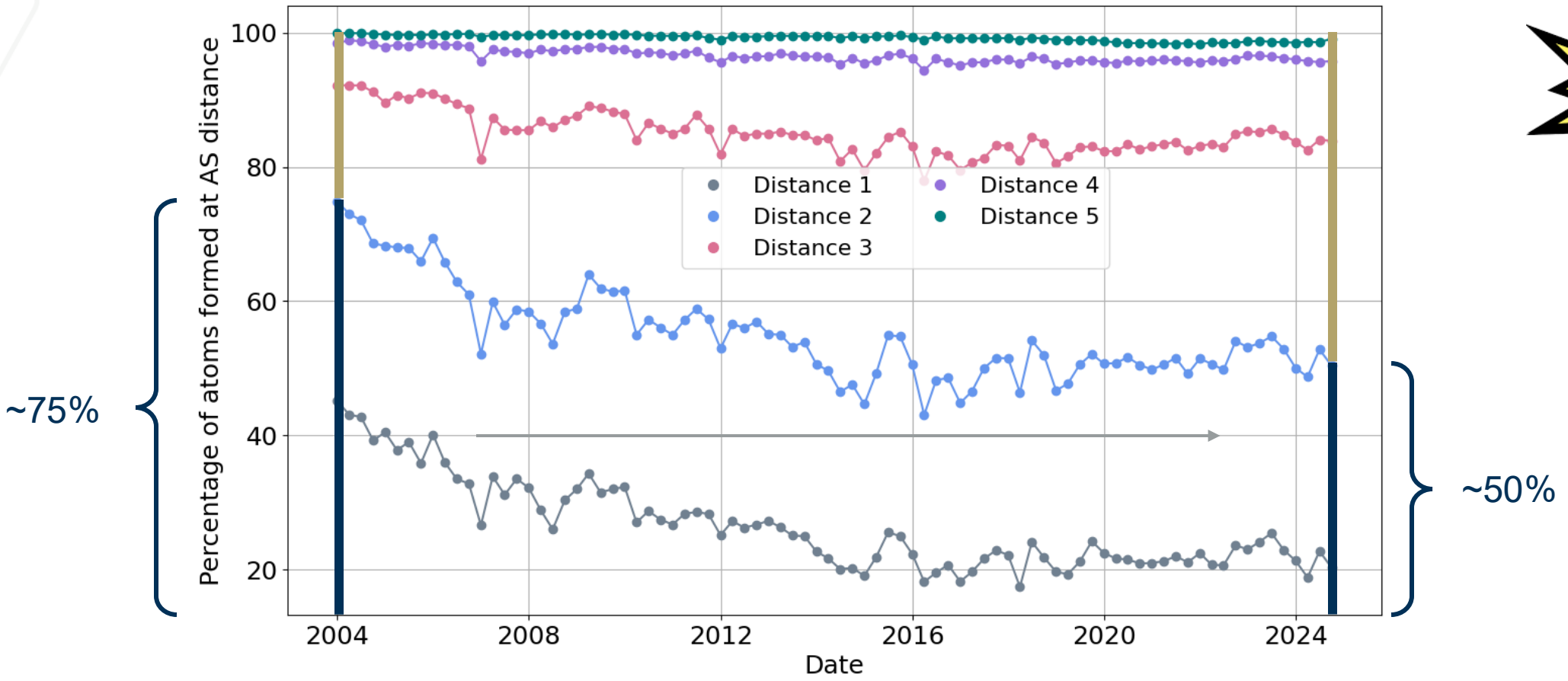
In 2024 policy atoms split further away from the origin



- In 2004, the majority of atoms were formed due to the routing policy of the origin AS

Georgia Tech

In 2024 policy atoms split further away from the origin



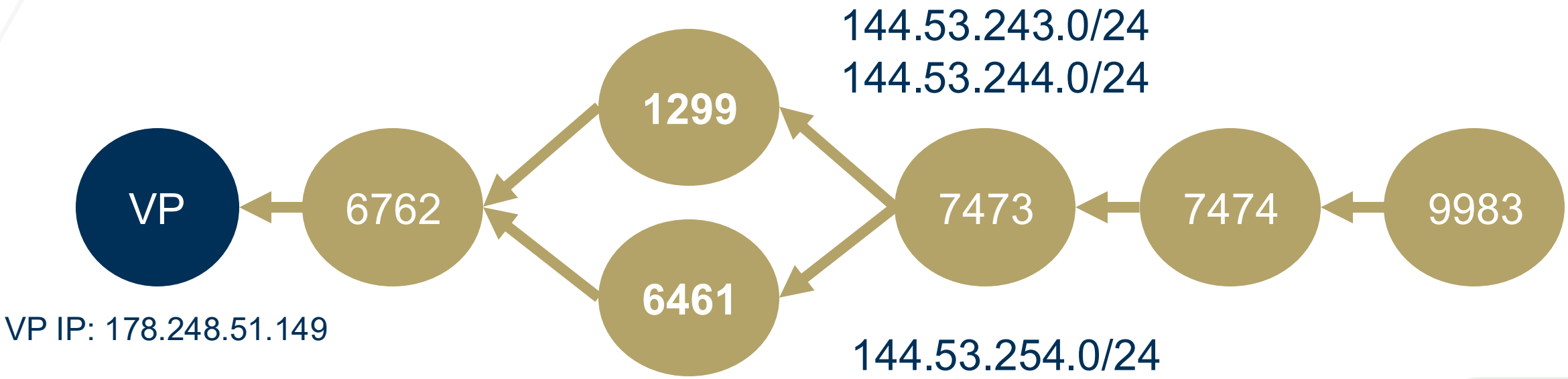
- In 2024, there are more atoms formed further away from the origin AS
- The majority of atoms are not formed due to the routing policy of the origin AS

Georgia Tech

In 2024 policy atoms split further away from the origin

- Most relevant cause: **Transit ASes**
  - selective-export or community-based policies

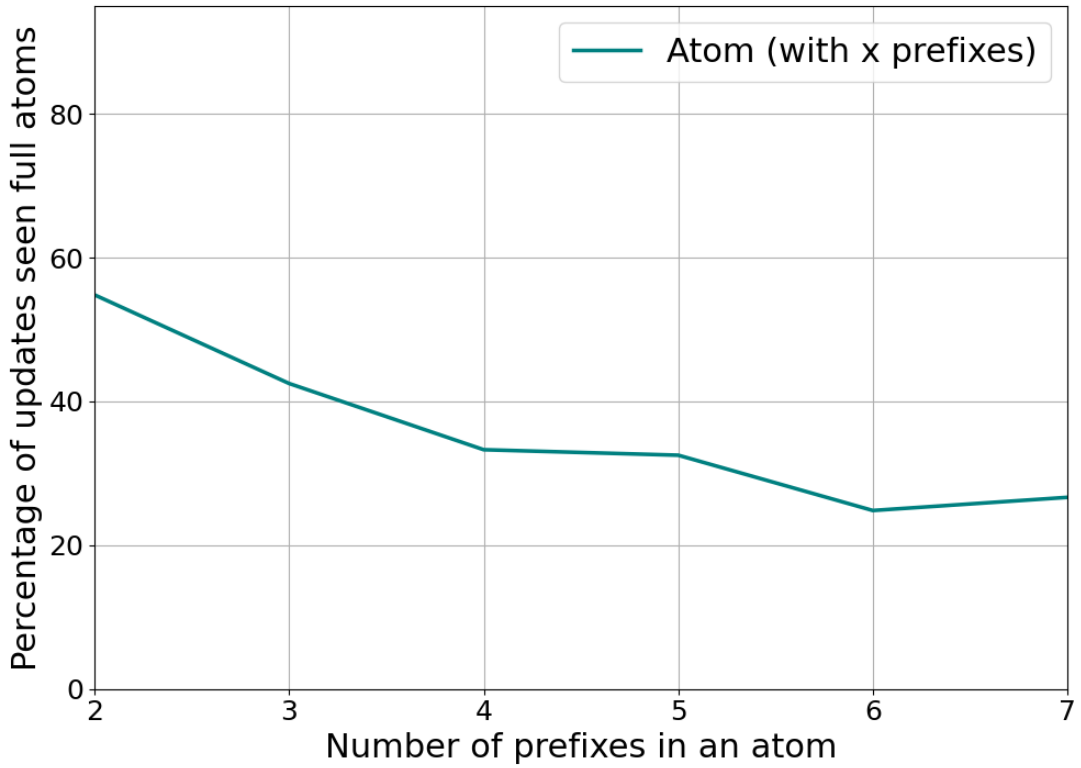
- Example (AS9983):

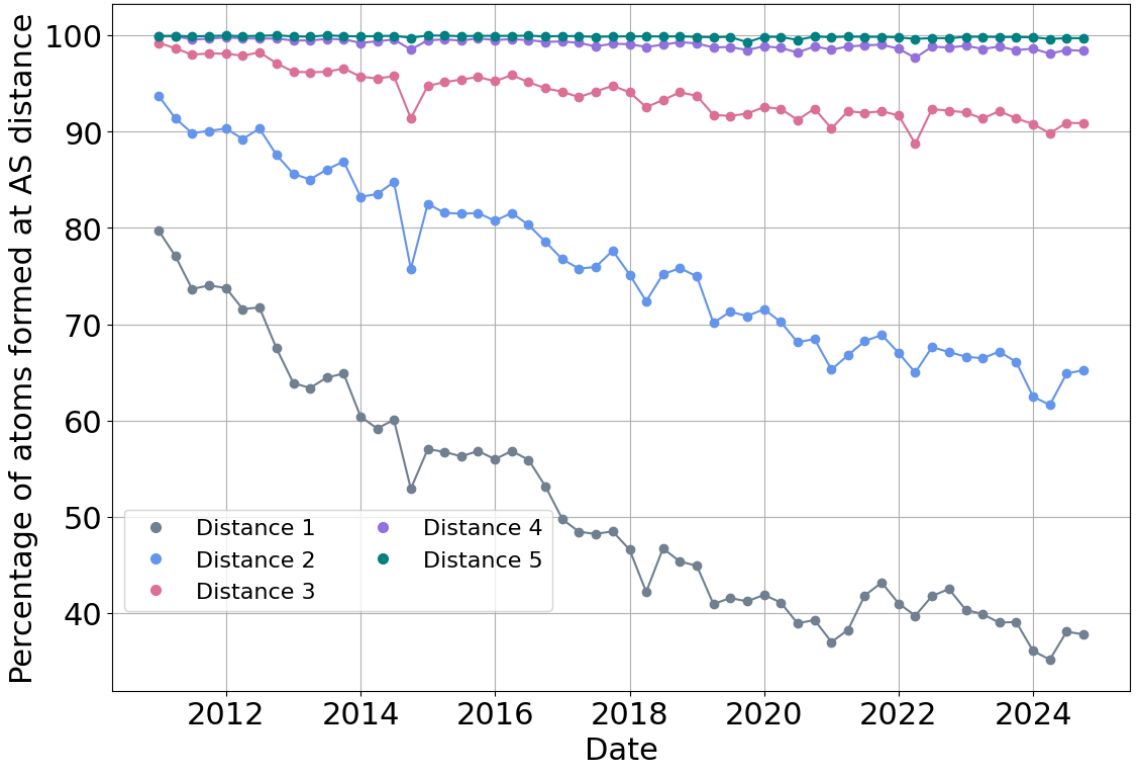


Georgia Tech

IPv6 shows similar trends as in IPv4

- An atom's prefixes tend to appear together in BGP updates
- Policy atoms formed further away from the origin AS





Georgia Tech

Final takeaways

- Policy atoms stand the test of time
  - Also in IPv6
- The atom splits tend to be further from the origin
- Future work:
  - Leverage policy atoms to reveal where networks apply consistent routing policies across v4 and v6