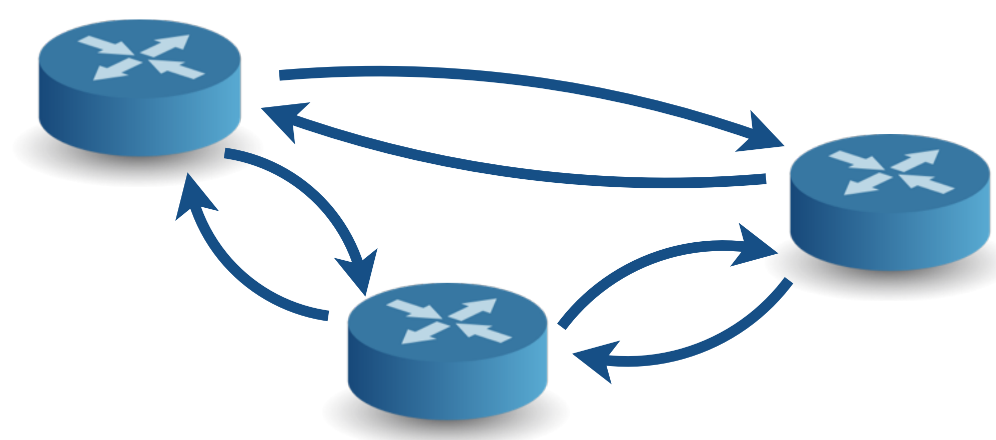
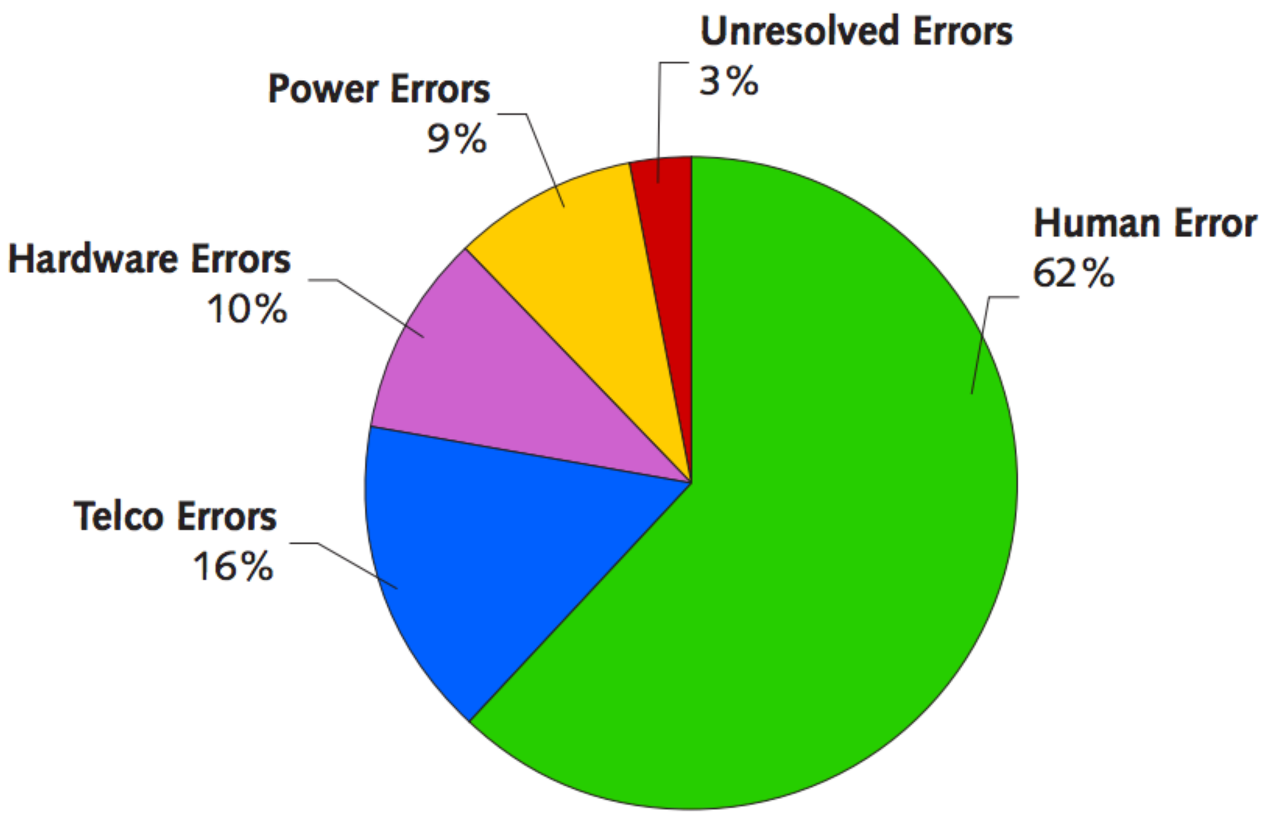
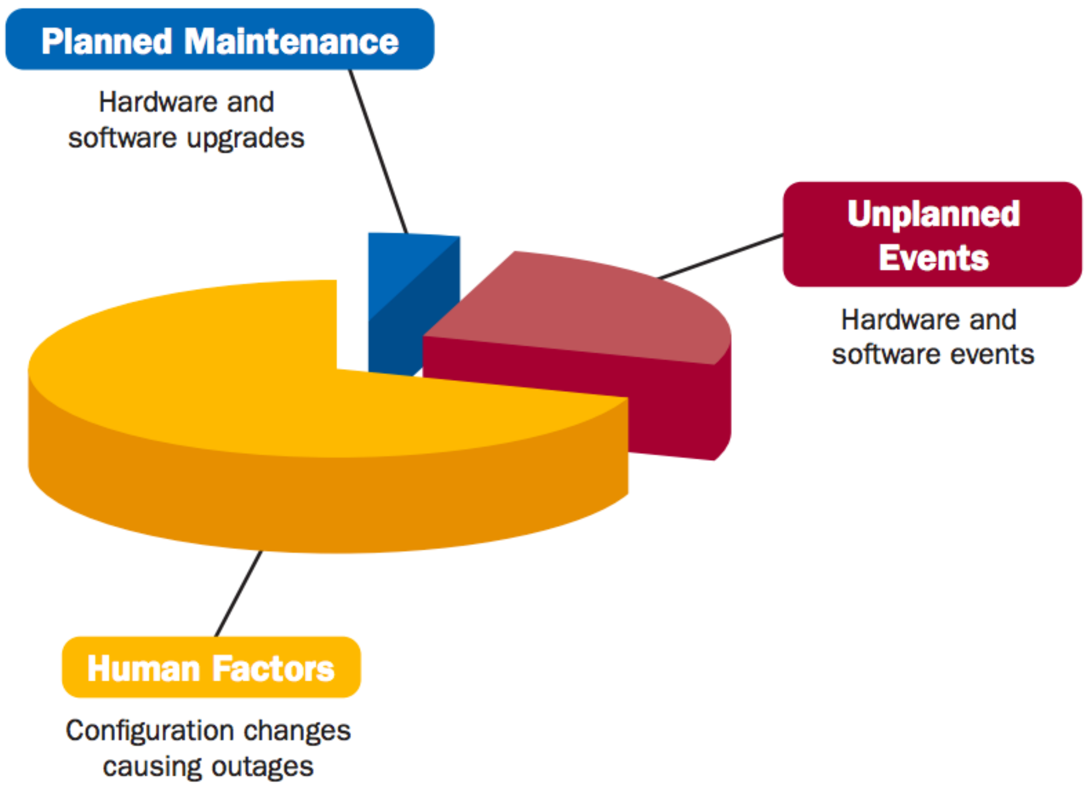
Don’t Mind the Gap:
Bridging Network-wide Objectives and Device-level Configurations

Ryan Beckett (Princeton, MSR)
Ratul Mahajan (MSR)
Todd Millstein (UCLA)
Jitu Padhye (MSR)
David Walker (Princeton)



Configuring Networks is Error-Prone





~60% of network downtime is caused by human error

-Yankee group 2002

50-80% of outages from configuration changes

-Juniper 2008

Configuring Networks is Error-Prone

Sign In | Register

NETWORKWORLD

YouTube/Pakistan incident: Could something similar whack your site?

Configuring BGP properly is key to avoidance, 'Net registry official says

By Carolyn Duffy Marsan
Network World | Mar 10, 2008 1:00 AM PT

In light of Pakistan Telecom/YouTube incident, Internet registry official explains how you can avoid having your web site victimized by such an attack.

When Pakistan Telecom blocked YouTube's traffic one Sunday evening in February, the ISP created an international incident that wreaked havoc on the popular video site for more than two hours.

RIPE NCC, the European registry for Internet addresses, has conducted an analysis of what happened during Pakistan Telecom's hijacking of YouTube's traffic and the steps that YouTube took to stop the attack.

We posed some questions to RIPE NCC's Chief Scientist Daniel Karrenberg about the YouTube incident. Here's what he had to say:

How frequently do hijacking incidents like the Pakistan Telecom/YouTube incident happen?

Misconfigurations of iBGP (internal BGP, the protocol used between the routers in the same Autonomous System) happen regularly and are usually the result of an error. One such misconfiguration caused the Pakistan Telecom/YouTube incident. It appears that the Pakistan Telecom/YouTube incident was not an "attack" as some have labeled it, but a configuration error. (See Columnist Johna Till Johnson's take on the topic.)

What is significant about the YouTube incident?

Configuring Networks is Error-Prone

NETWORKWOR

YouTube/Pakistan i your site?

Configuring BGP prope says

By Carolyn Duffy Marsan
Network World | Mar 10, 2008 1:00 AM PT

In light of Pakistan Telecom/YouTu your web site victimized by such an

When Pakistan Telecom blocked Y international incident that wreake

RIPE NCC, the European registry fo during Pakistan Telecom's hijackir attack.

We posed some questions to RIPE Here's what he had to say:

How frequently do hijacking inci

Misconfigurations of iBGP (interna Autonomous System) happen regu misconfiguration caused the Pakis Telecom/YouTube incident was no Columnist Johna Till Johnson's tal

What is significant about the You

(https://www.thousandeyes.com) Product Solutions Customers (https://www.thousandeyes.com/customers) Resources About Login (https://app.thousandeyes.com) (https://www.thousandeyes.com/search) Sign Up (https://www.thousandeyes.com/signup)

← Blog Home (/)

Time Warner Cable Outage Causes Widespread Routing and DNS Impacts

Posted by Pete Anderson (https://blog.thousandeyes.com/author/pete/) on August 28, 2014

By now a lot of you have probably read about the Time Warner Cable (TWC) outage on August 27th. Yesterday morning I was greeted with a slew of alarms, names that wouldn't resolve, websites that wouldn't load and home office employees without any Internet access. It hadn't hit the news yet, but I could sense that a major outage was occurring and quickly opened up the ThousandEyes platform to get a handle on the situation.

Time Warner Outage

The alerts started coming in a little before 930 UTC (5:30 Eastern). I observed several different issues including inaccessible websites, DNS names failing to resolve, BGP reachability issues and agents losing access to the Internet. Companies that peer with Time Warner experienced degraded HTTP availability, affecting critical services such as supply chain portals (Figure 1).

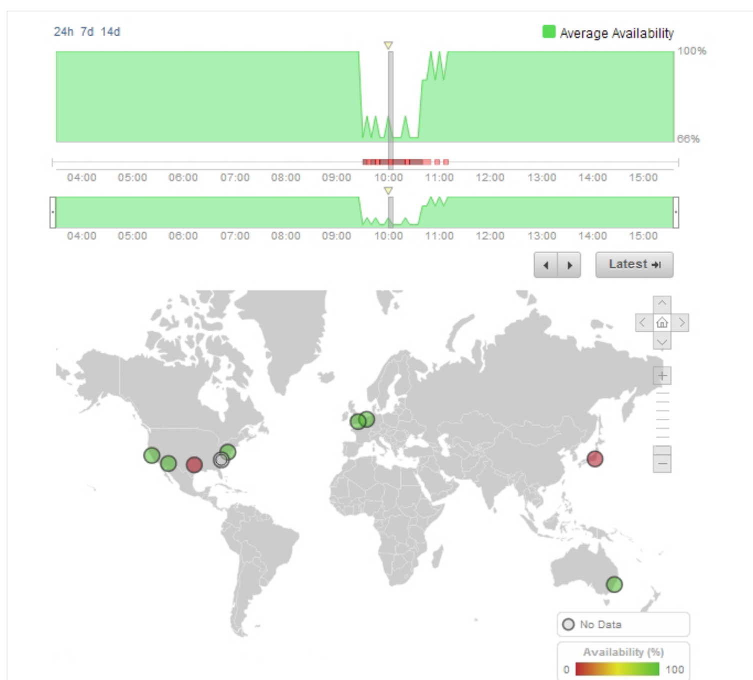


Figure 1: Supply chain portal with limited availability during Time Warner Cable outage.

I could see right away that users and networks that connect through Time Warner were unable to reach the supply chain portal, indicating the issue was in the Time Warner network. In this case, I'd expect to see a brief service interruption while all the traffic re-routed through their other upstream ISP AT&T. However, the availability issues continued for the entire duration of the outage. I was surprised to still see issues, so I took a look at the path visualization view to figure out exactly where traffic was getting dropped on the way to this site. Normally, two locations (Tokyo and Dallas) transit Road Runner (Time Warner) to reach this supply chain portal, while the rest go through AT&T (Figure 2).

Configuring Networks is Error-Prone

NETWORKWOR

YouTube/Pakistan i your site?

Configuring BGP proper says

By Carolyn Duffy Marsan

Network World | Mar 10, 2008 1:00 AM PT

In light of Pakistan Telecom/YouTub your web site victimized by such an

When Pakistan Telecom blocked Y international incident that wreake

RIPE NCC, the European registry fo during Pakistan Telecom's hijackin attack.

We posed some questions to RIPE Here's what he had to say:

How frequently do hijacking inci

Misconfigurations of iBGP (internal Autonomous System) happen regu misconfiguration caused the Pakis Telecom/YouTube incident was no Columnist Johna Till Johnson's tal

What is significant about the You

(https://www.thousandeyes.com)

← Blog Home (/)

Time Warner Ca Impacts

Posted by Pete Anderson (ht

By now a lot of you have probably re morning I was greeted with a slew o home office employees without any outage was occurring and quickly op

Time Warner Outage

The alerts started coming in a little b inaccessible websites, DNS names f Internet. Companies that peer with T services such as supply chain portal

Figure 1: Supply c

I could see right away that users and supply chain portal, indicating the is service interruption while all the traff availability issues continued for the e look at the path visualization view to Normally, two locations (Tokyo and I while the rest go through AT&T (Figu

2/5/2016 China routing snafu briefly mangles interweb • The Register

Log in | Sign up Cash'n'Carrion | Whitepapers | The Channel | The Next Platform

GET YOUR SMOKEY ON LEARN HOW » ONLY YOU CAN PREVENT WILDFIRES.

DATA CENTER SOFTWARE NETWORKS SECURITY INFRASTRUCTURE DEVOPS BUSINESS HARDWARE SCIENCE BOOTNOTES FORUMS

Networks ▸ Broadband

China routing snafu briefly mangles interweb

Cockup, not conspiracy

9 Apr 2010 at 12:24, John Leyden 5 0

Bad routing information sourced from China has disrupted the internet for the second time in a fortnight.

Global BGP (Border Gateway Routing) lookup tables sucked in data from a small ISP called IDC China Telecommunication, apparently accidentally broadcast by state-owned carrier China Telecommunications, IDG reports. ISPs including AT&T, France Telcom, Level3, Deutsche Telekom, Qwest and Telefonica accepted ill-thought out traffic routes as a result of the incident.

BGP is a core routing protocol which maps options for the best available routes for traffic to flow across the net. Several routing options are normally included. The China BGP incident is the internet routing equivalent of TomTom publishing routes via Shanghai for motorists looking for alternative routes between London and Paris.

IDC China Telecommunication published ill-conceived routes for between 32,000 and 37,000 networks - about 10 per cent of the net - instead of the normal 40 or so routes, and this information was taken as viable routing options by many service providers for about 20 minutes early on Thursday morning (US time) after China Telecommunications republished it and before the mix-up was resolved. Routers in Asia would have been more likely to adopt the false routes as potentially viable, but effects of the incident were recorded all over the world.

BGPmon.net, a BGP monitoring service, has a detailed technical write-up of the snafu, which it described as a prefix hijack, here.

> **Although it seems they [IDC China Telecommunication] have leaked a whole table, only about 10 per cent of these prefixes propagated outside of the Chinese network. These include prefixes for popular websites such as dell.com, cnn.com, www.amazon.de, www.rapidshare.com and www.geocities.jp.**
> **A large number of networks impacted this morning were actually Chinese networks. These include some popular Chinese website such as www.joy.cn , www.pconline.com.cn , www.huanqiu.com, www.tianya.cn and www.chinaz.com**

A cock-up is suspected, rather than a conspiracy, at least by BGPmon.net.

> **Given the large number of prefixes and short interval I don't believe this is an intentional hijack. Most likely it's because of configuration issue, i.e. fat fingers. But again, this is just speculation.**

The practical consequences of the screw-up are still being assessed but it could have resulted in dropped connections or, worse, traffic routed through unknown systems in China. The mess provides one of the clearest illustrations of the security shortcomings of BGP, a somewhat obscure but nonetheless important network protocol.

The China BGP global routing represents a rare but not unprecedented mix-up in global internet traffic management. For example, just two weeks ago bad routing data resulted in the redirection of Chilean internet traffic through a DNS (Domain Name System) server in China, as explained in a detailed post-mortem by internet monitoring firm Renesys here. Bad BGP routing information from Pakistan caused

More like this: China, Network Security

About the FT

The surest investment you'll make this year

The FT's comprehensive coverage of global business provides the insight and analysis you need to stay one step ahead in 2016 and beyond.

Explore More

The surest investment you'll make this year.

Subscribe & save 33%

Most read

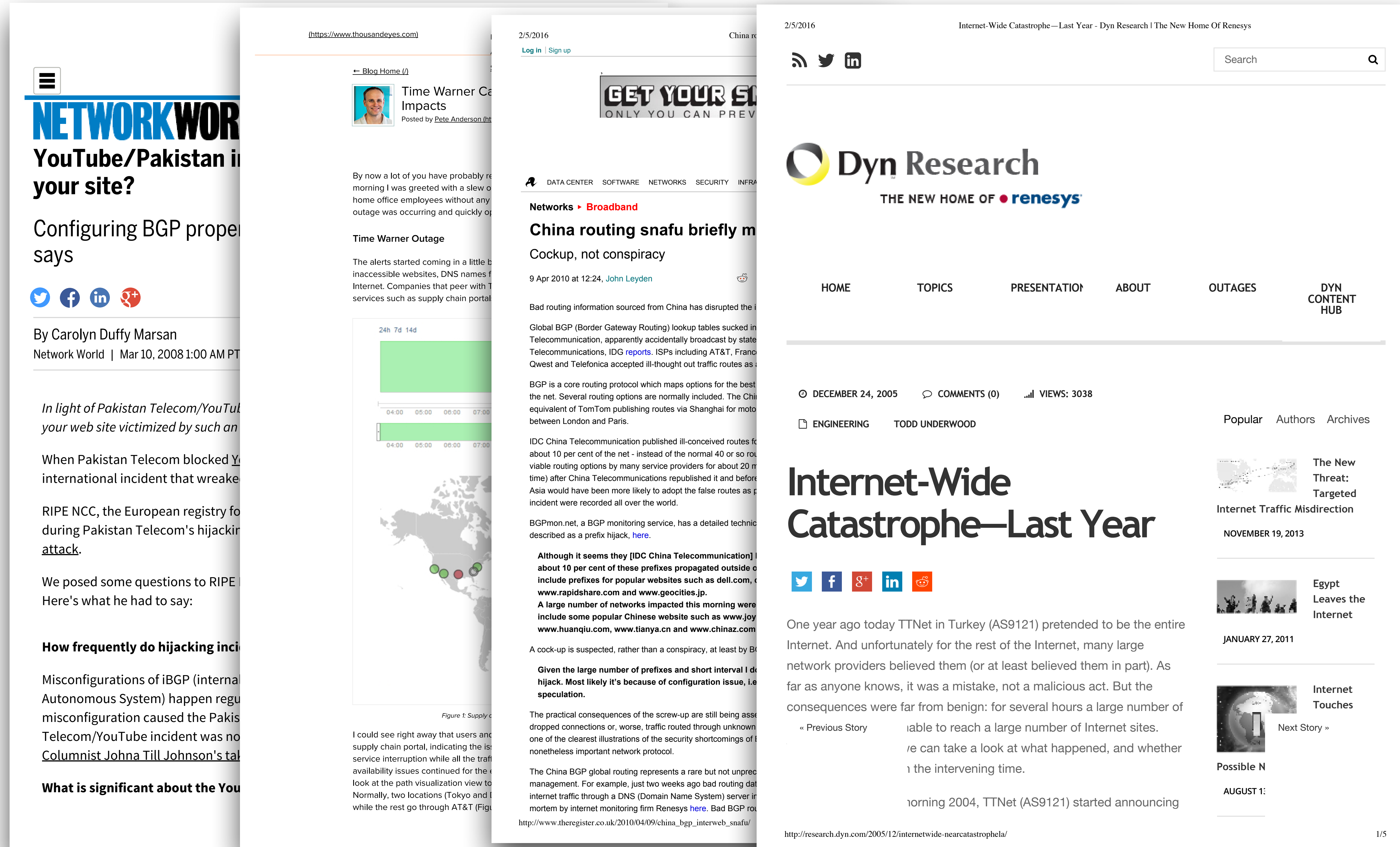
German Chancellor fires hydrogen plasma with the push of a button

Who would code a self-destruct feature into their own web browser? Oh, hello, Apple

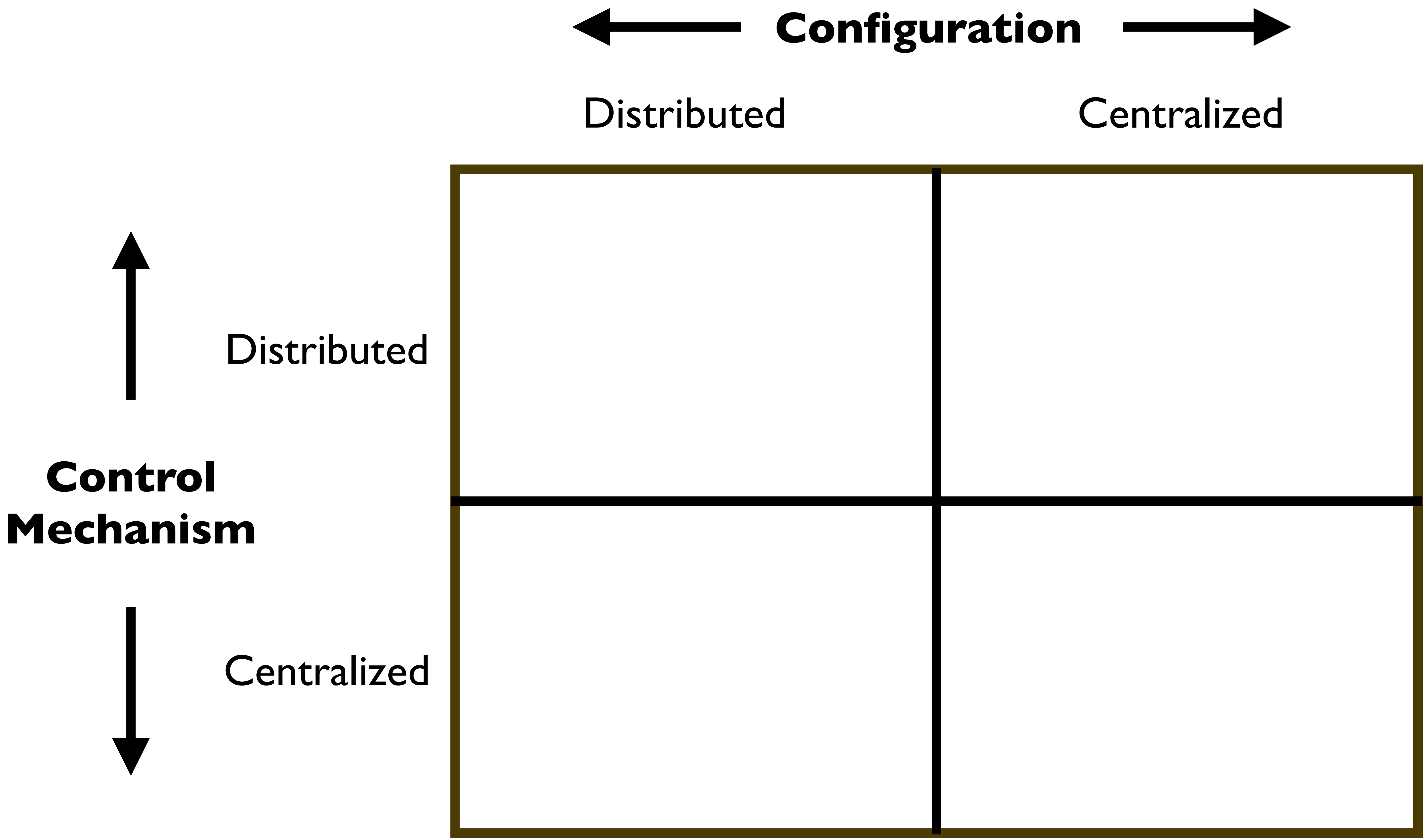
Who wants a quad-core 4.2GHz, 64GB, 5TB SSD

http://www.theregister.co.uk/2010/04/09/china_bgp_interweb_snafu/ 1/4

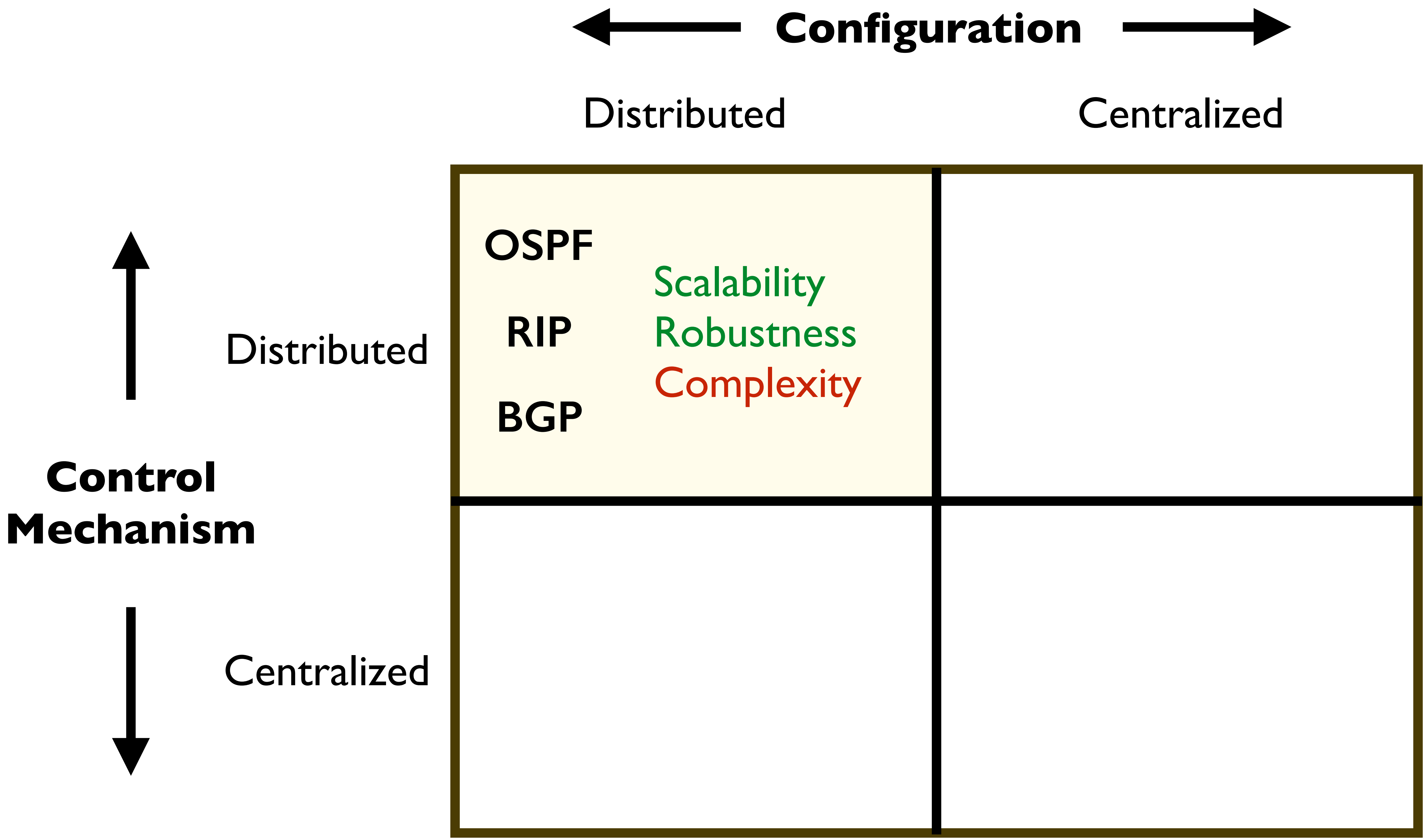
Configuring Networks is Error-Prone



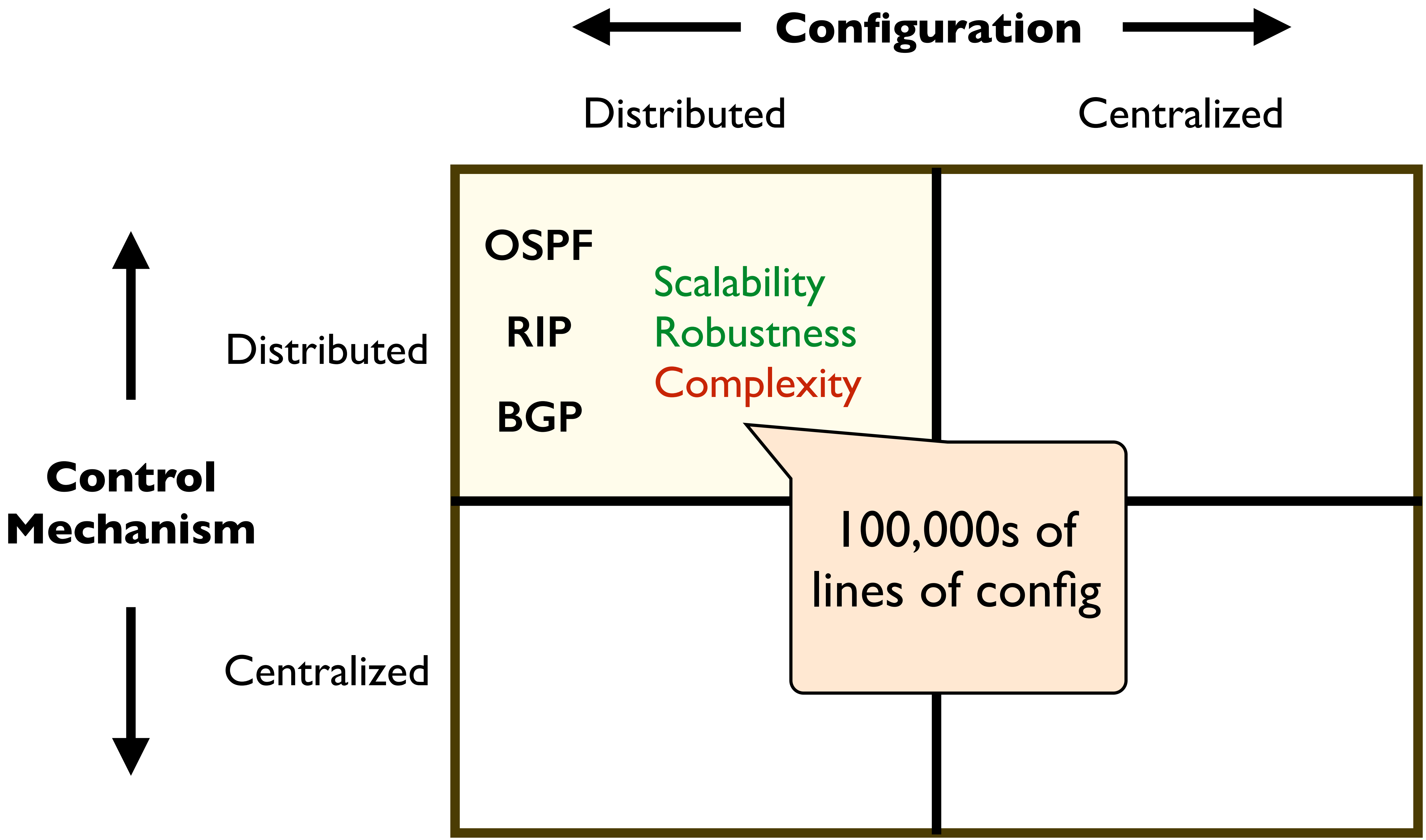
Fundamental Tradeoff?



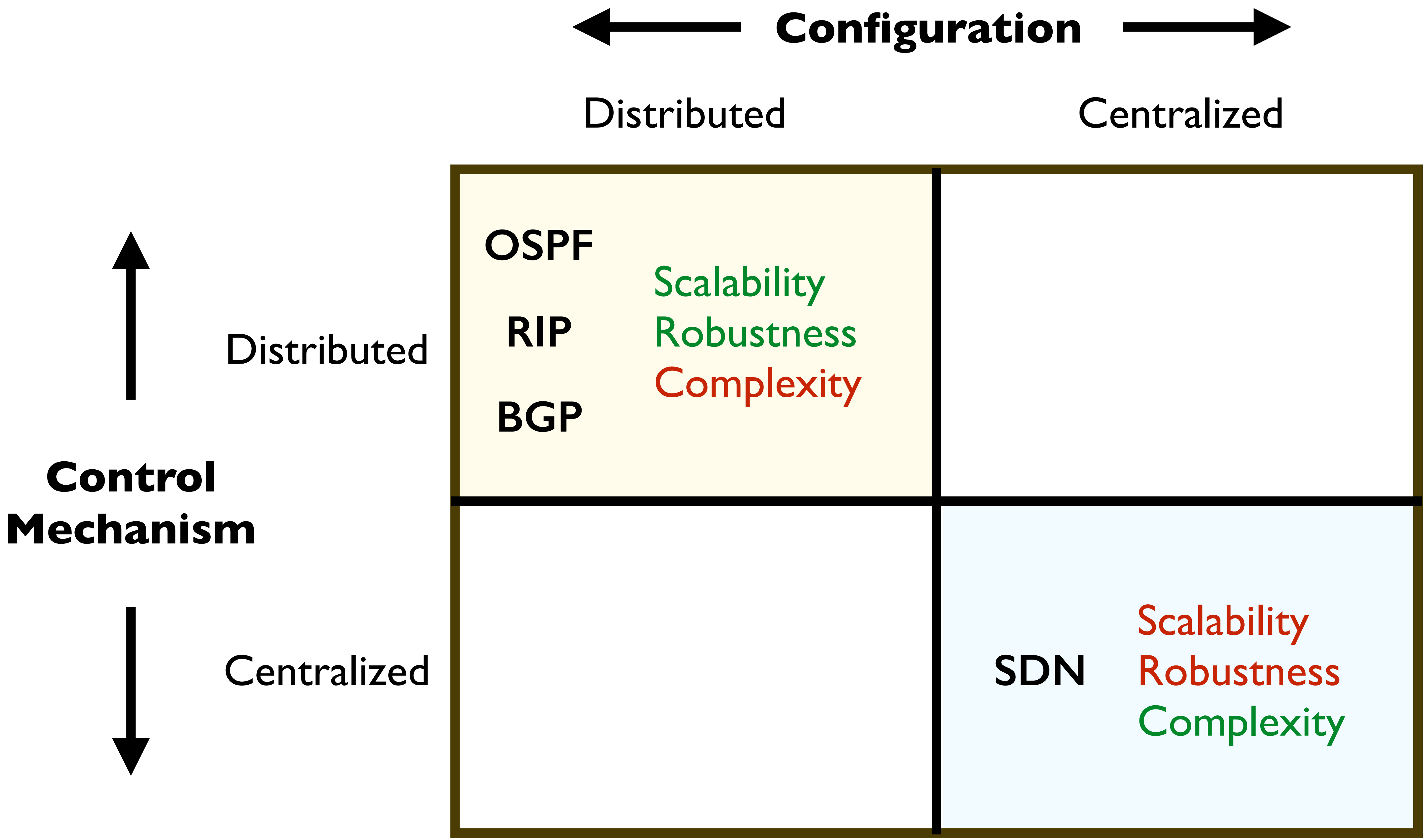
Fundamental Tradeoff?



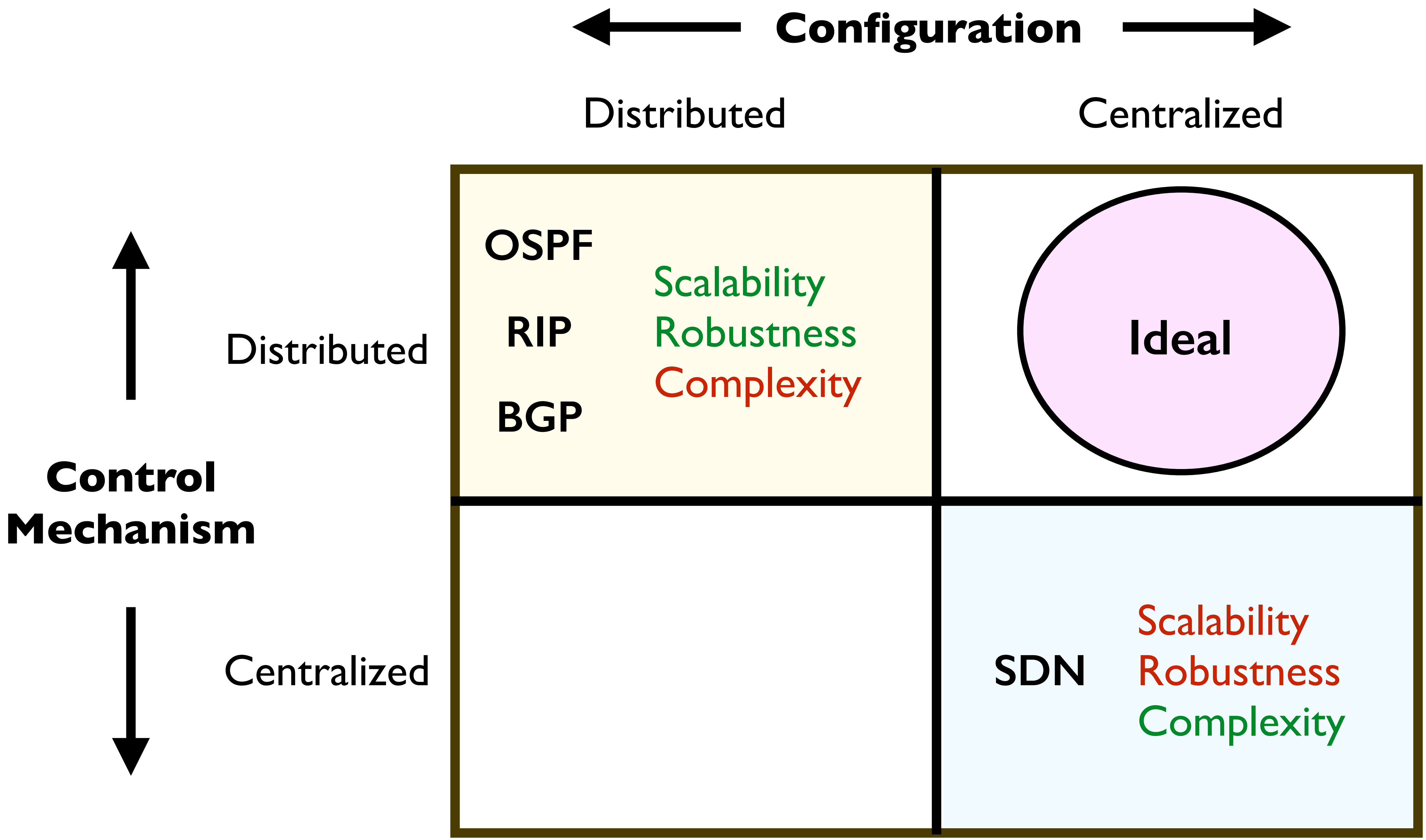
Fundamental Tradeoff?



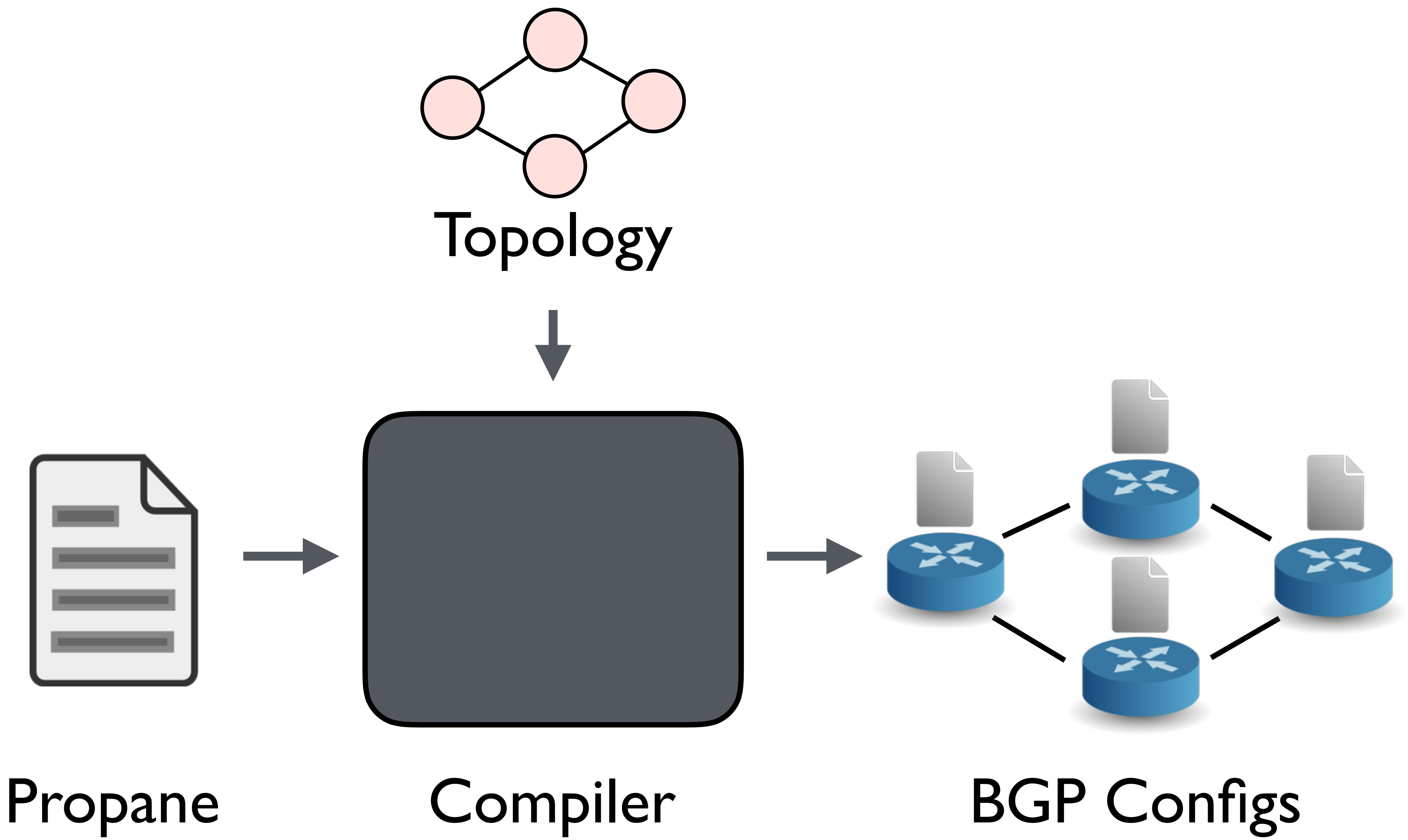
Fundamental Tradeoff?



Fundamental Tradeoff?



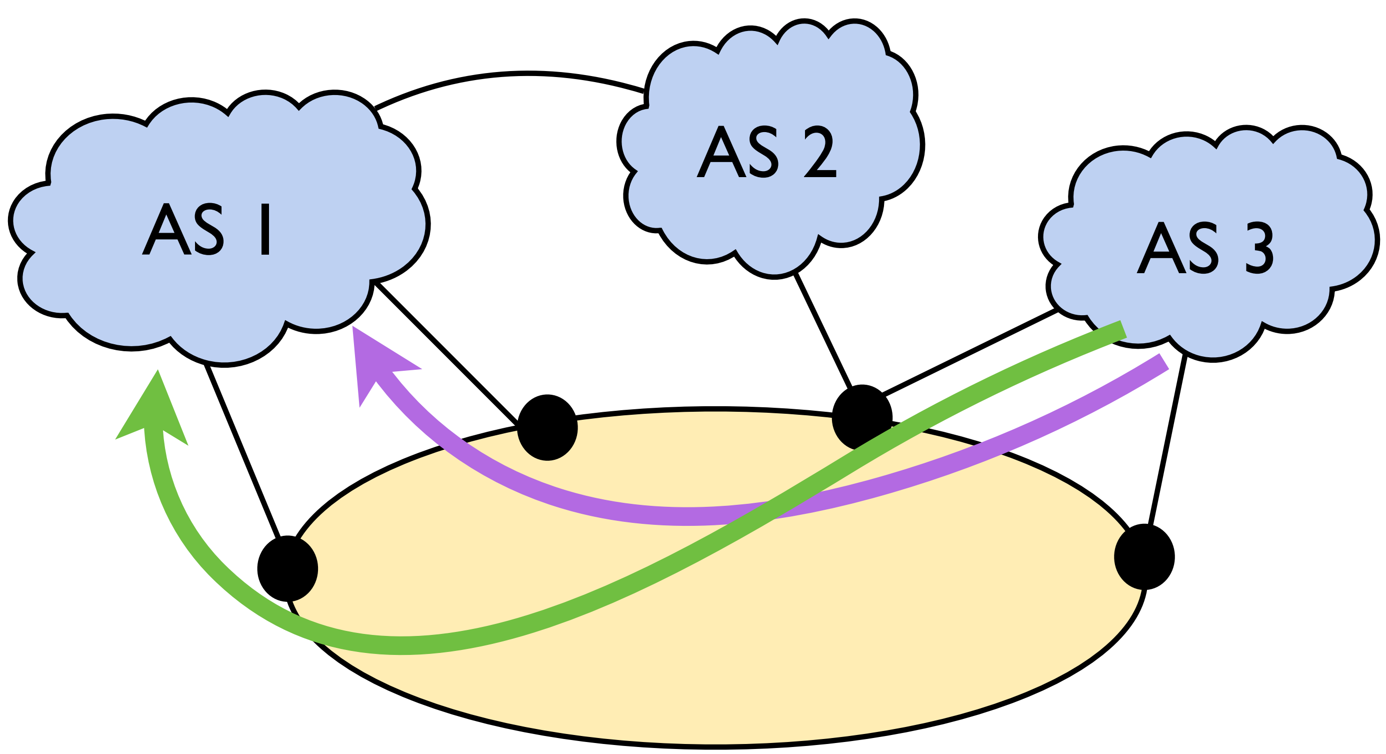
Propane Overview



Propane System

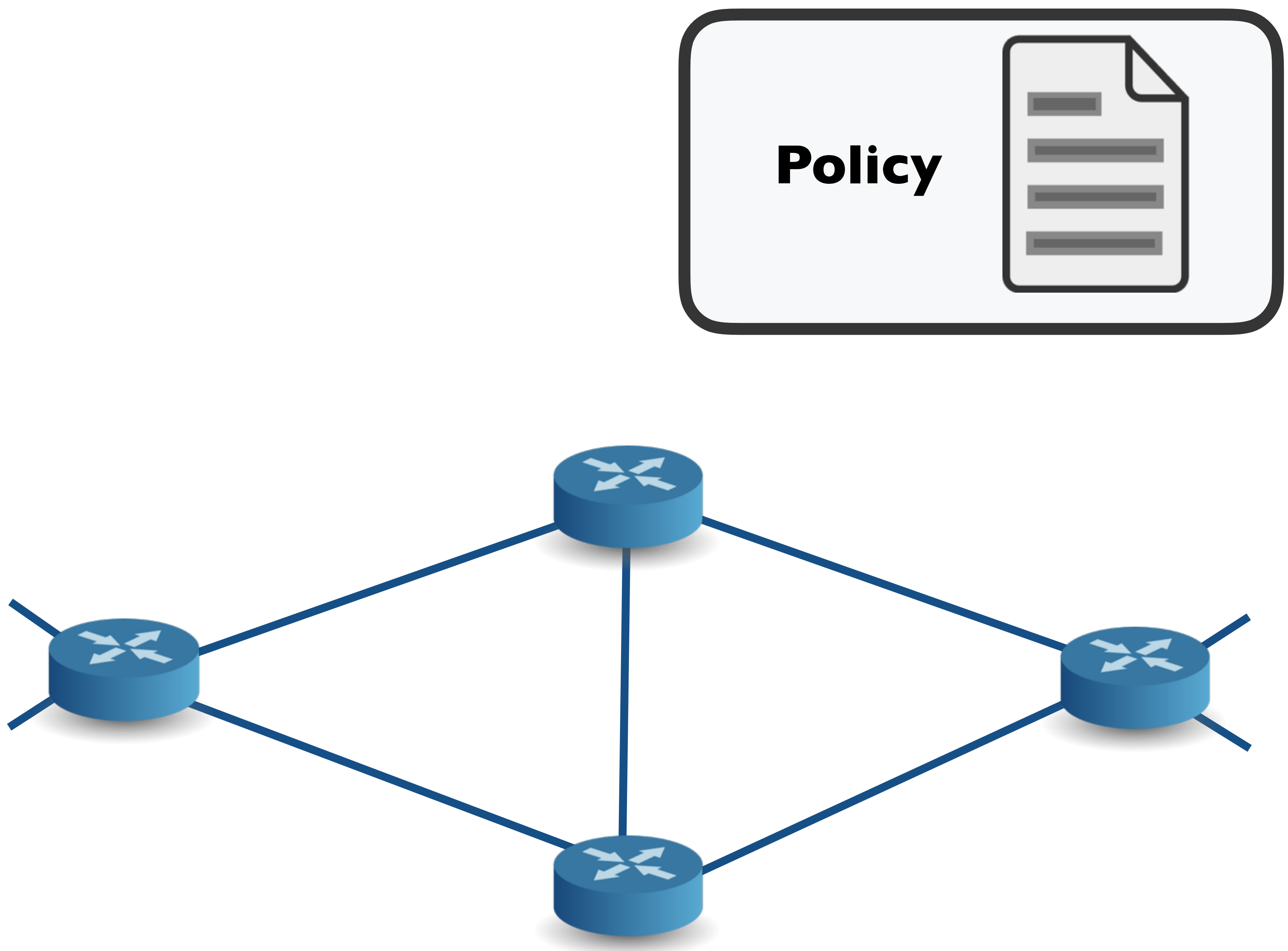
1) Language for expressing network-wide objectives with:

- Path **constraints** and **preferences** in case of failures
- Uniform abstractions for **intra**- and **inter**-domain routing



Propane System

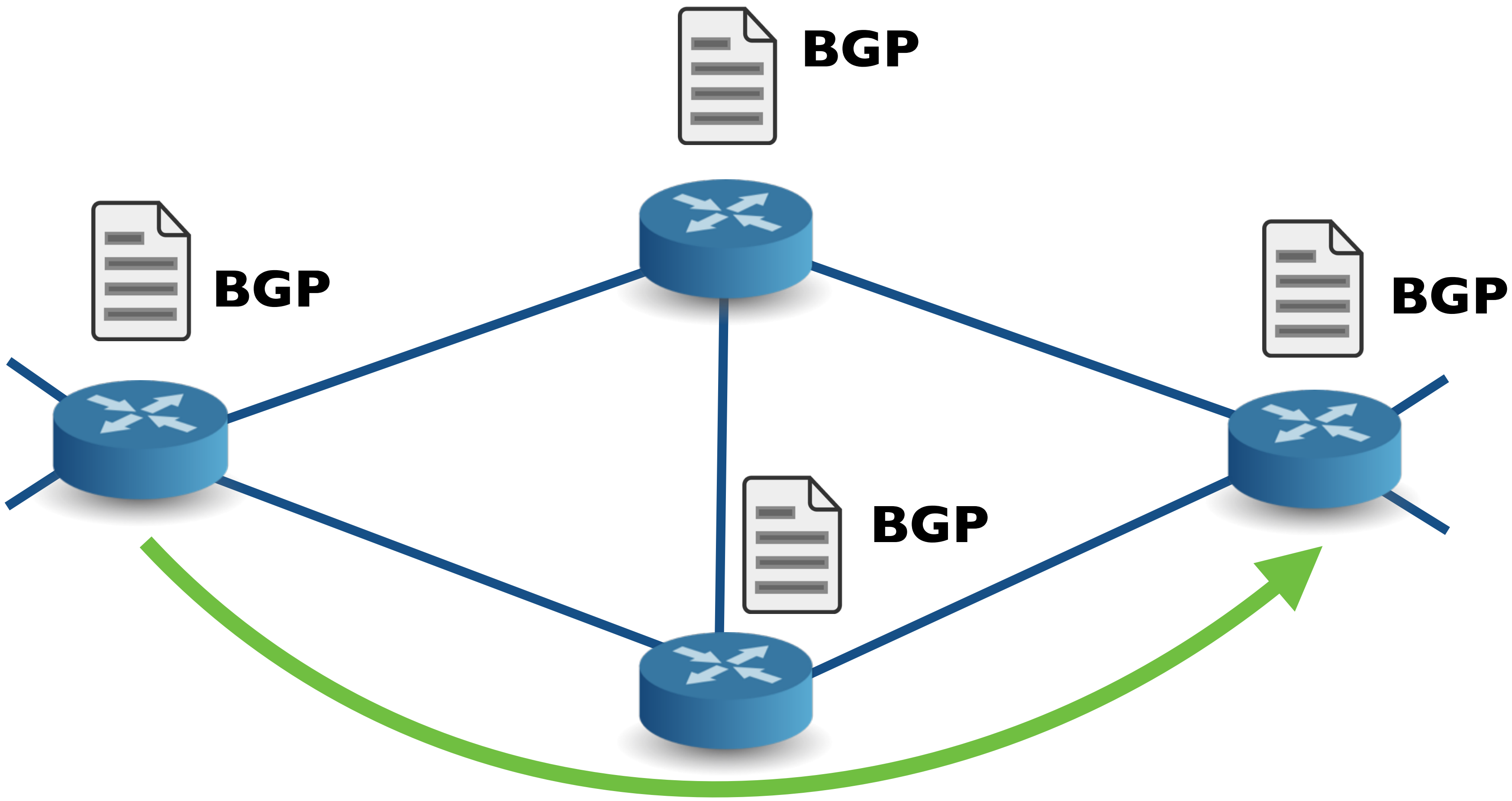
2) Compiler for a purely distributed implementation



Propane System

2) Compiler for a purely distributed implementation

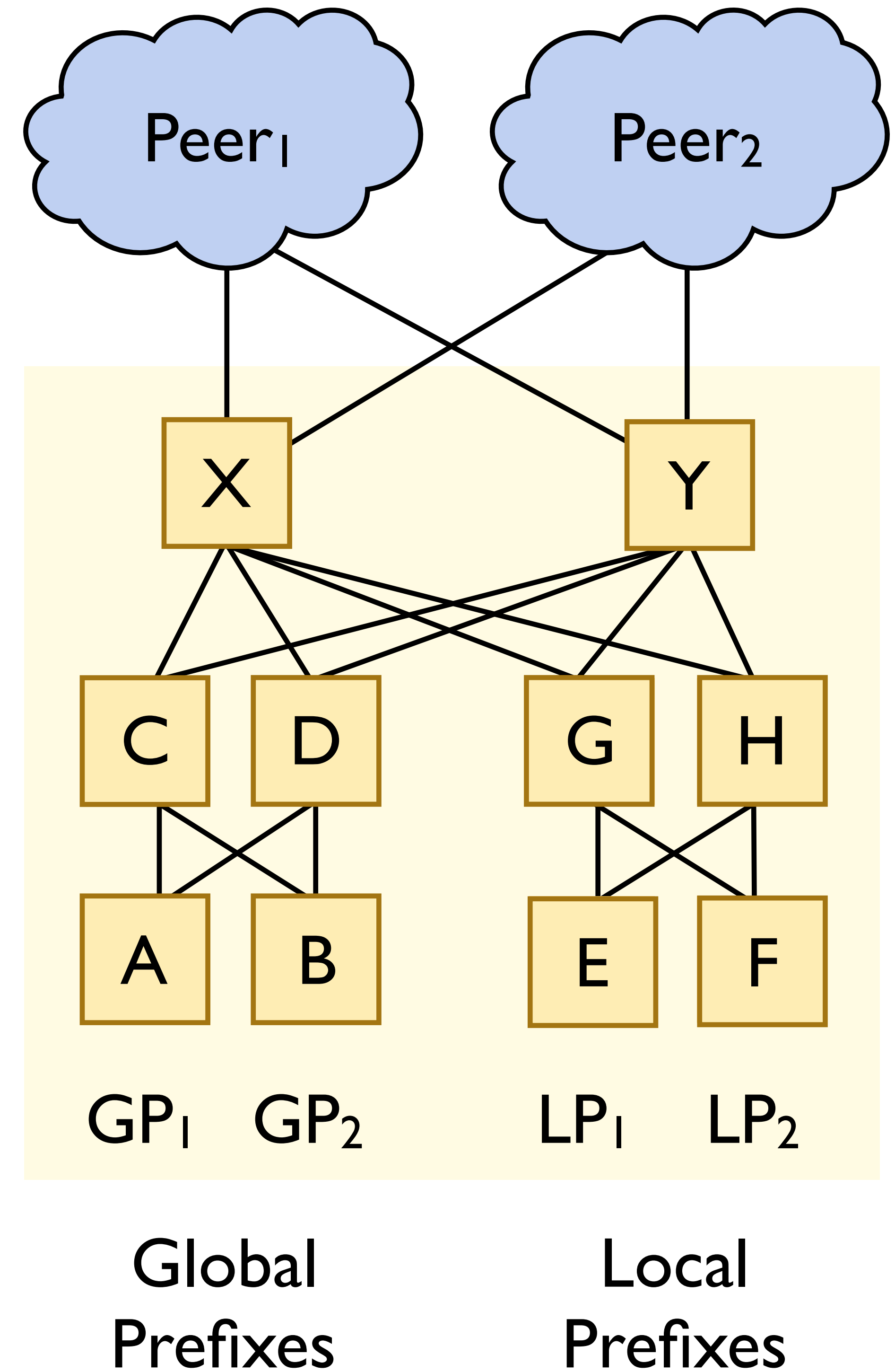
- Generate **BGP** configs for each router
- Compiler guarantees **policy-compliance** for **all failures**



Example: A DC network with traditional configs

Goals

- Local prefixes reachable only internally
- Global prefixes reachable externally
- Aggregate global prefixes as GP
- Prefer leaving through $Peer_1$ over $Peer_2$
- Prevent transit traffic between peers



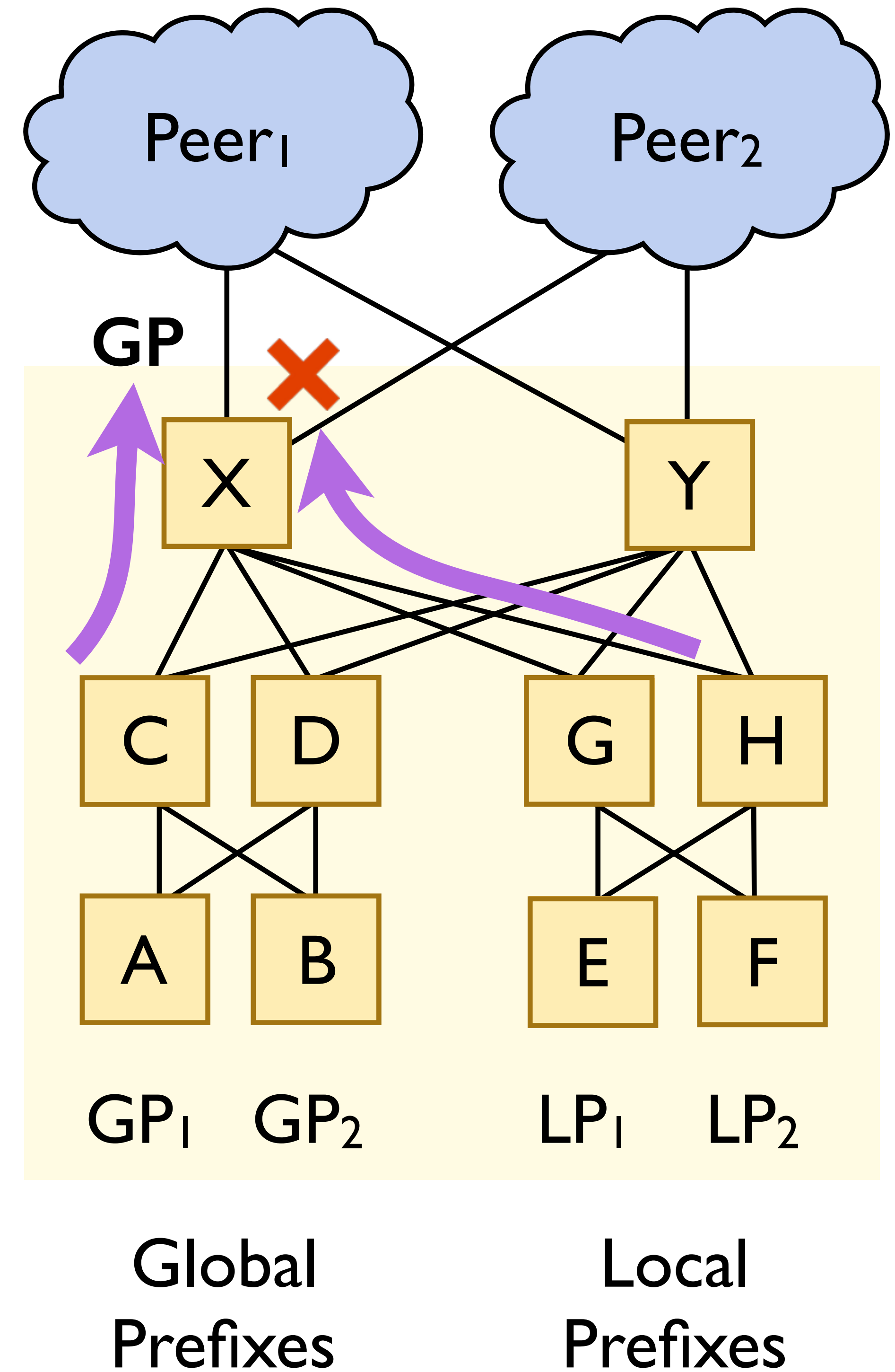
Example: A DC network with traditional configs

Goals

- Local prefixes reachable only internally
- Global prefixes reachable externally
- Aggregate global prefixes as GP
- Prefer leaving through $Peer_1$ over $Peer_2$
- Prevent transit traffic between peers

Configuration Attempt

- Don't export from G, H to external
- Aggregate externally as GP



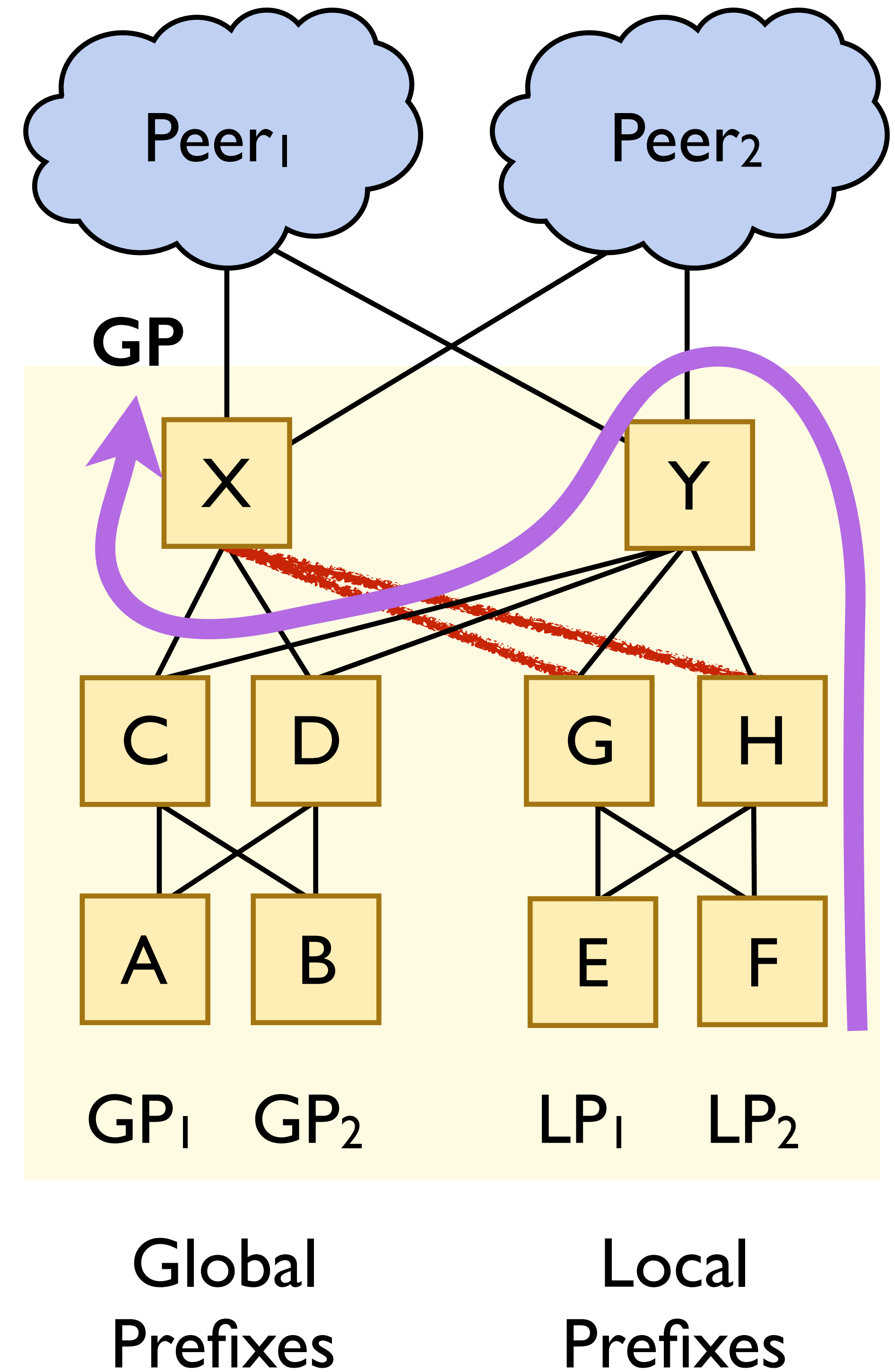
Example: A DC network with traditional configs

Goals

- Local prefixes reachable only internally
- Global prefixes reachable externally
- Aggregate global prefixes as GP
- Prefer leaving through $Peer_1$ over $Peer_2$
- Prevent transit traffic between peers

Configuration Attempt

- Don't export from G, H to external
- Aggregate externally as GP



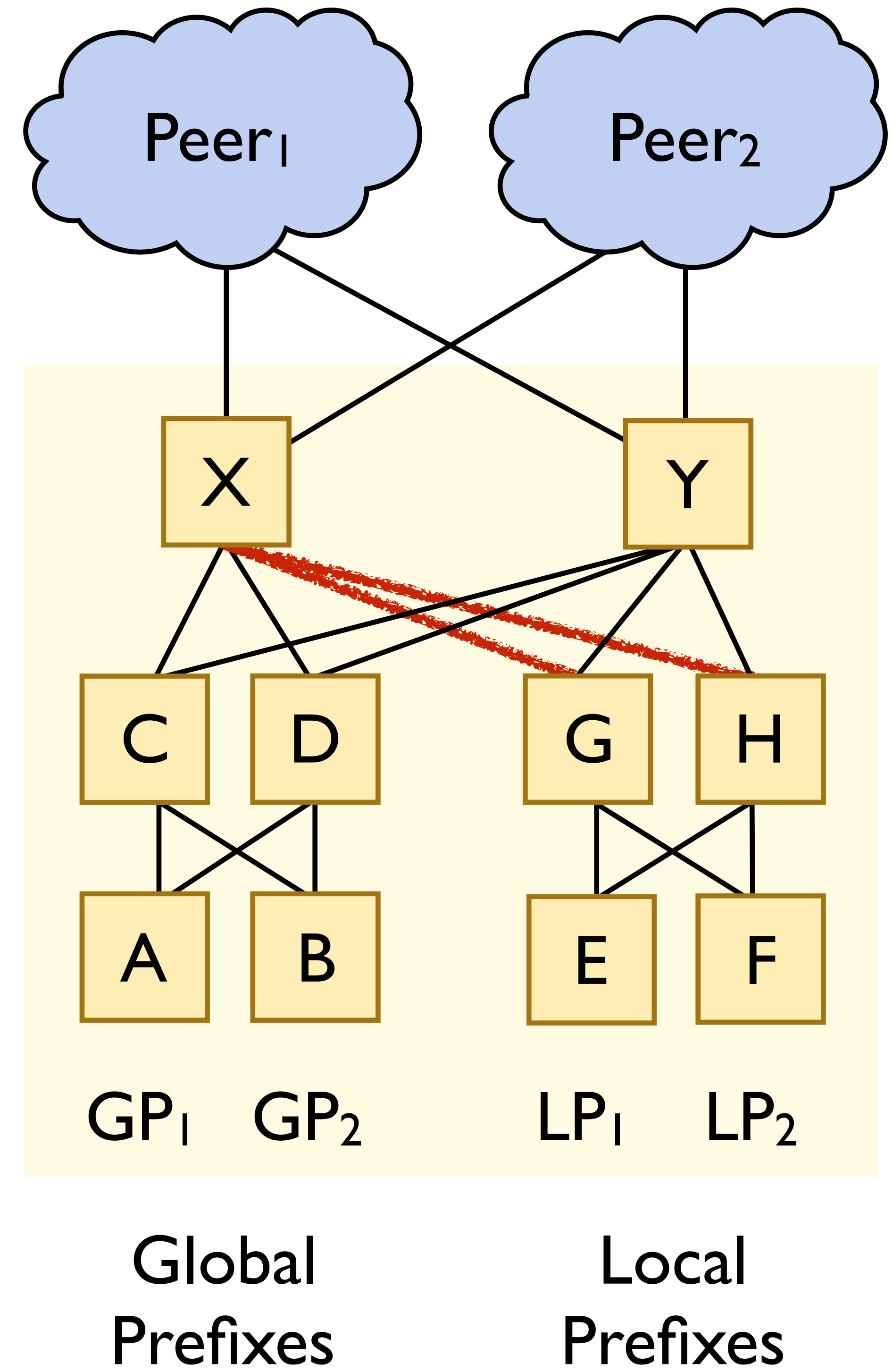
Example: A DC network with traditional configs

Goals

- Local prefixes reachable only internally
- Global prefixes reachable externally
- Aggregate global prefixes as GP
- Prefer leaving through $Peer_1$ over $Peer_2$
- Prevent transit traffic between peers

Configuration Attempt

- Don't export from G, H to external
- Aggregate externally as GP
- **X,Y block routes through each other**



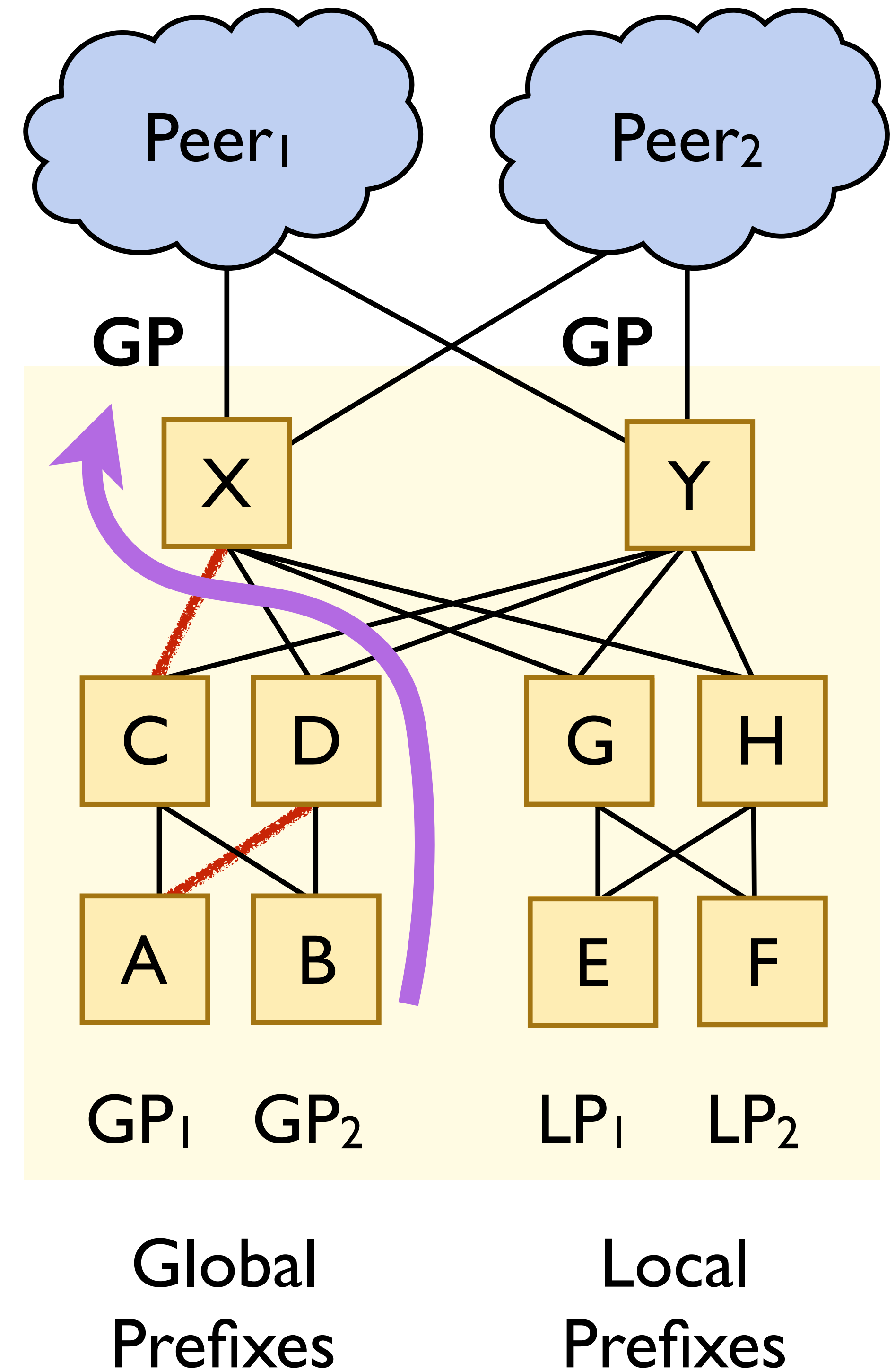
Example: A DC network with traditional configs

Goals

- Local prefixes reachable only internally
- Global prefixes reachable externally
- Aggregate global prefixes as GP
- Prefer leaving through $Peer_1$ over $Peer_2$
- Prevent transit traffic between peers

Configuration Attempt

- Don't export from G, H to external
- Aggregate externally as GP
- X,Y block routes through each other



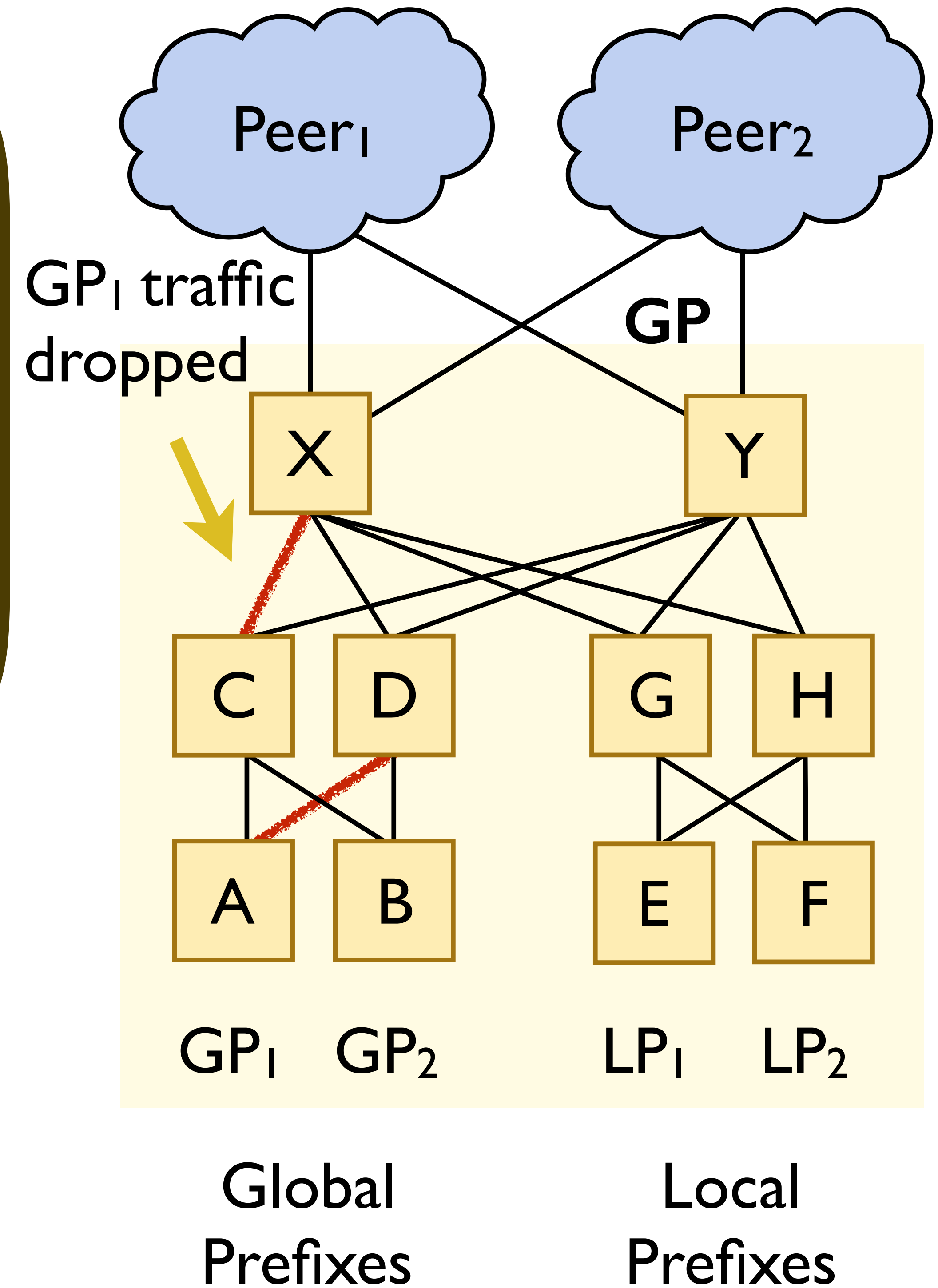
Example: A DC network with traditional configs

Goals

- Local prefixes reachable only internally
- Global prefixes reachable externally
- Aggregate global prefixes as GP
- Prefer leaving through $Peer_1$ over $Peer_2$
- Prevent transit traffic between peers

Configuration Attempt

- Don't export from G, H to external
- Aggregate externally as GP
- X,Y block routes through each other



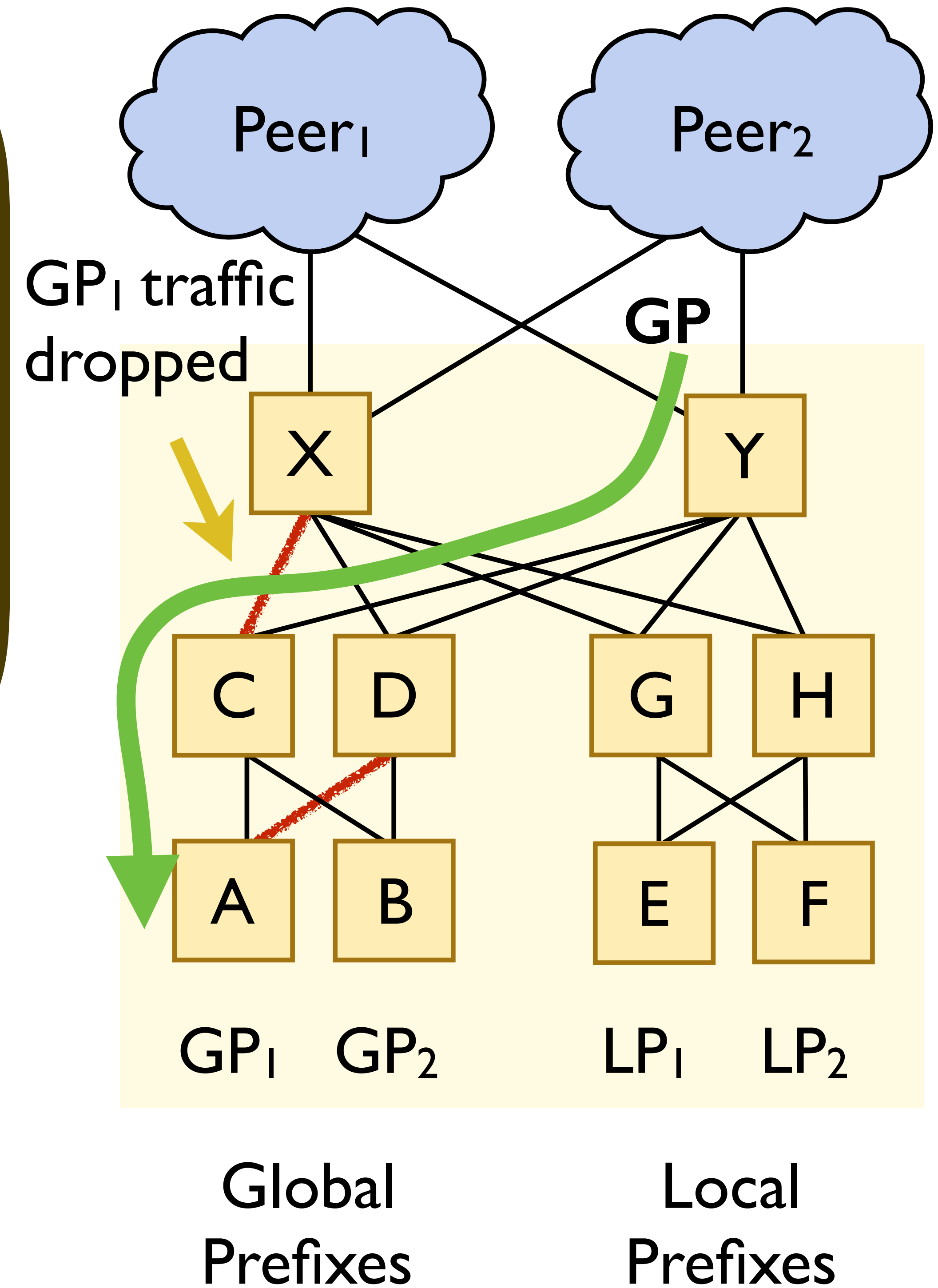
Example: A DC network with traditional configs

Goals

- Local prefixes reachable only internally
- Global prefixes reachable externally
- Aggregate global prefixes as GP
- Prefer leaving through $Peer_1$ over $Peer_2$
- Prevent transit traffic between peers

Configuration Attempt

- Don't export from G, H to external
- Aggregate externally as GP
- X,Y block routes through each other

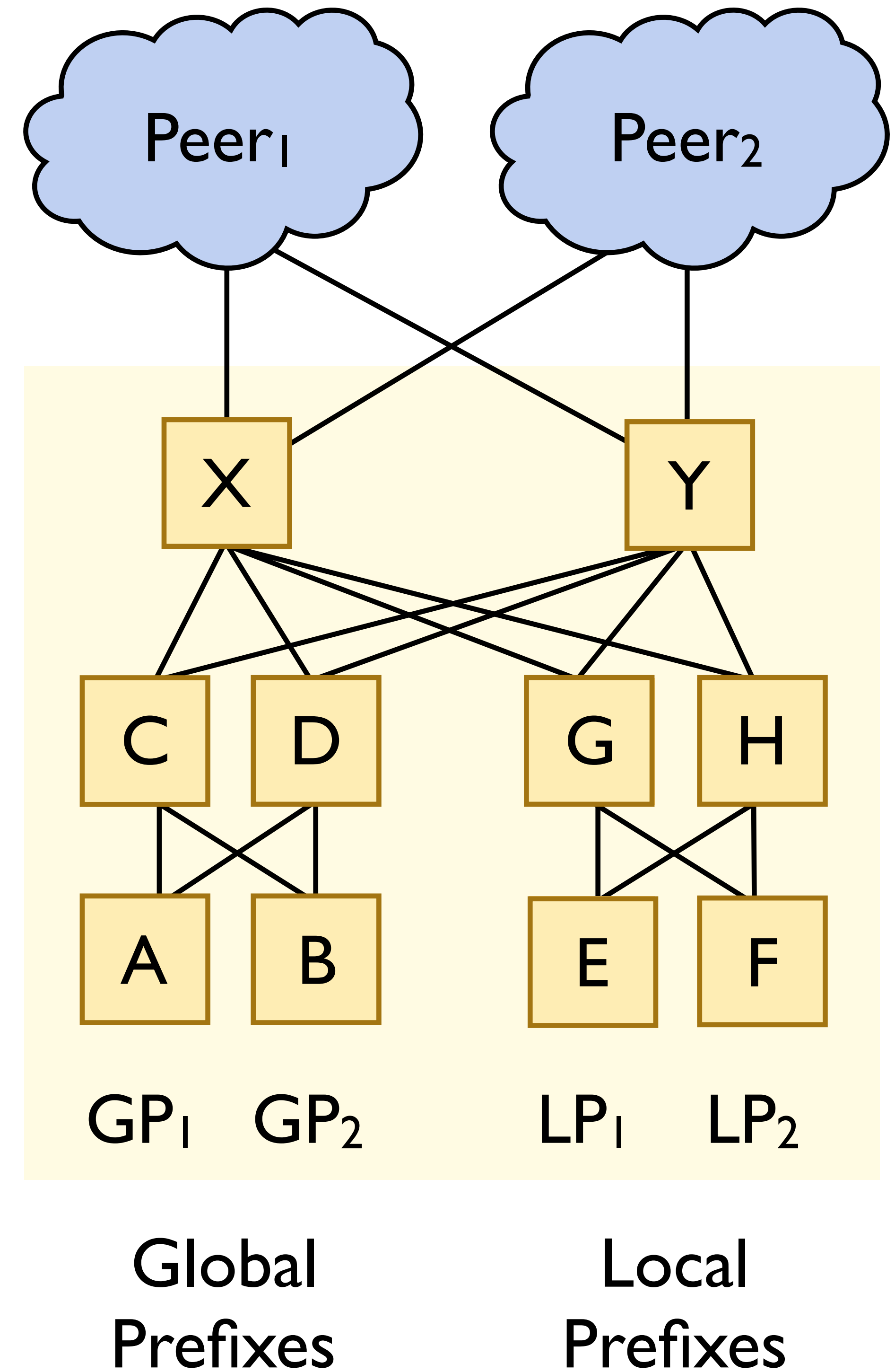


Aggregation-Induced Black Hole!

Example: A DC network with traditional configs

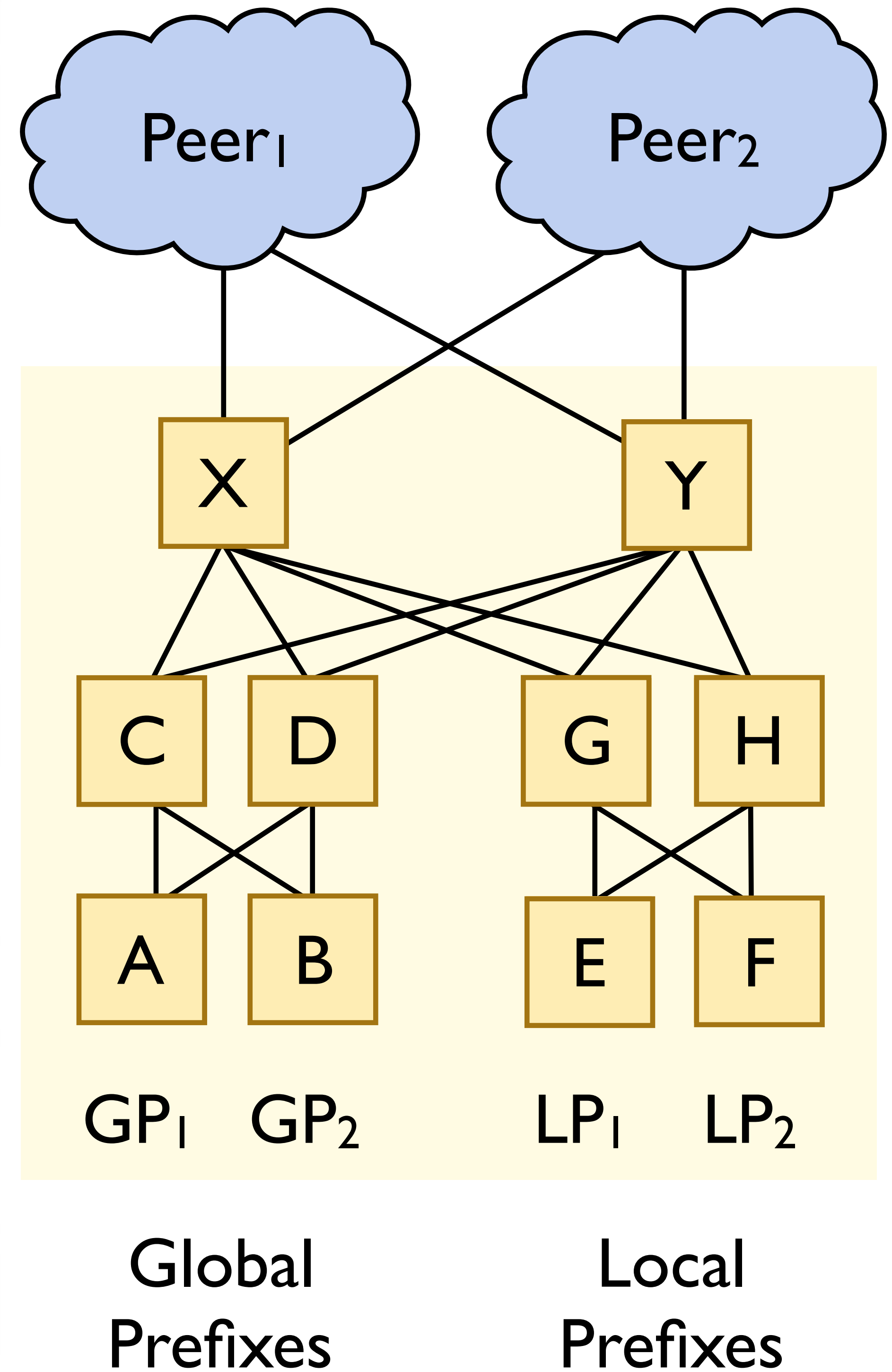
Goals

- Local prefixes reachable only internally
- Global prefixes reachable externally
- Aggregate global prefixes as GP
- Prefer leaving through $Peer_1$ over $Peer_2$
- Prevent transit traffic between peers



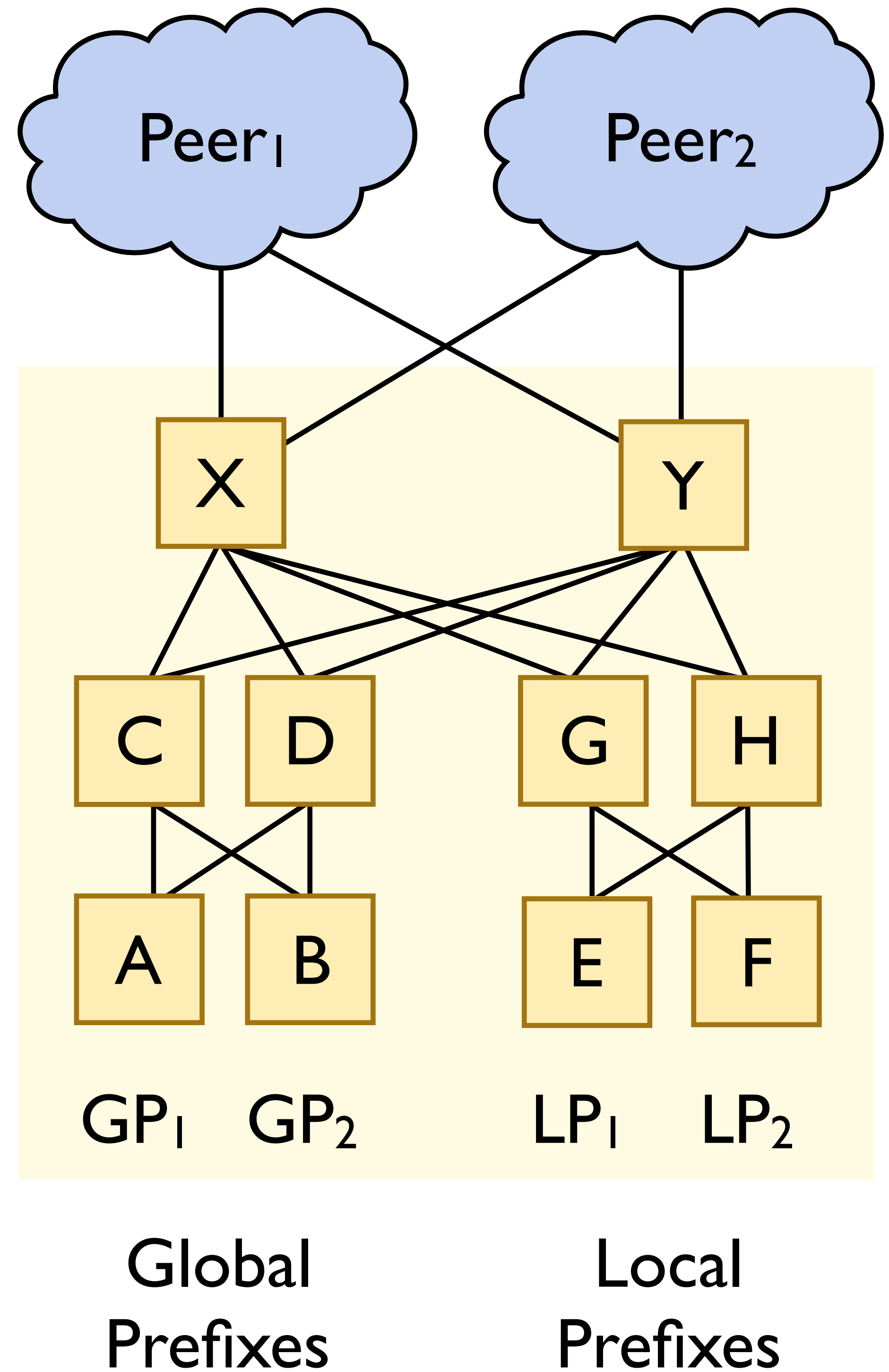
Example: A DC network with Propane

```
define Destination =
 {GP1   => end(A)
  GP2   => end(B)
  LP1   => end(E)
  LP2   => end(F)
  true => exit(Peer1 >> Peer2)}
```



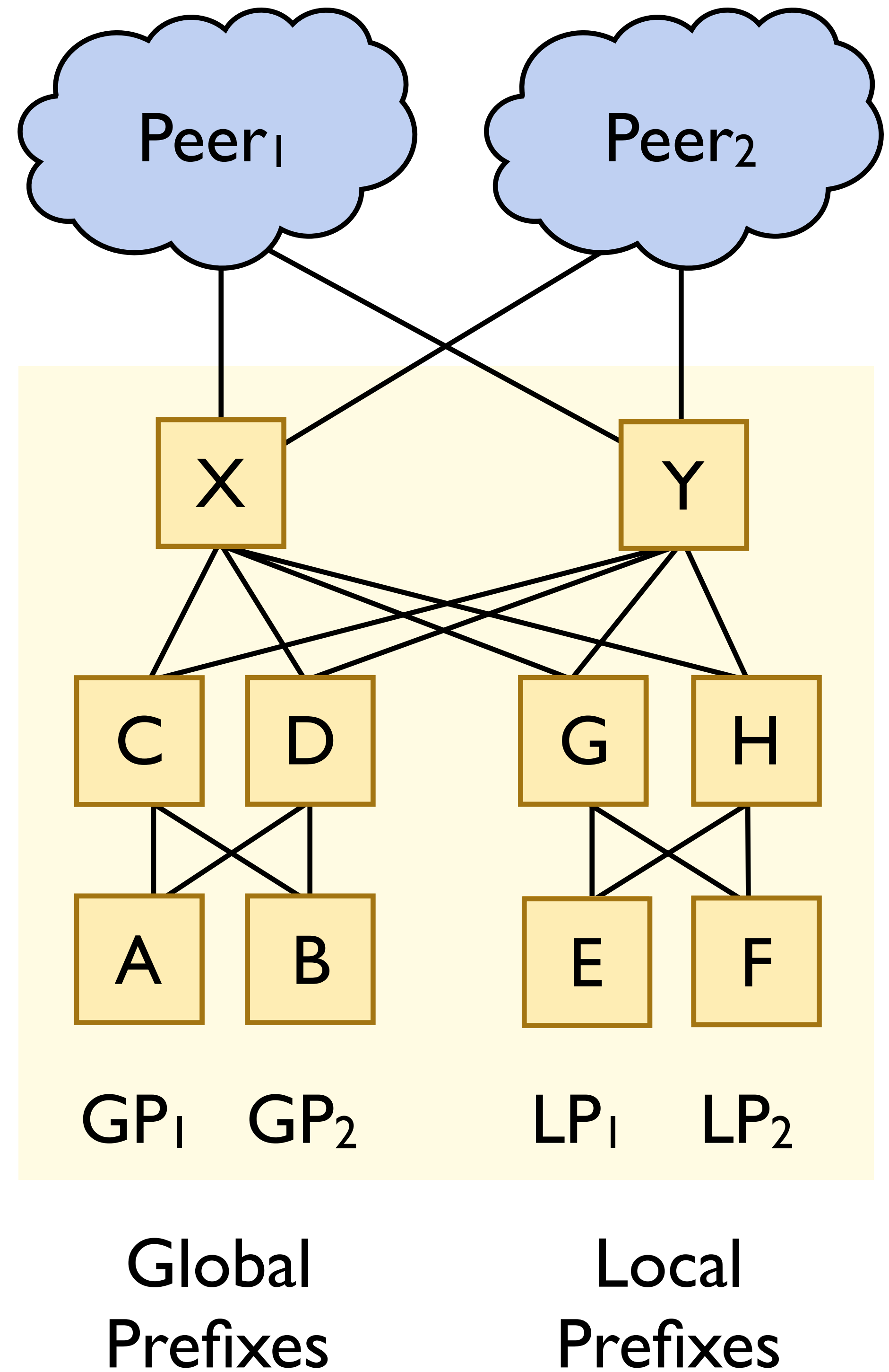
Example: A DC network with Propane

```
define Destination =
 {GP1  => end(A)
  GP2  => end(B)
  LP1  => end(E)
  LP2  => end(F)
  true => exit(Peer1 >> Peer2)}

define Locality =
 {LP1 | LP2 => internal}
```



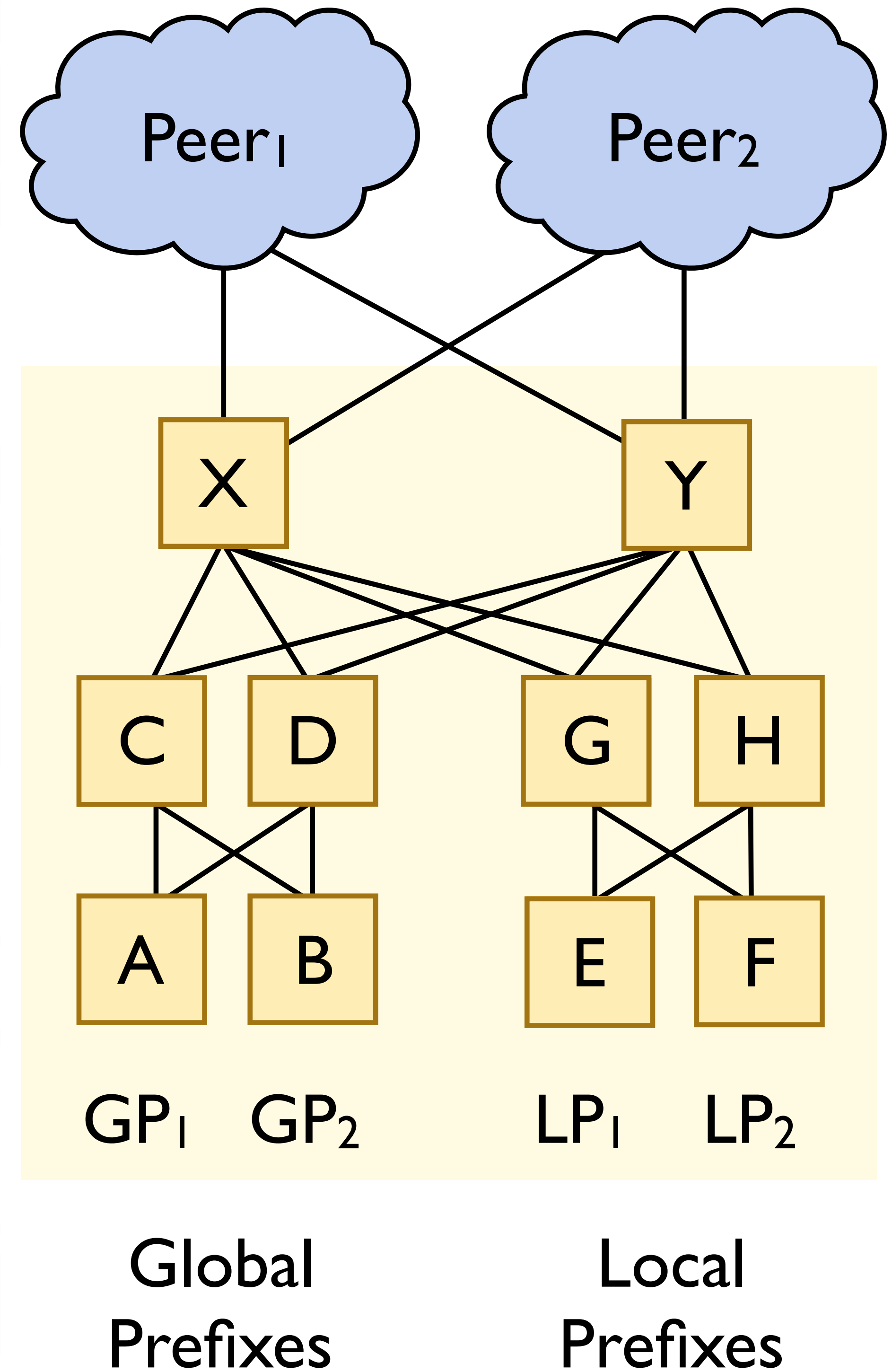
Example: A DC network with Propane

```
define Destination =
 {GP1  => end(A)
  GP2  => end(B)
  LP1  => end(E)
  LP2  => end(F)
  true => exit(Peer1 >> Peer2)}

define Locality =
 {LP1 | LP2 => internal}

define transit(X,Y) =
 enter(X|Y) and exit(X|Y)
```



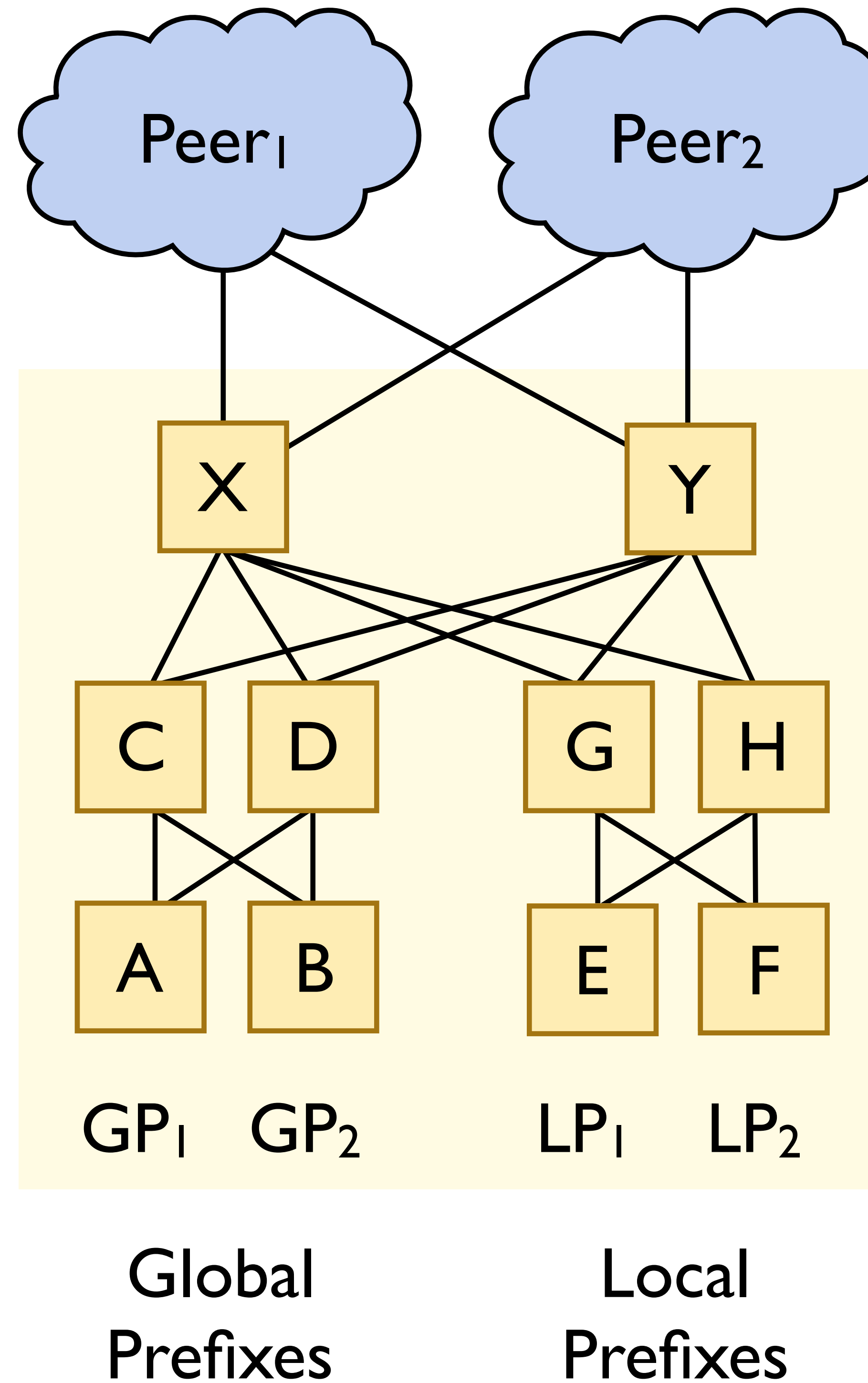
Example: A DC network with Propane

```
define Destination =
 {GP1  => end(A)
  GP2  => end(B)
  LP1  => end(E)
  LP2  => end(F)
  true => exit(Peer1 >> Peer2)}

define Locality =
 {LP1 | LP2 => internal}

define transit(X,Y) =
 enter(X|Y) and exit(X|Y)

define NoTransit =
 {true => !transit(Peer1,Peer2)}
```

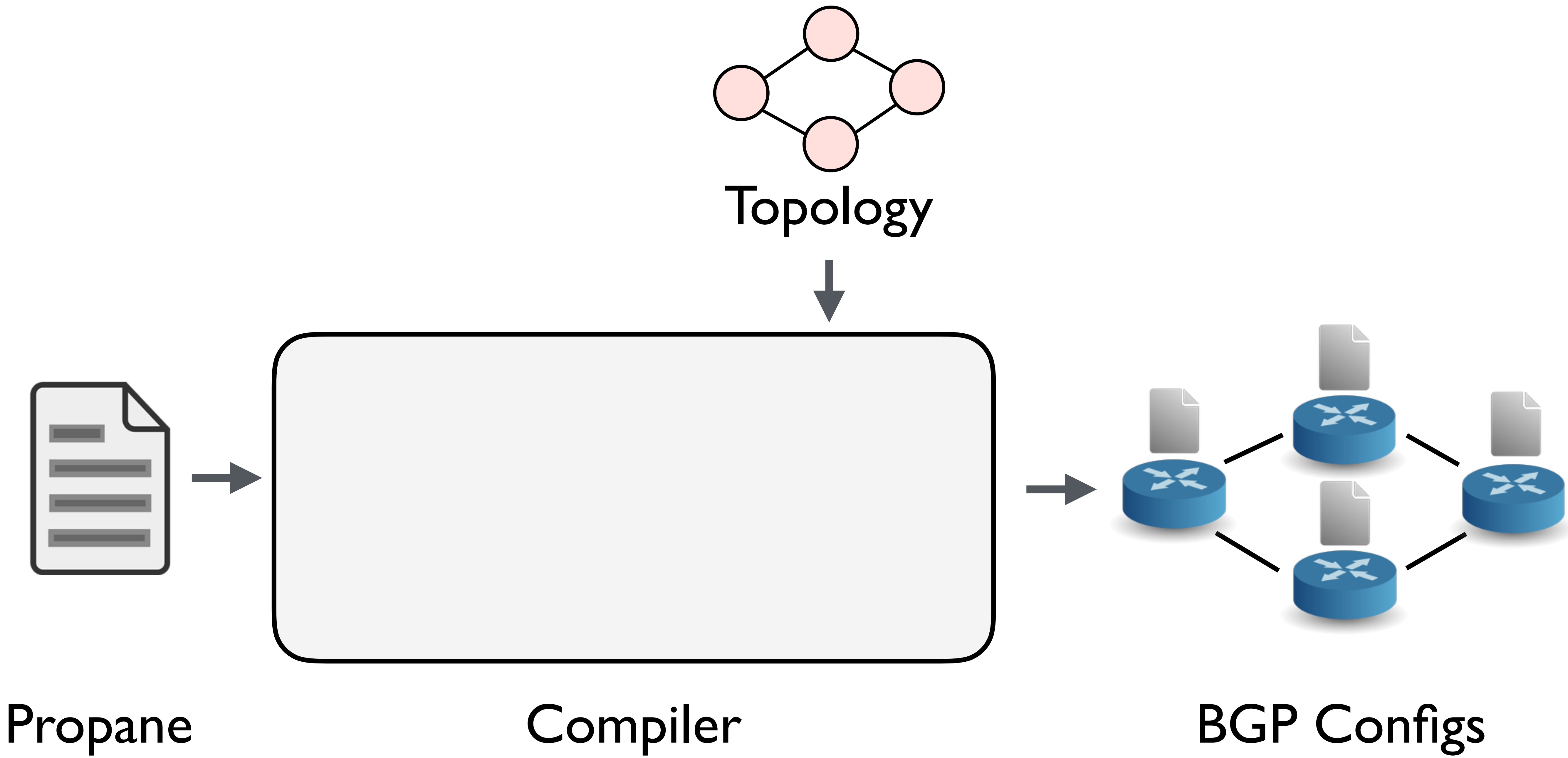


Example: A DC network with Propane

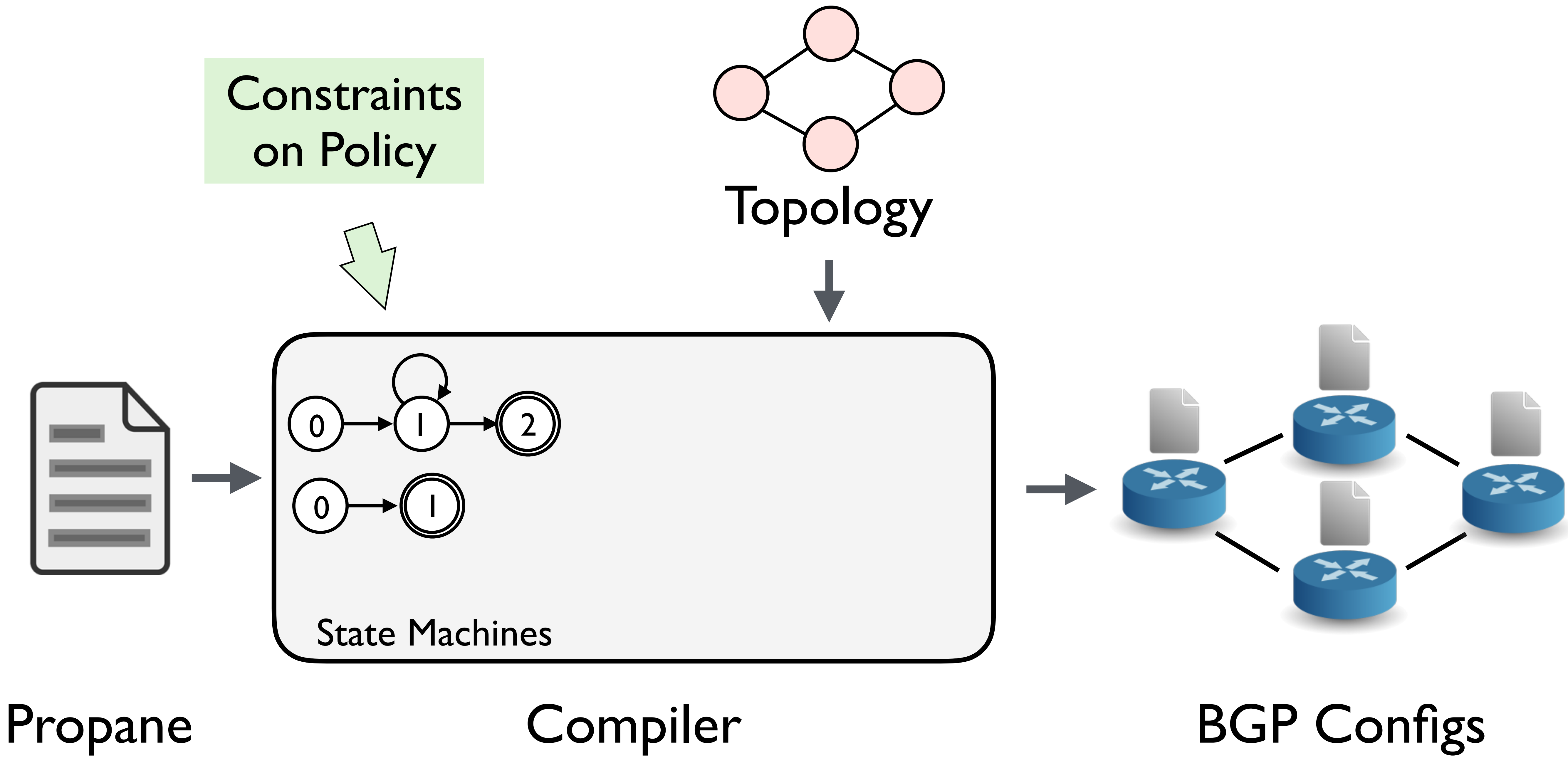
```
define Destination =
 {GP1  => end(A)
  GP2  => end(B)
  LP1  => end(E)
  LP2  => end(F)
  true => exit(Peer1 >> Peer2)}

define Locality =
 {LP1 | LP2 => internal}

define transit(X,Y) =
 enter(X|Y) and exit(X|Y)

define NoTransit =
 {true => !transit(Peer1,Peer2)}

define Main =
   Destination & Locality &
   NoTransit & agg(GP, in -> out)
```



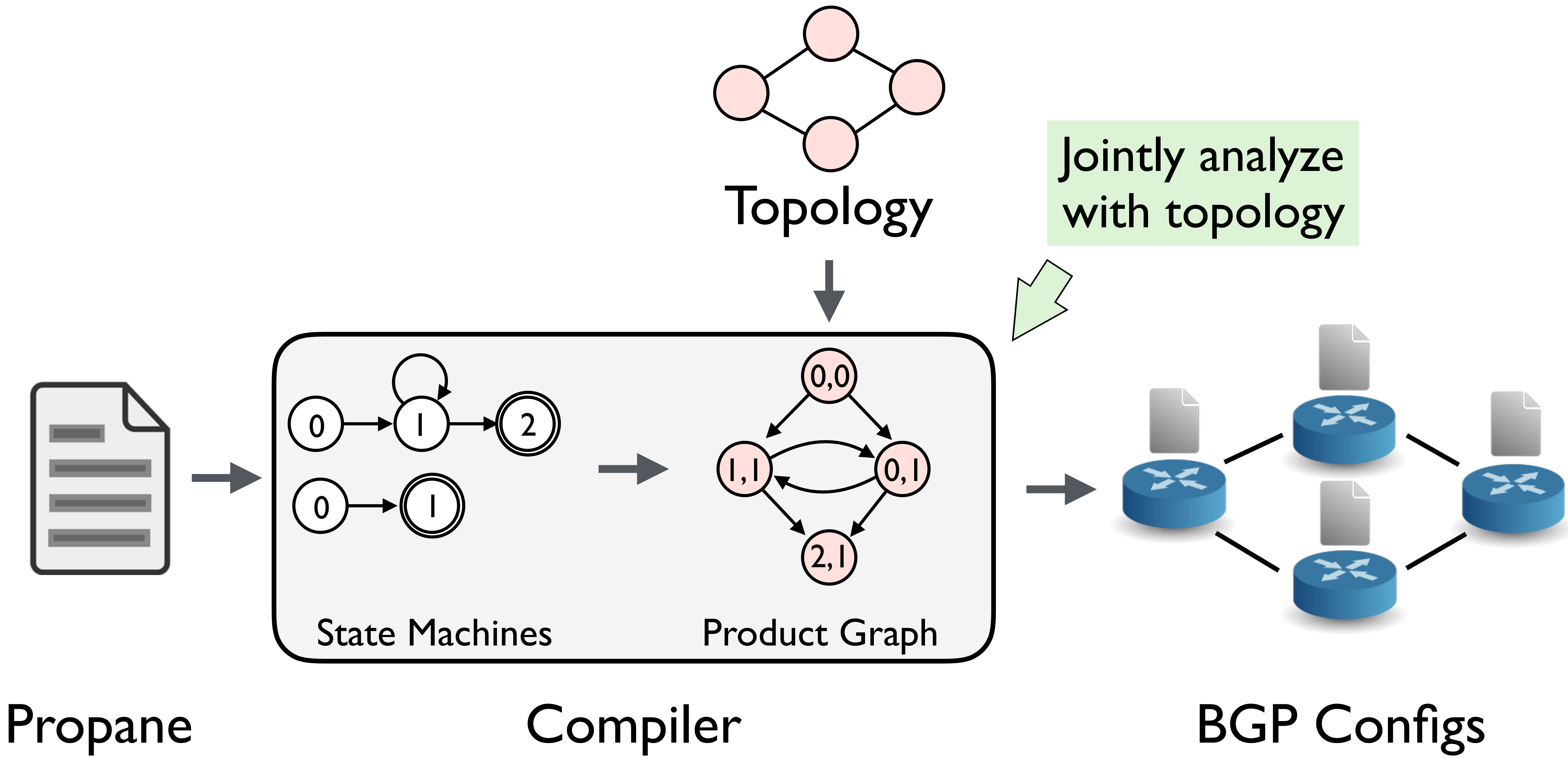
Compilation



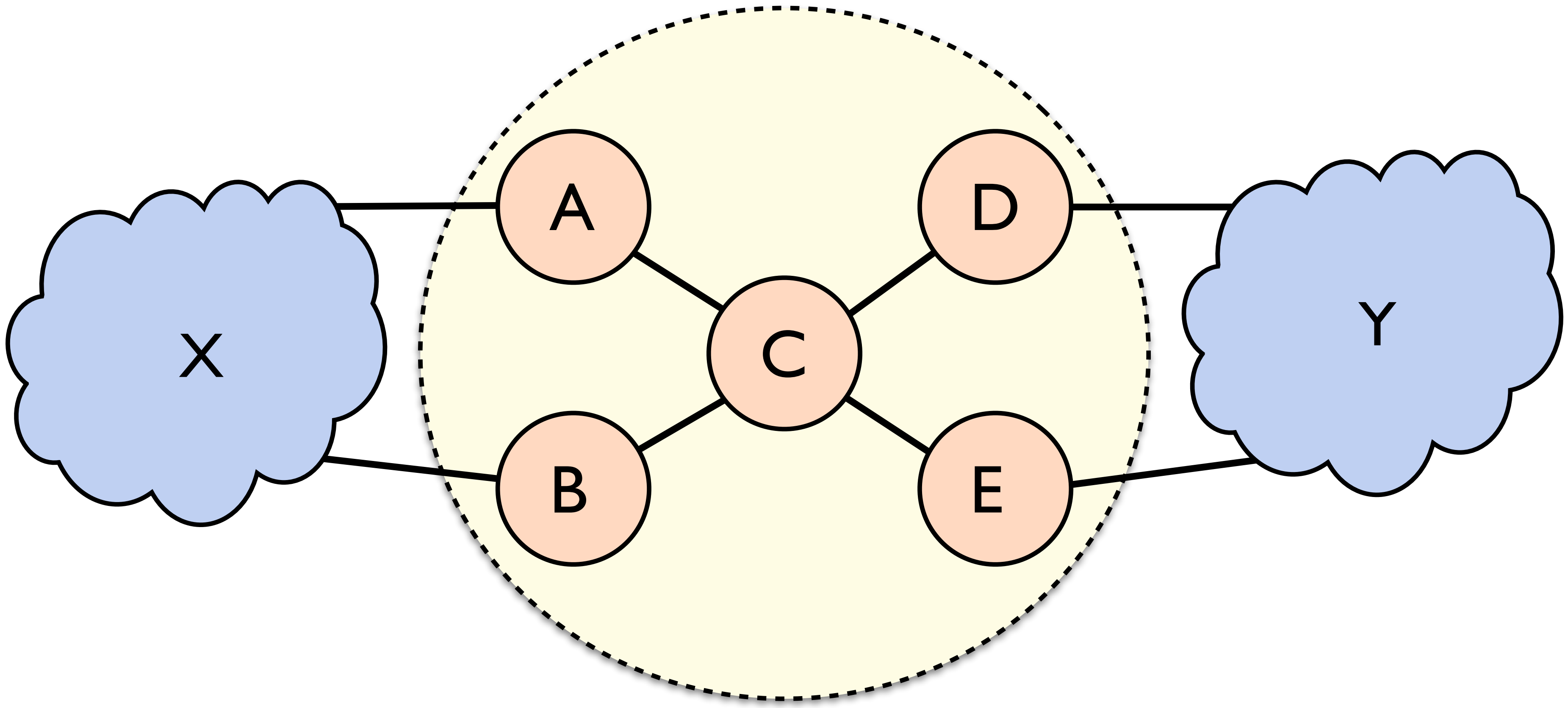
Compilation



Compilation

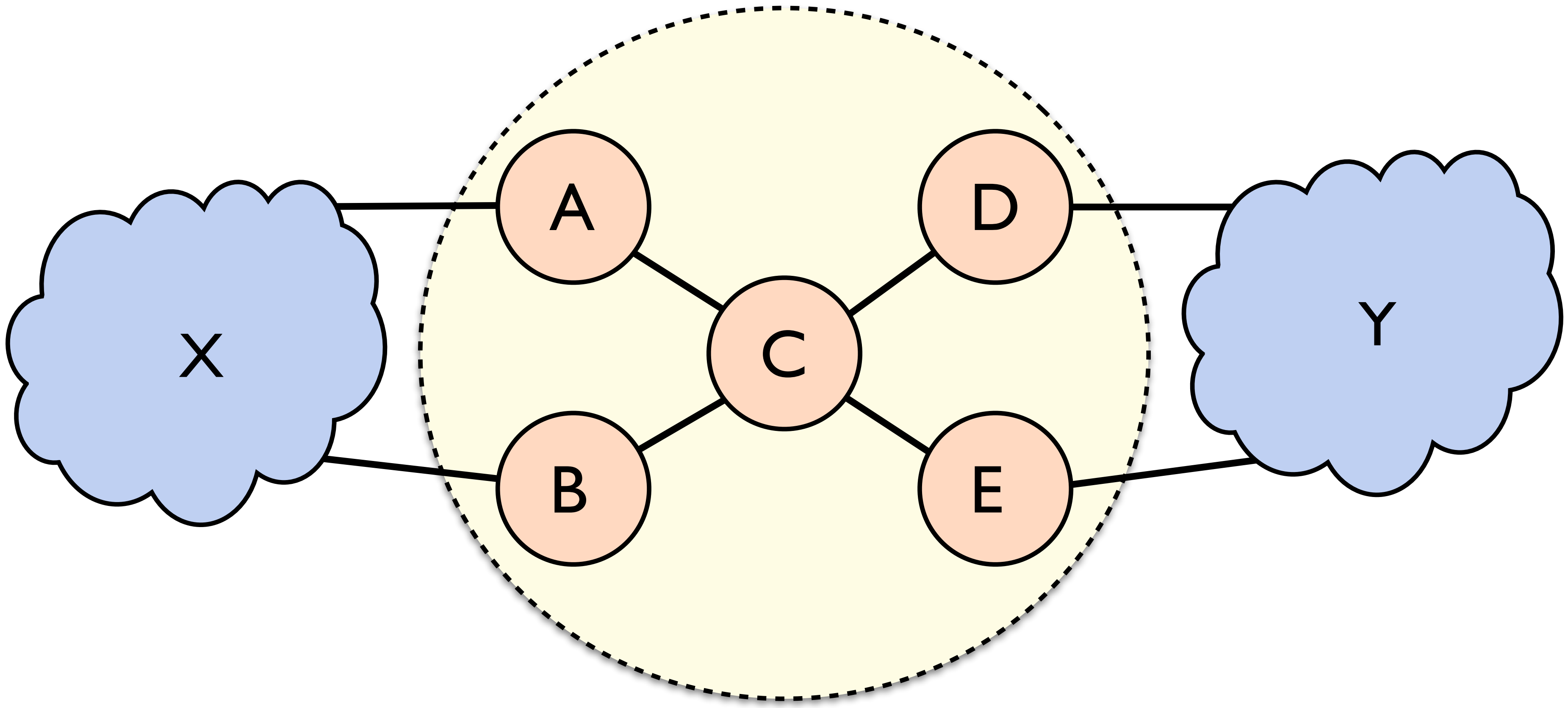


Compilation: A simple Example

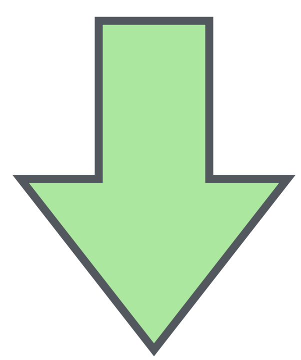


end(Y) & (**path**(A,C,D) >> **any**)

Compilation: A simple Example



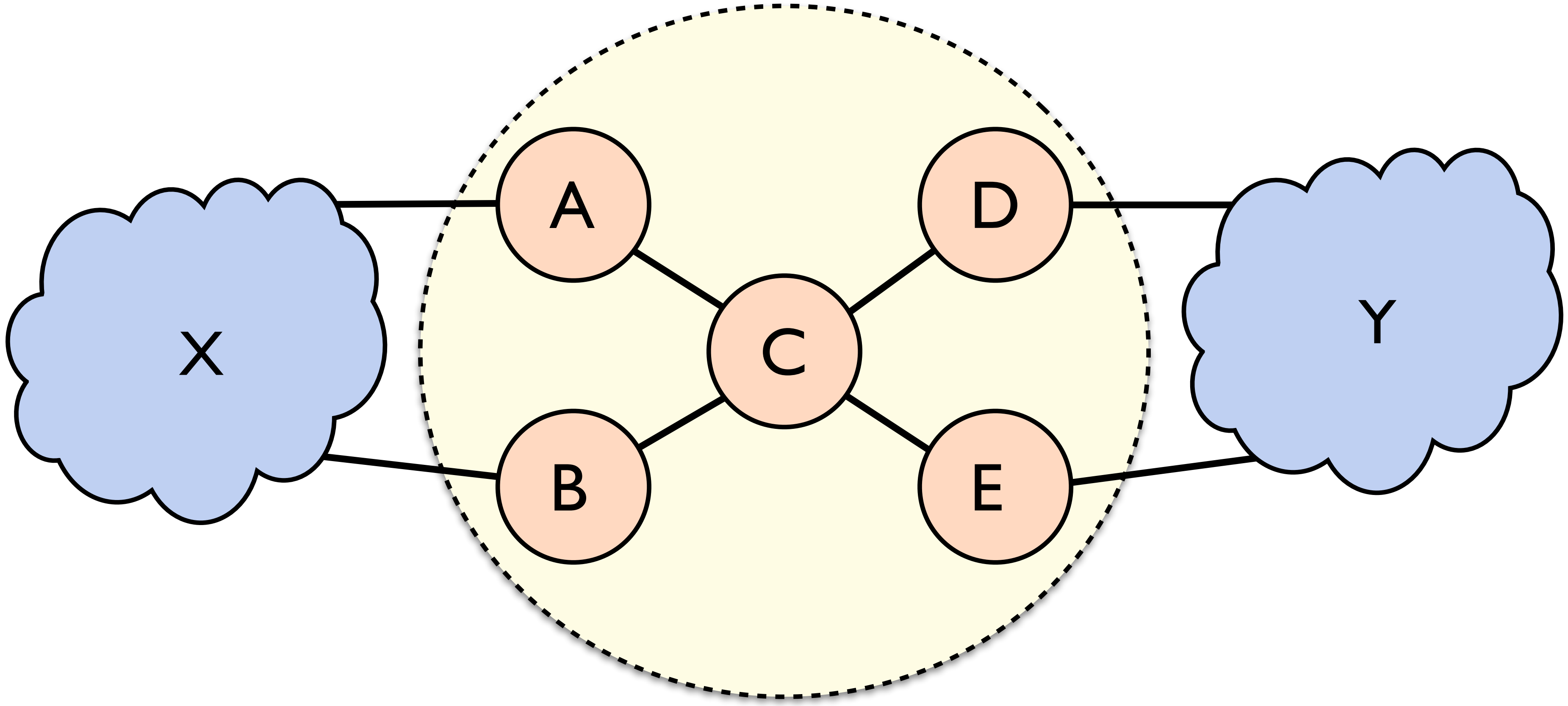
end(Y) & (**path**(A,C,D) >> **any**)



Convert to Regex

XACDY >> (Σ*)Y

Reversed Automata from Policies



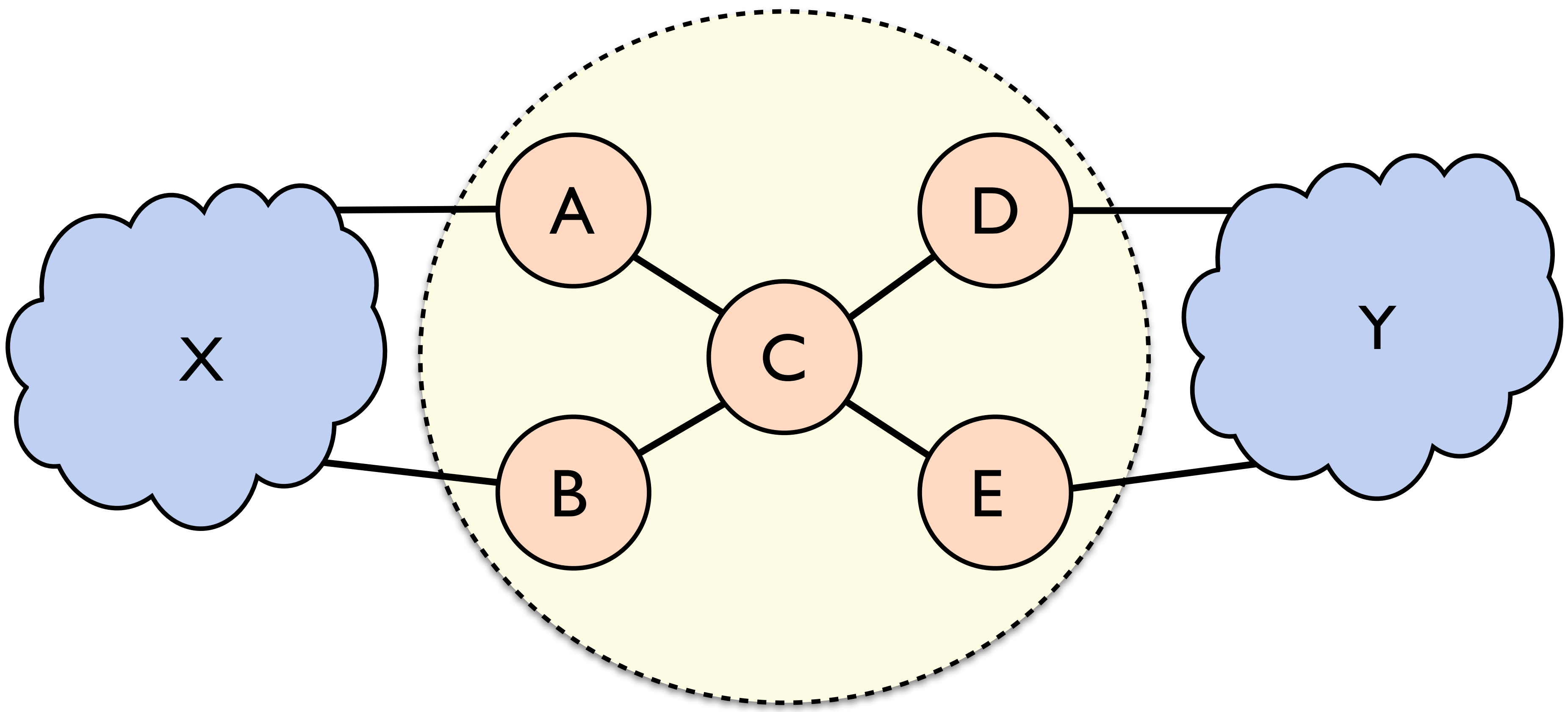
Policy:

```
1.  XACDY

2. (Σ*)Y
```

More preferred paths

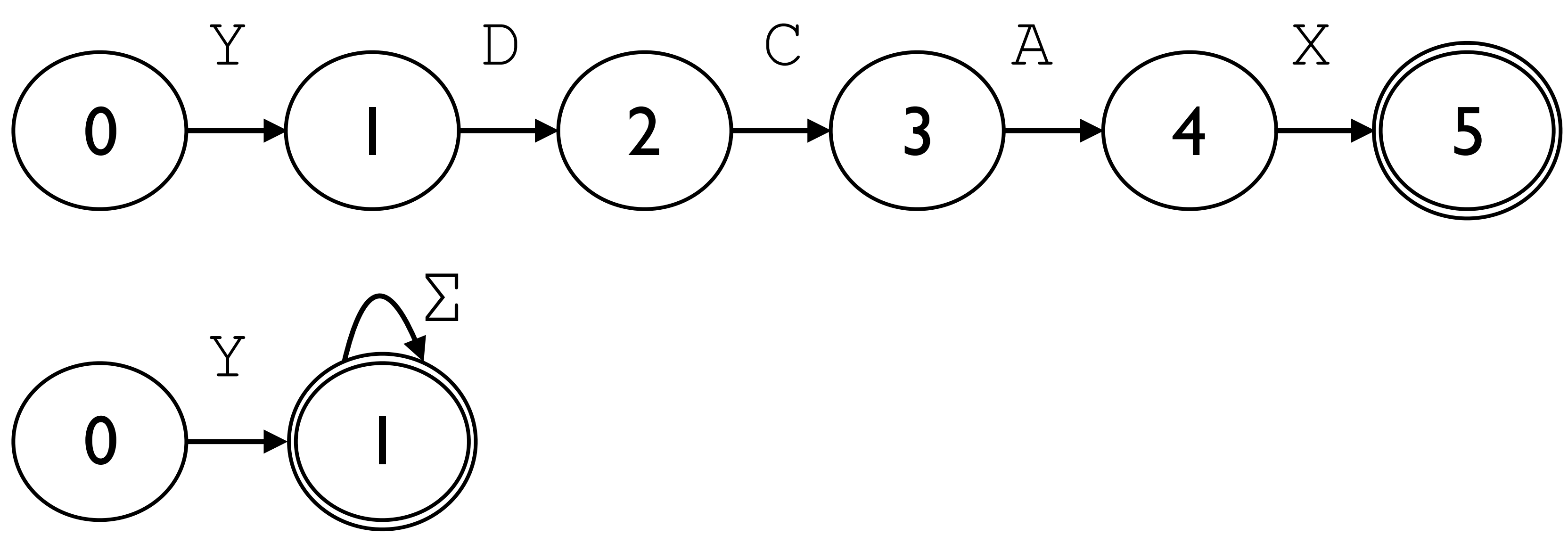
Less preferred paths

Reversed Automata from Policies

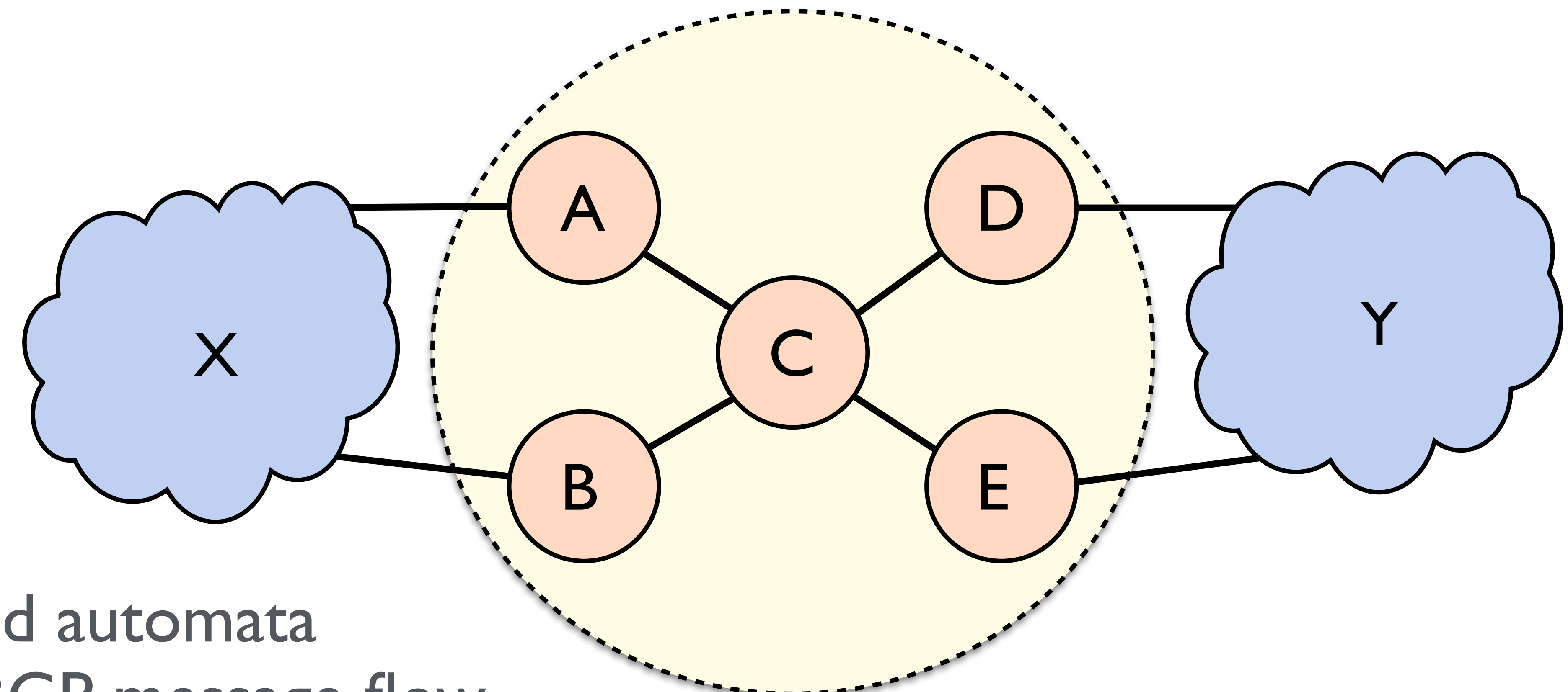


Policy:

1. XACDY
2. (Σ*)Y



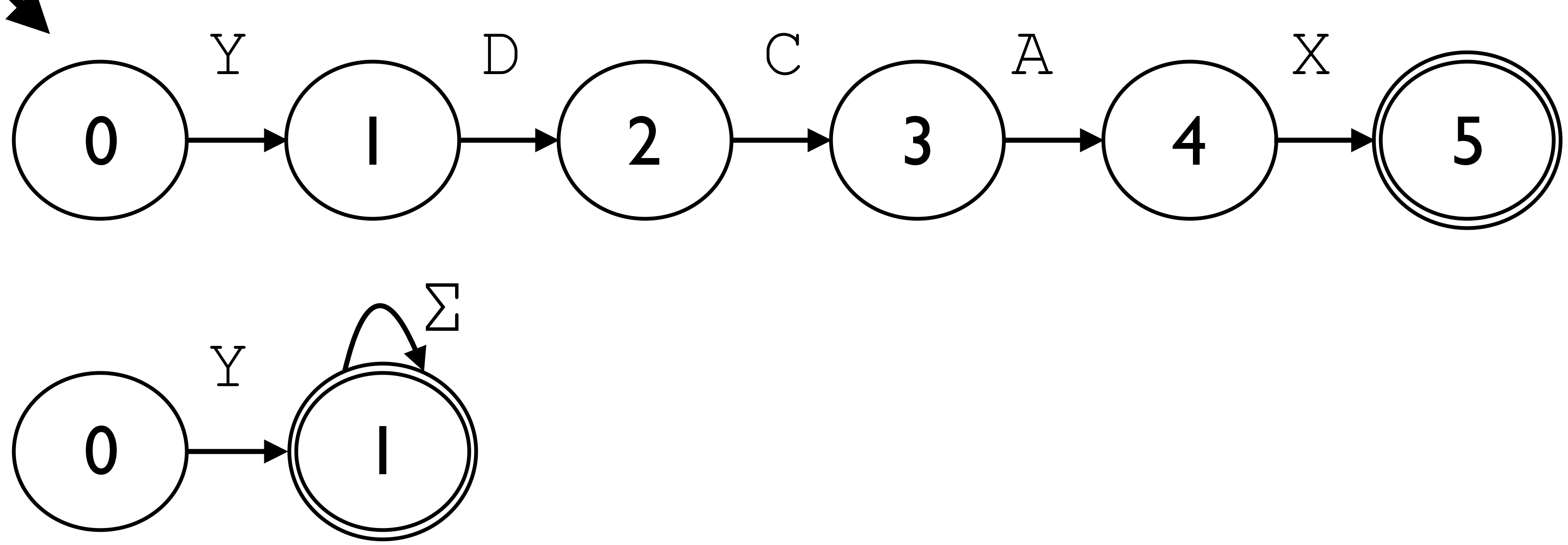
Reversed Automata from Policies



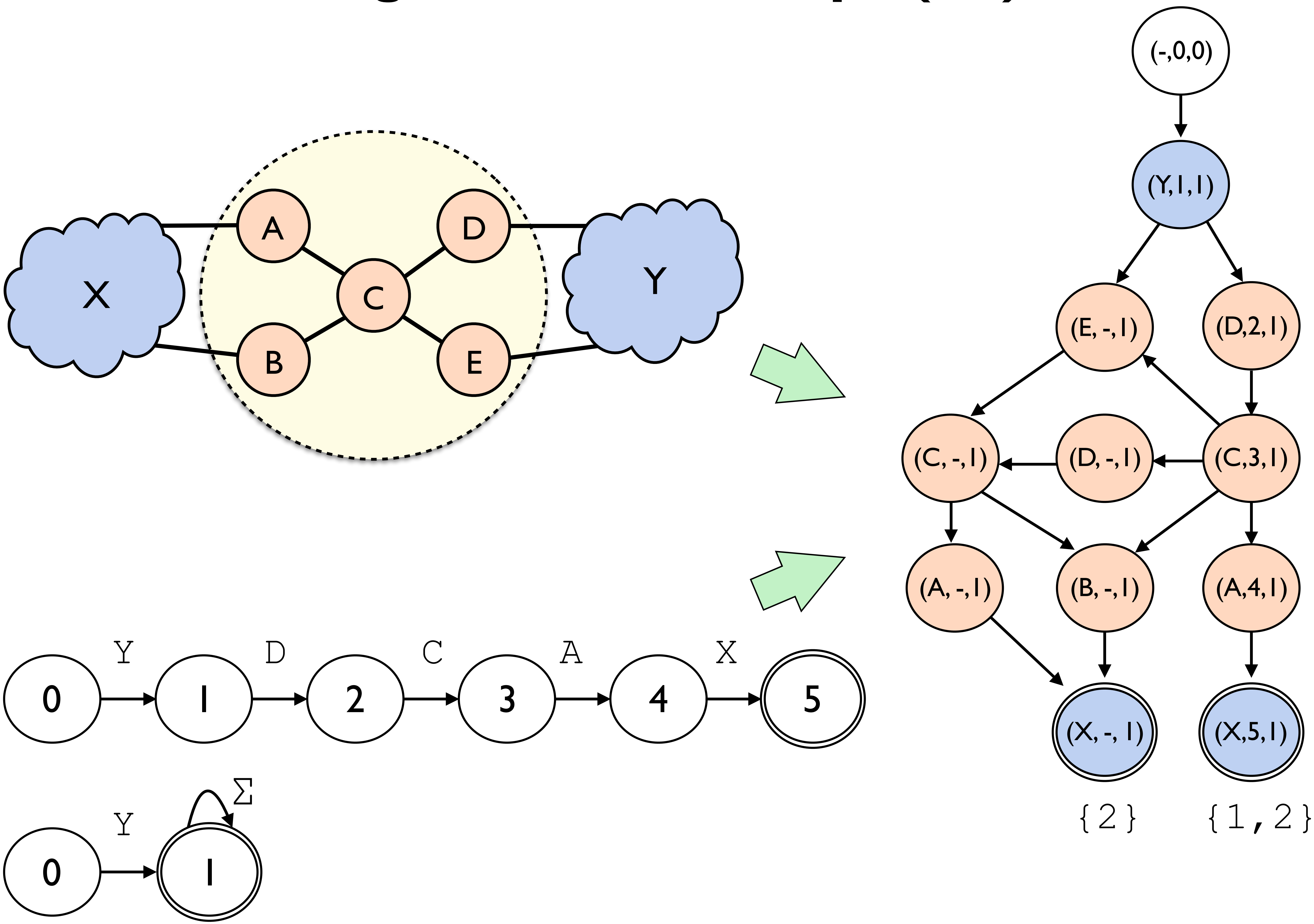
Reversed automata
tracks BGP message flow

Policy:

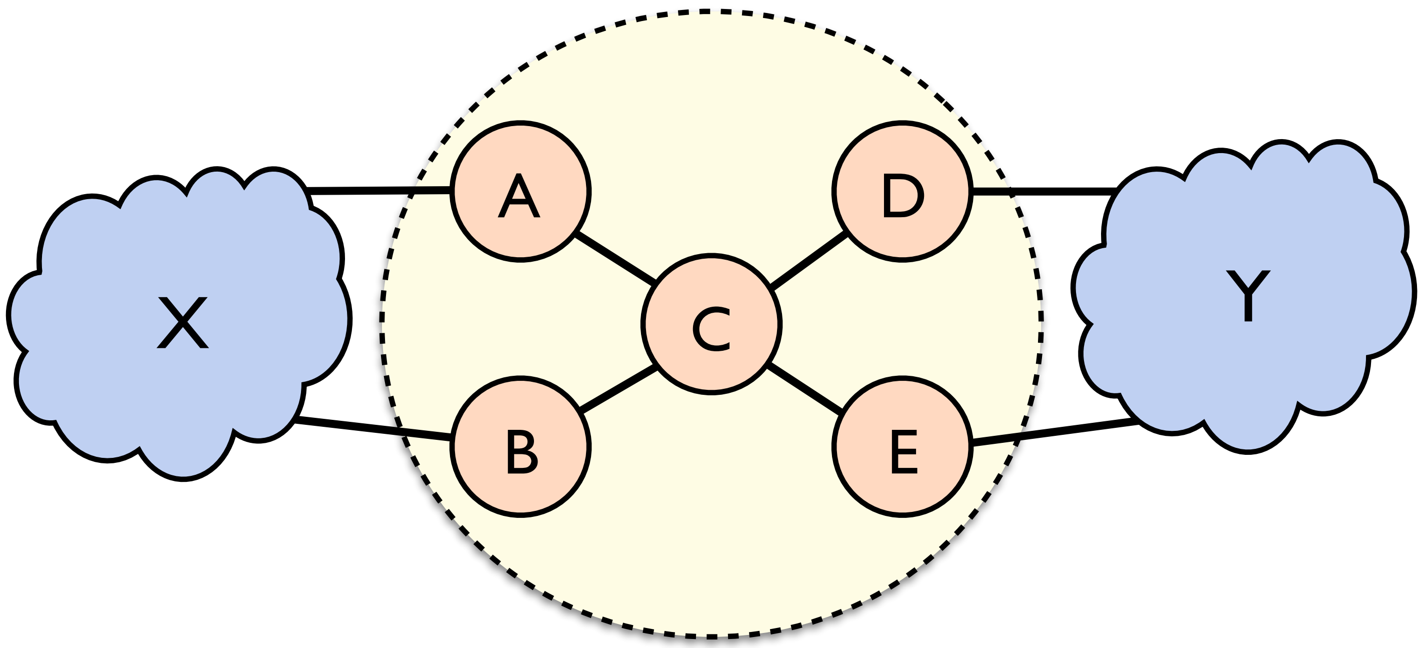
```
1.  XACDY

2. (Σ*)Y
```

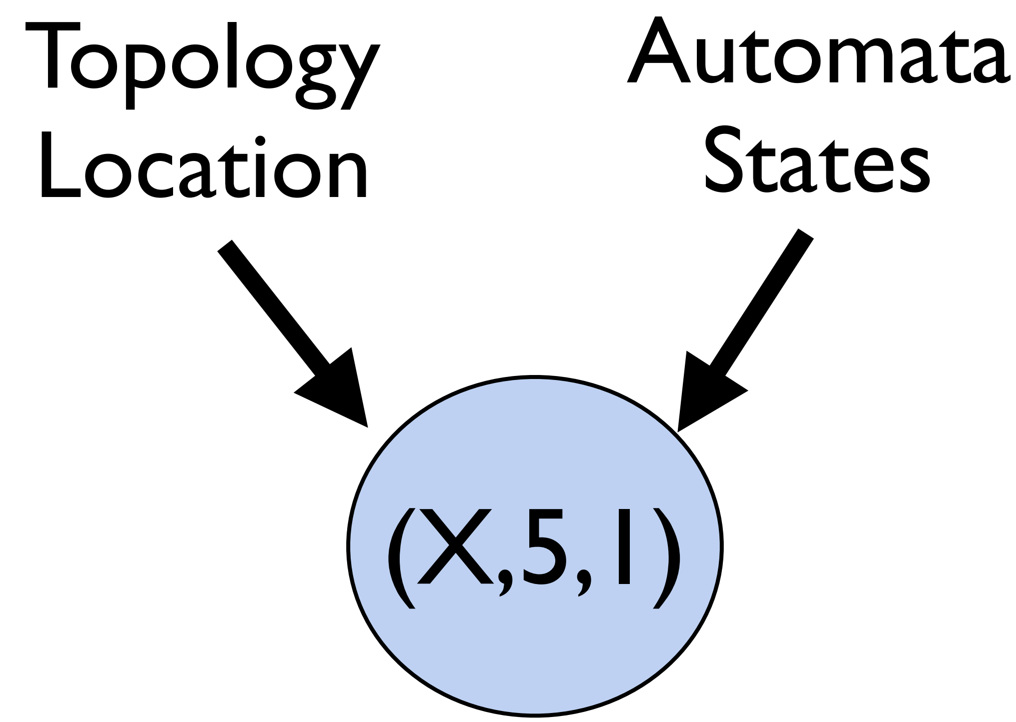


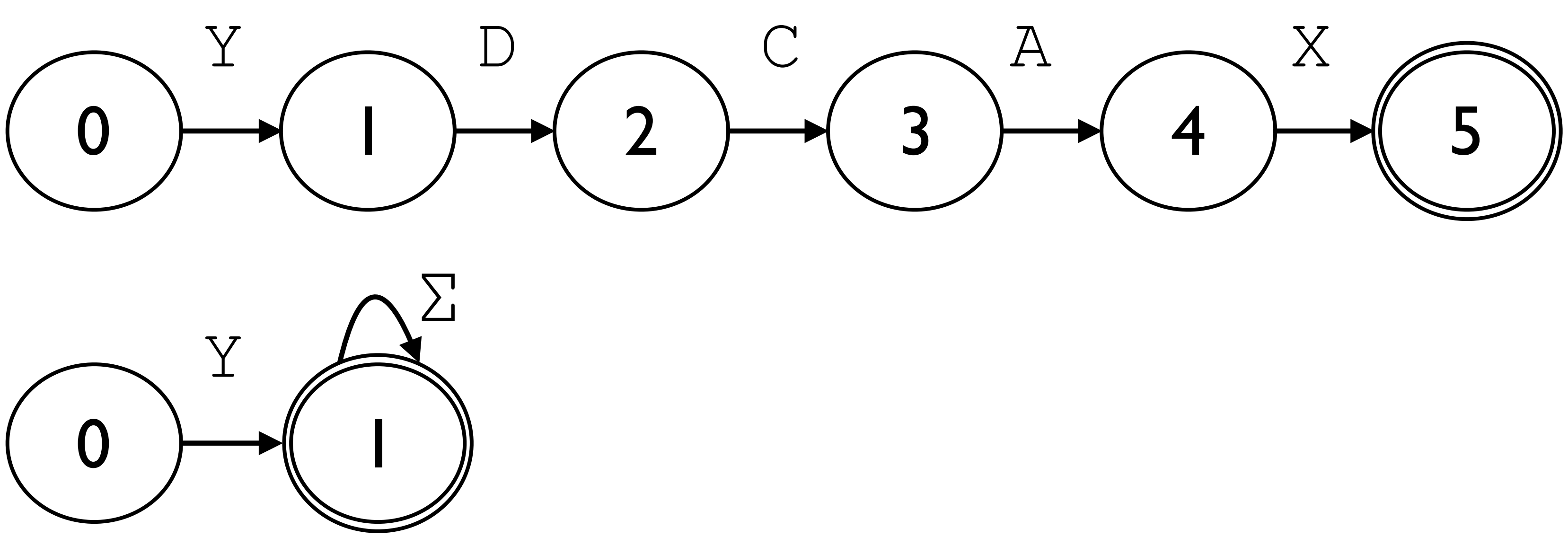
Constructing the Product Graph (PG)



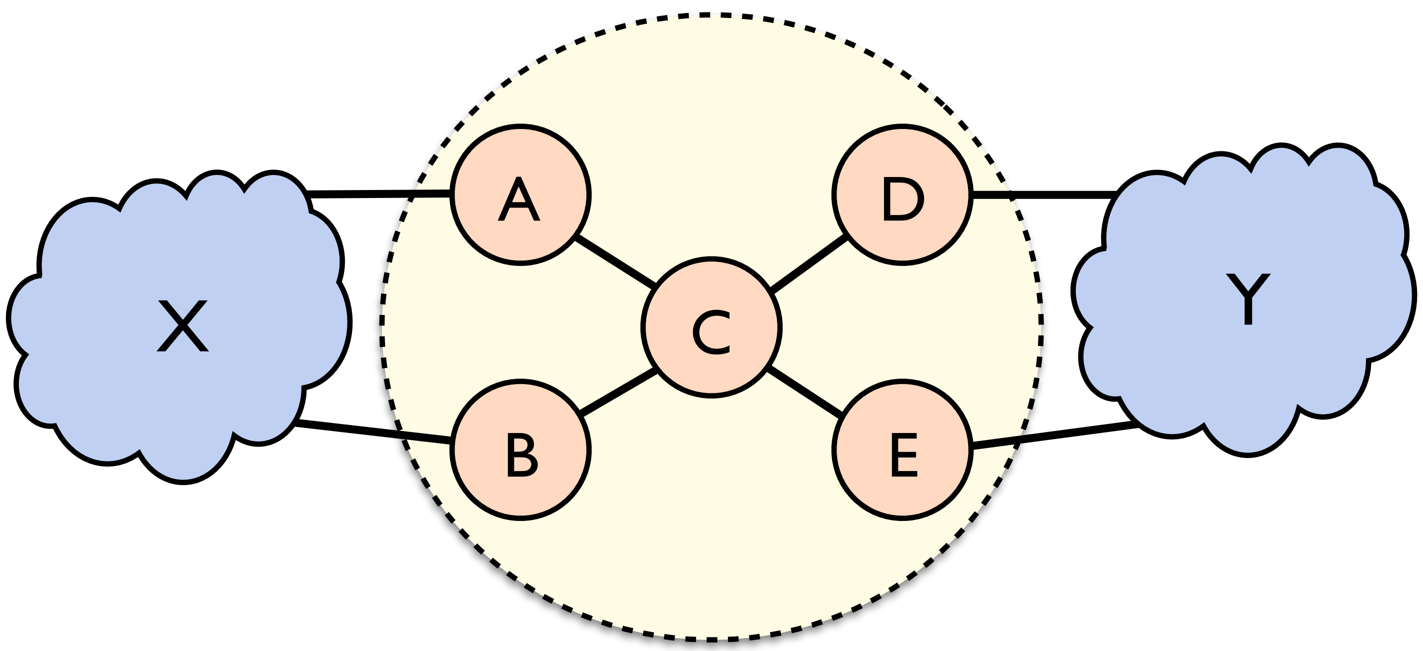
Constructing the Product Graph (PG)

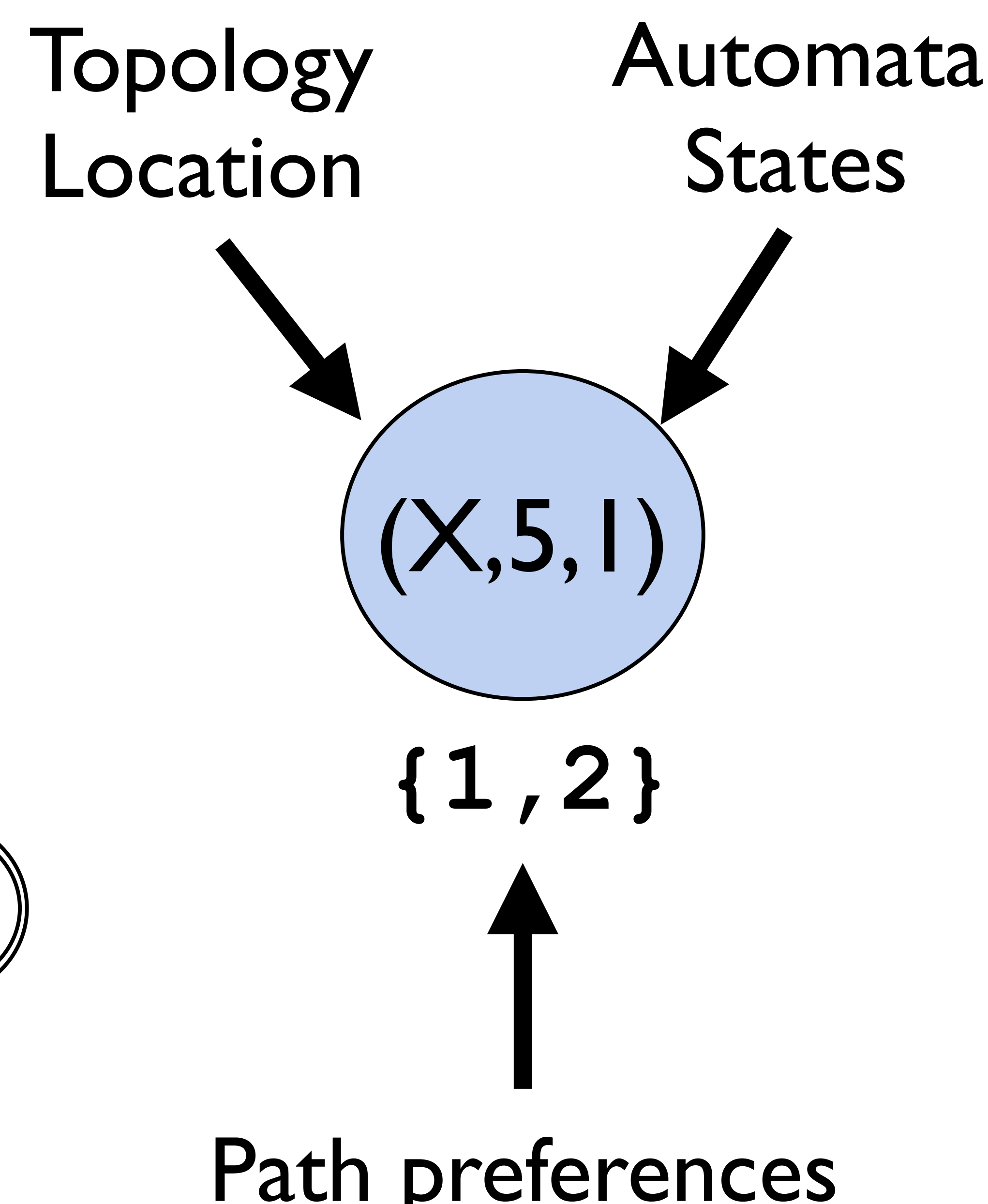


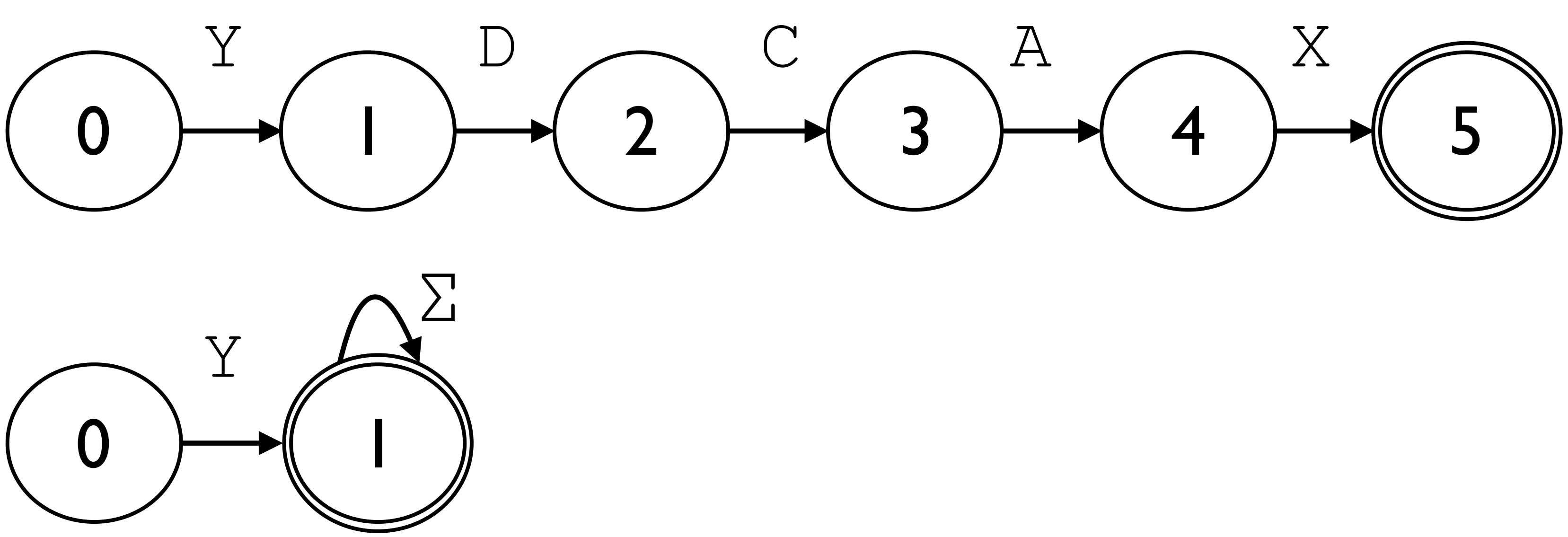




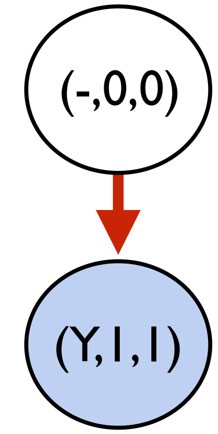
Constructing the Product Graph (PG)

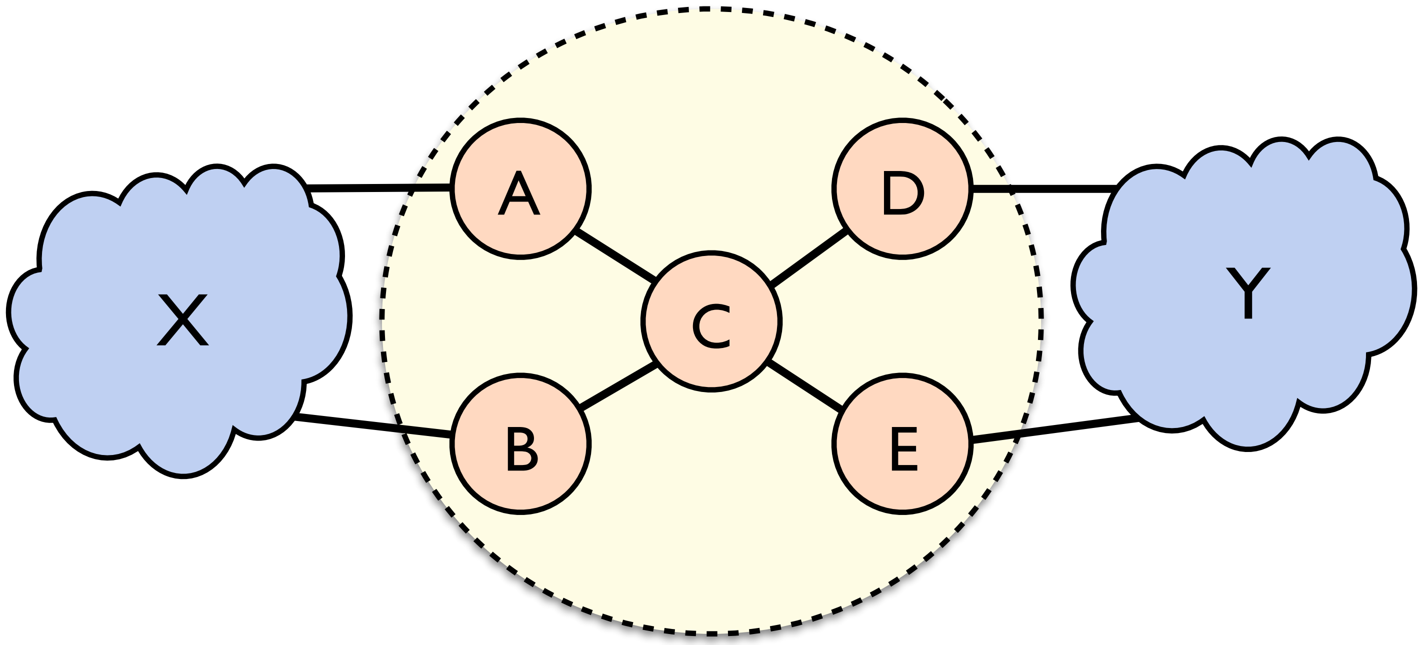


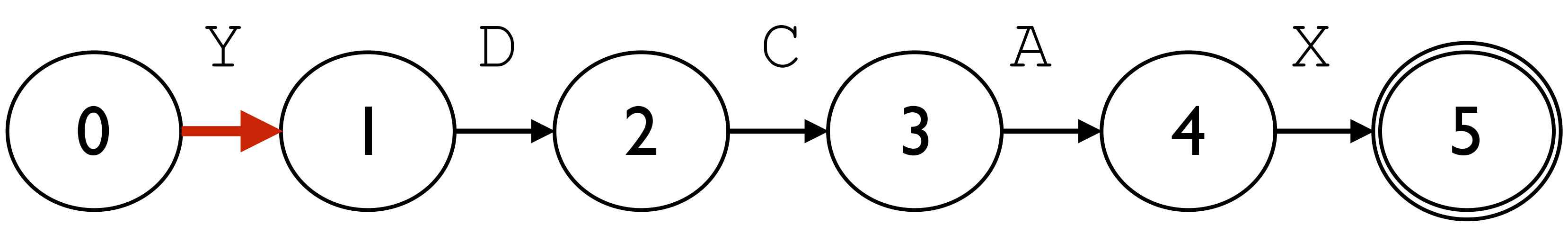


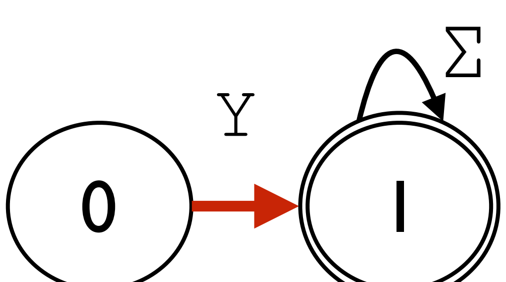


Constructing the Product Graph (PG)

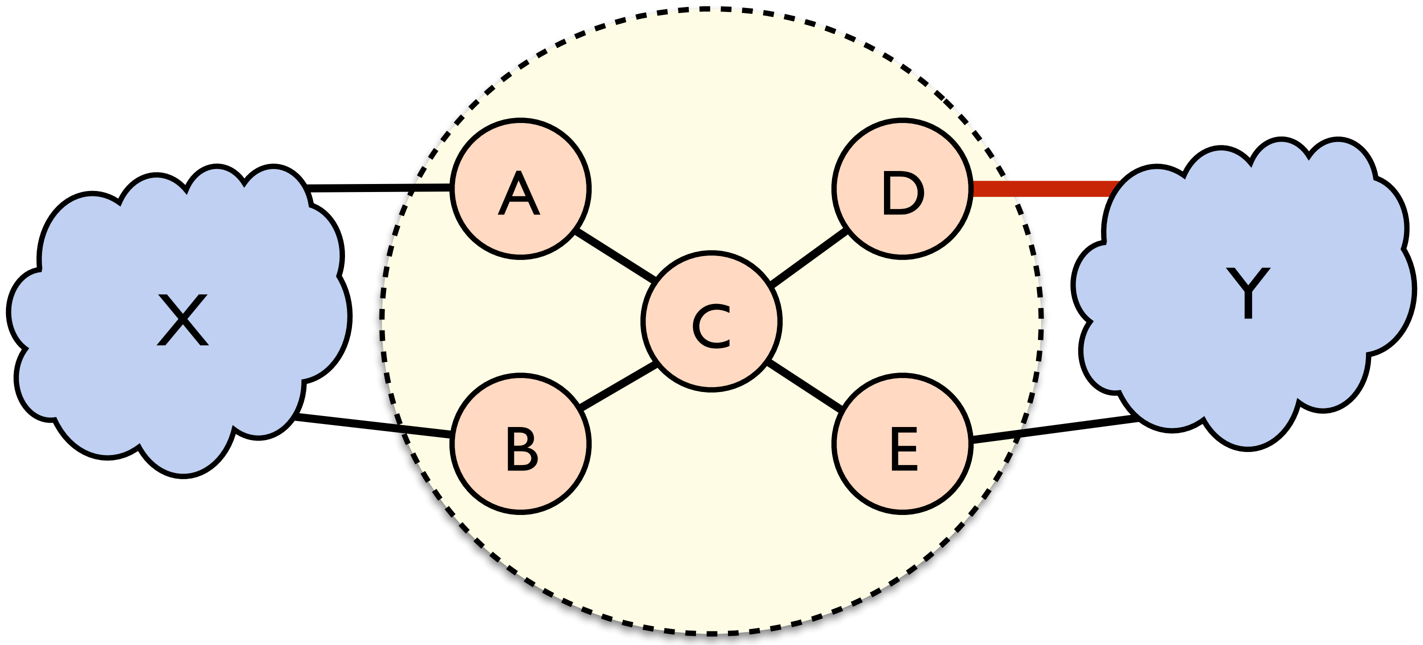


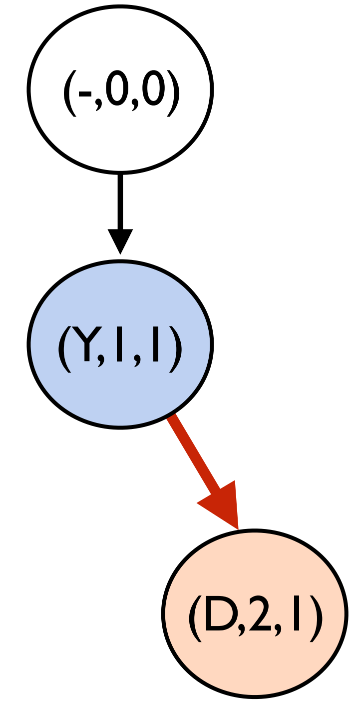


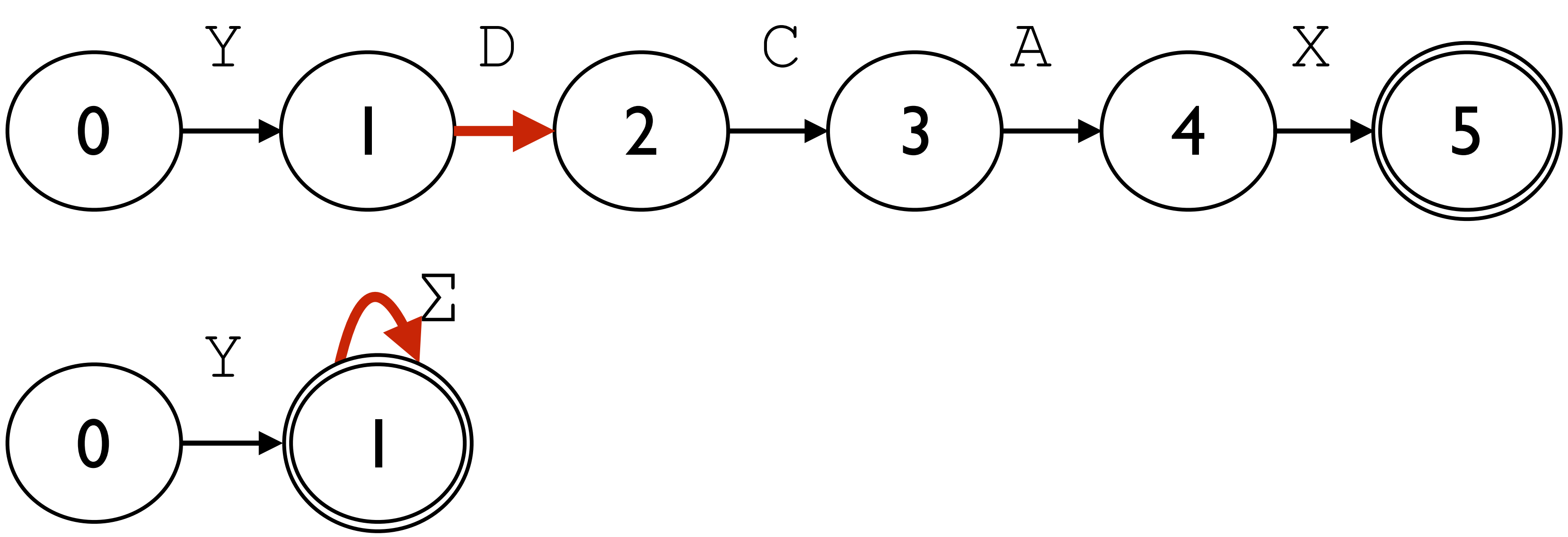




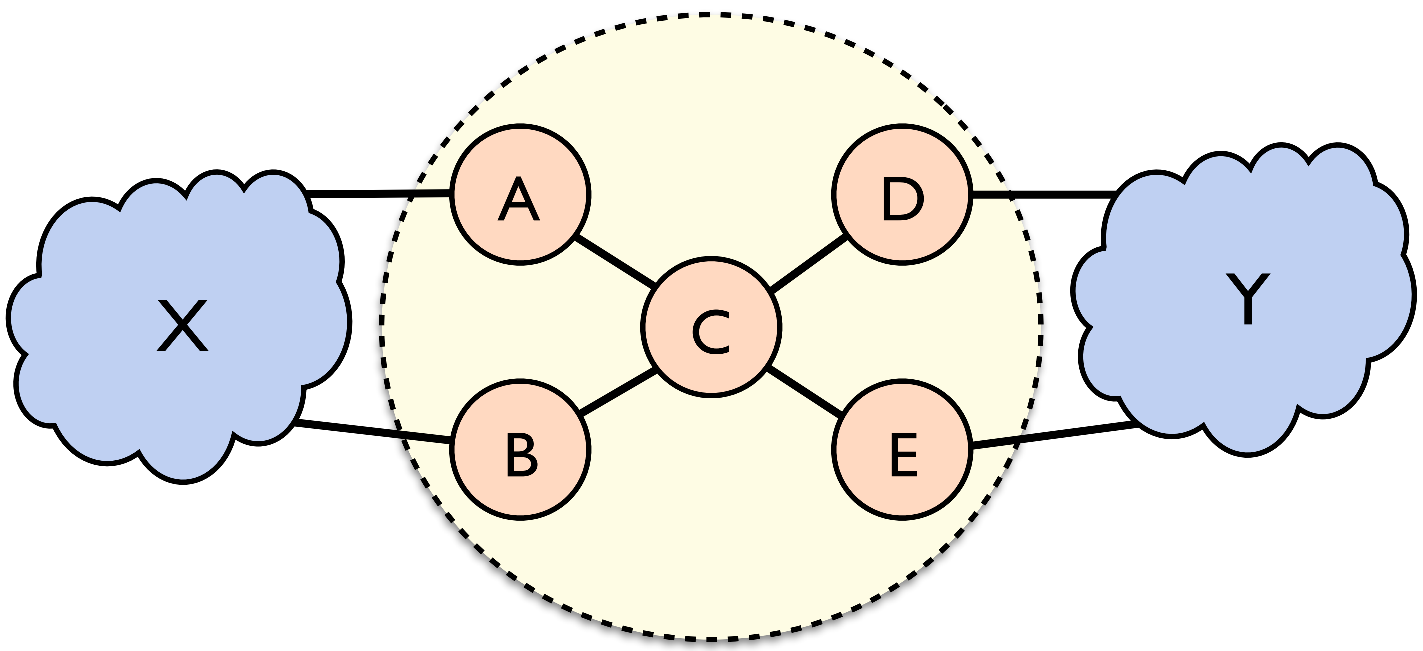
Constructing the Product Graph (PG)

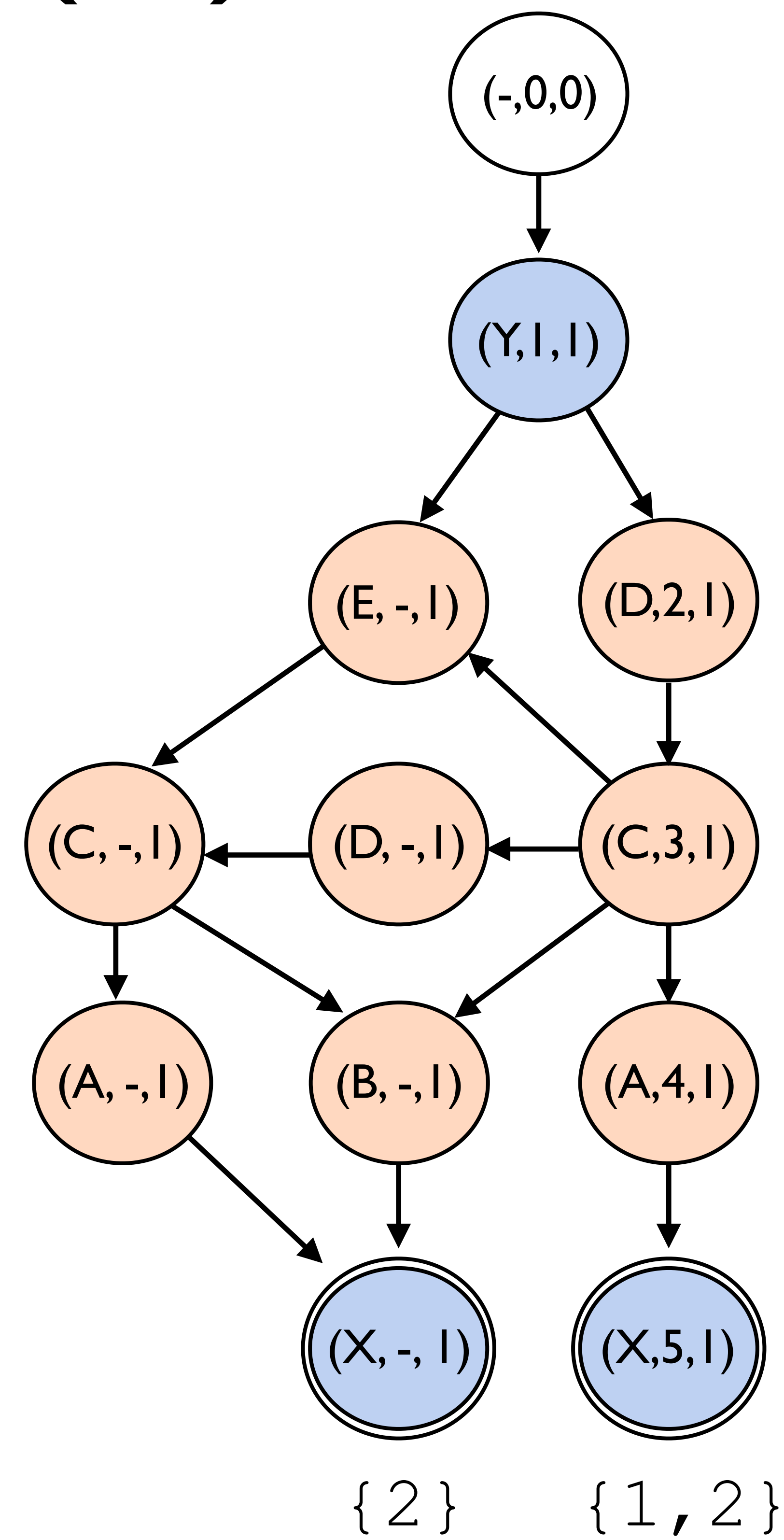






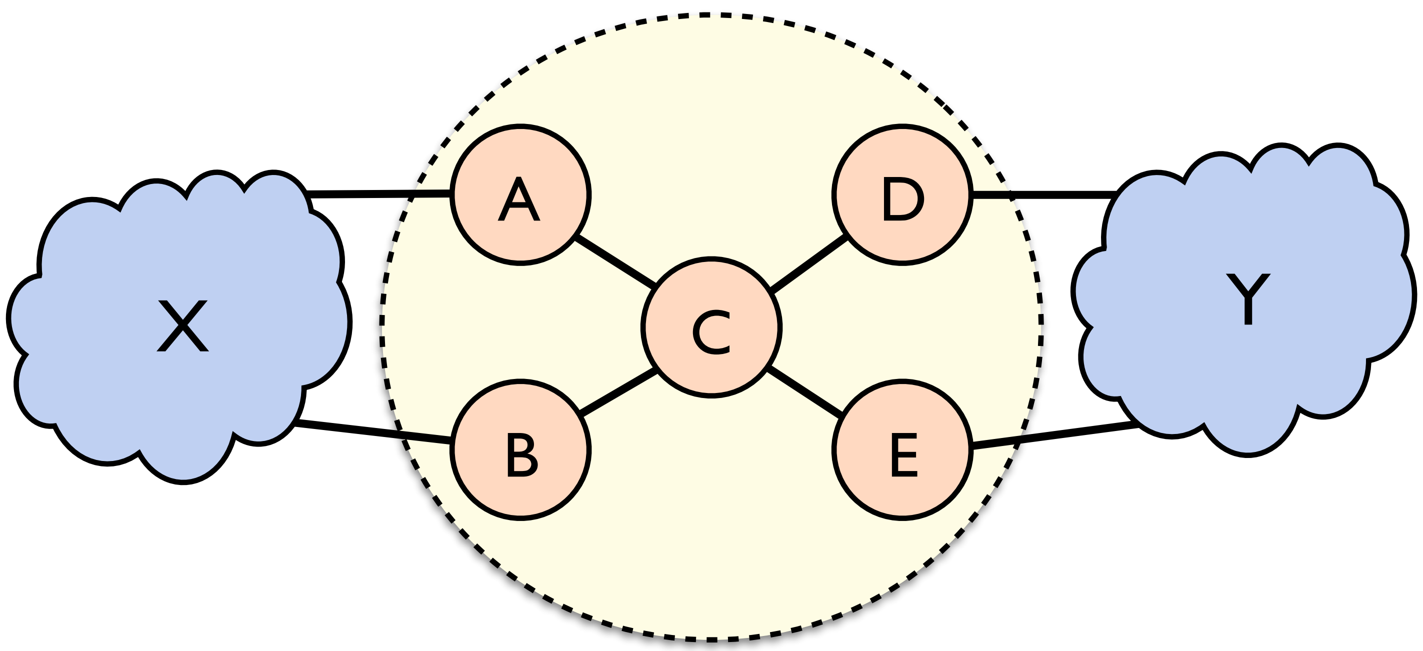
Constructing the Product Graph (PG)

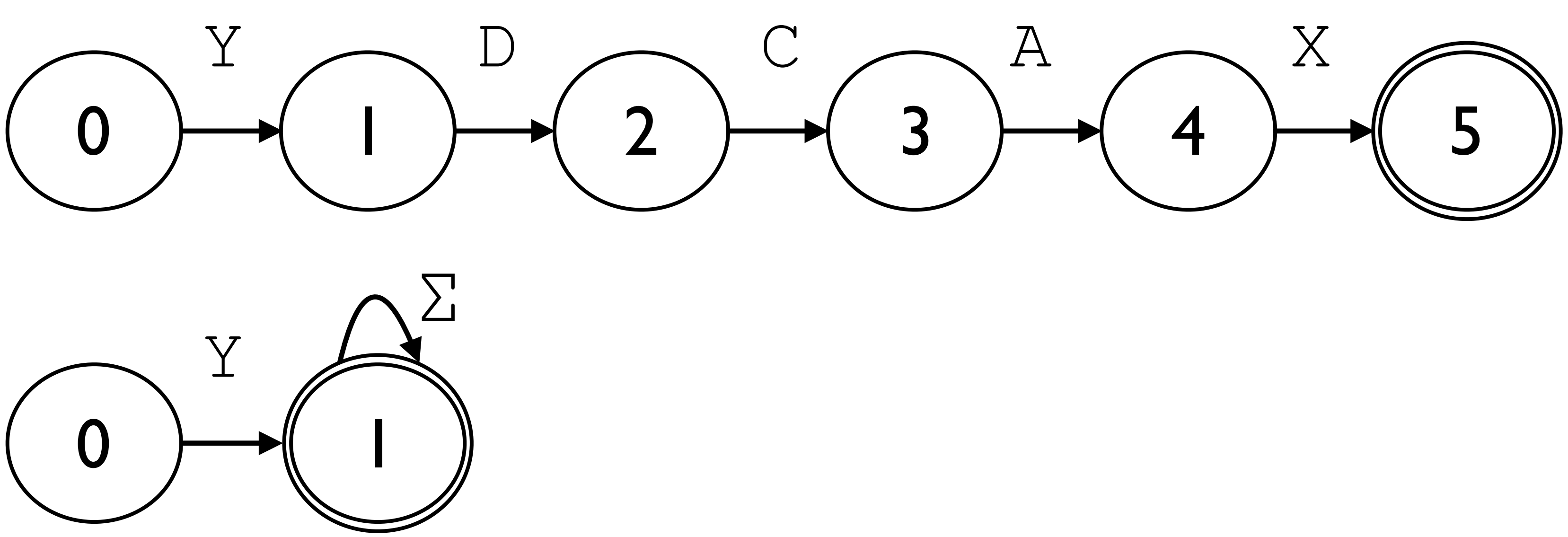


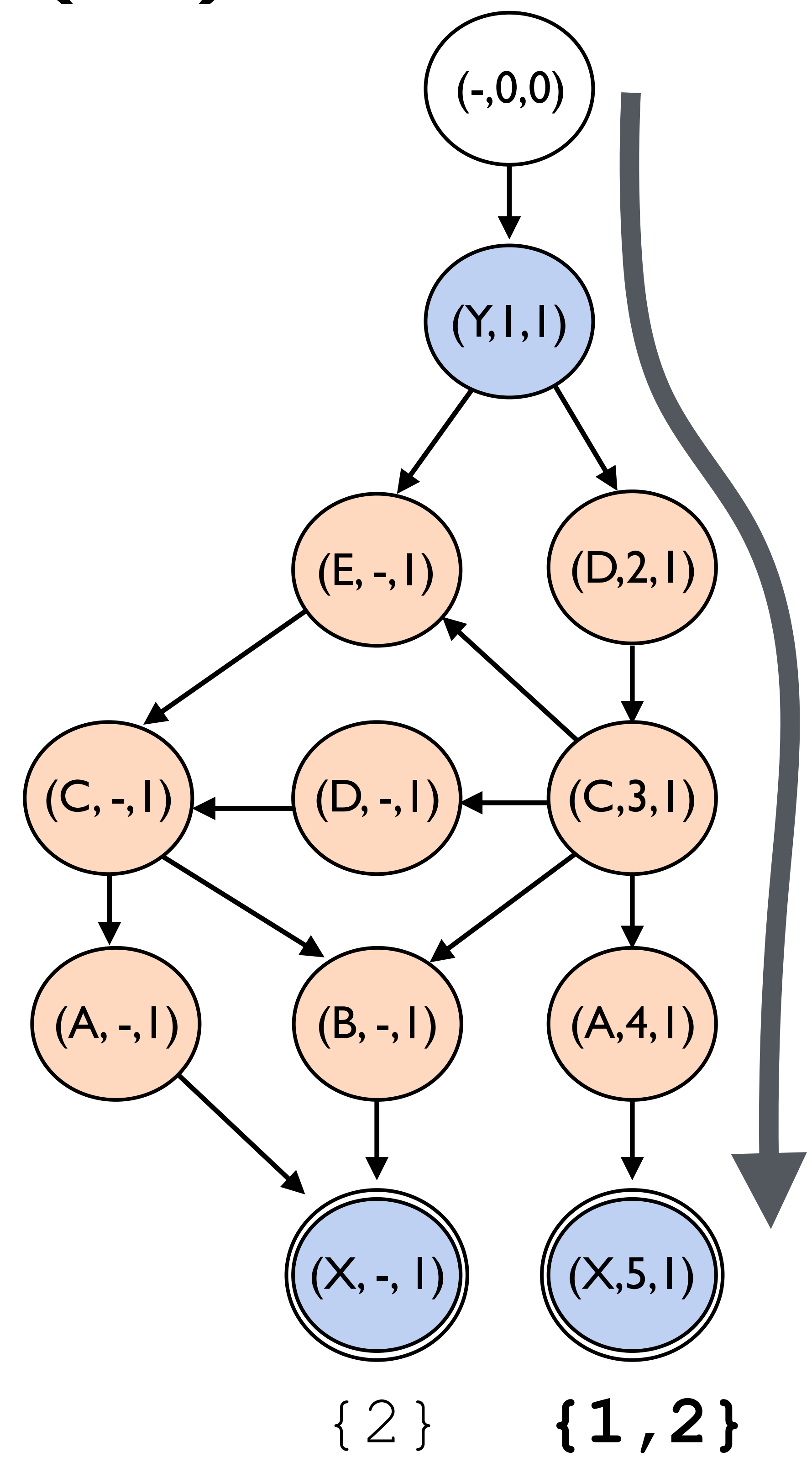


Graph capturing all possible policy-compliant paths through the topology

Constructing the Product Graph (PG)

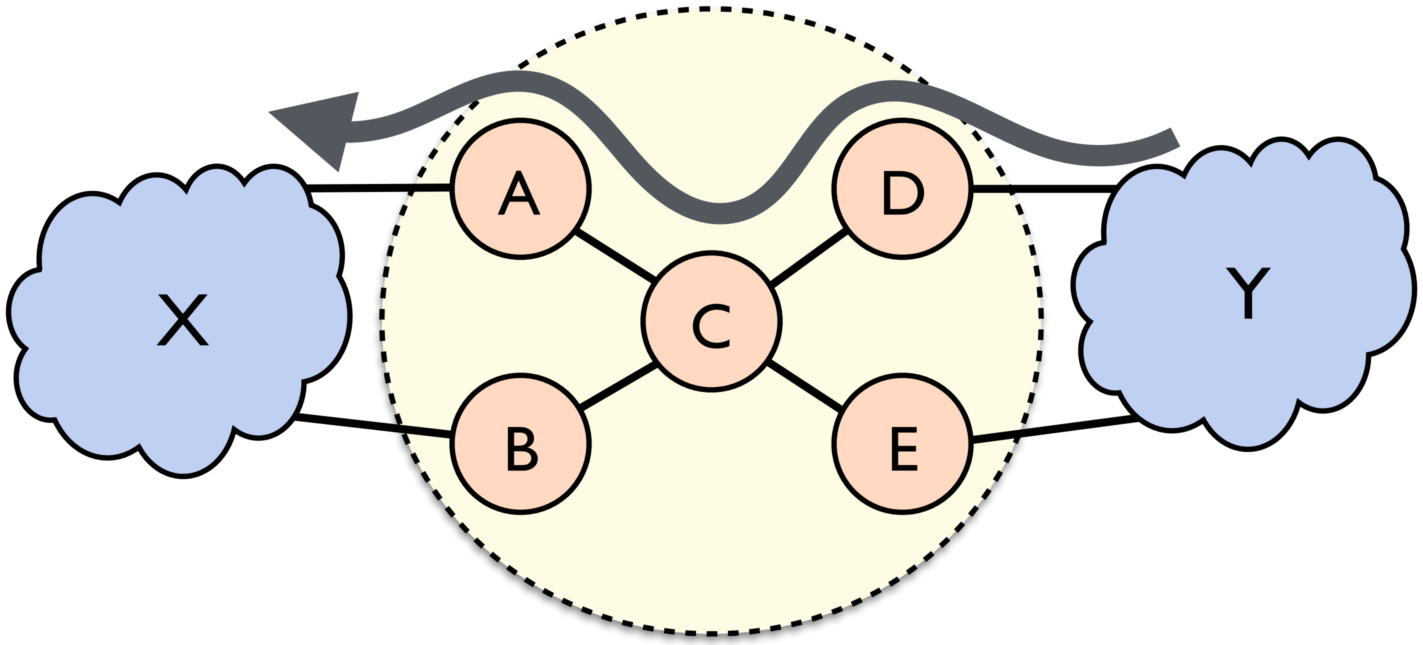


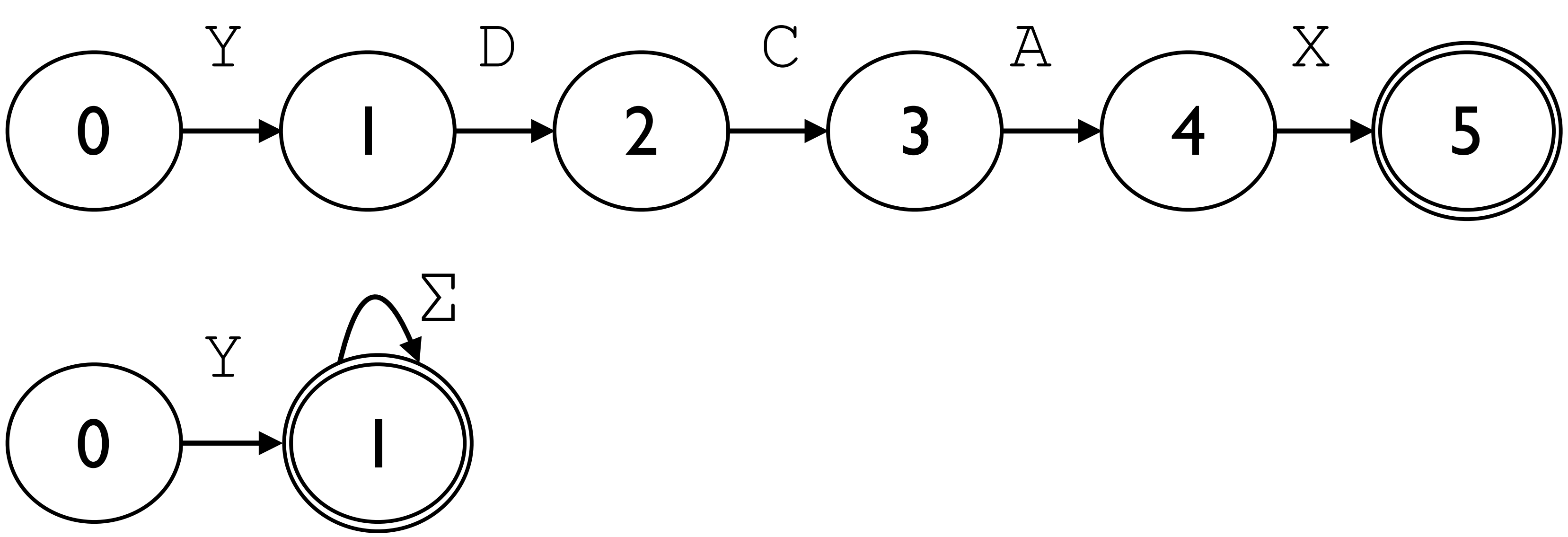


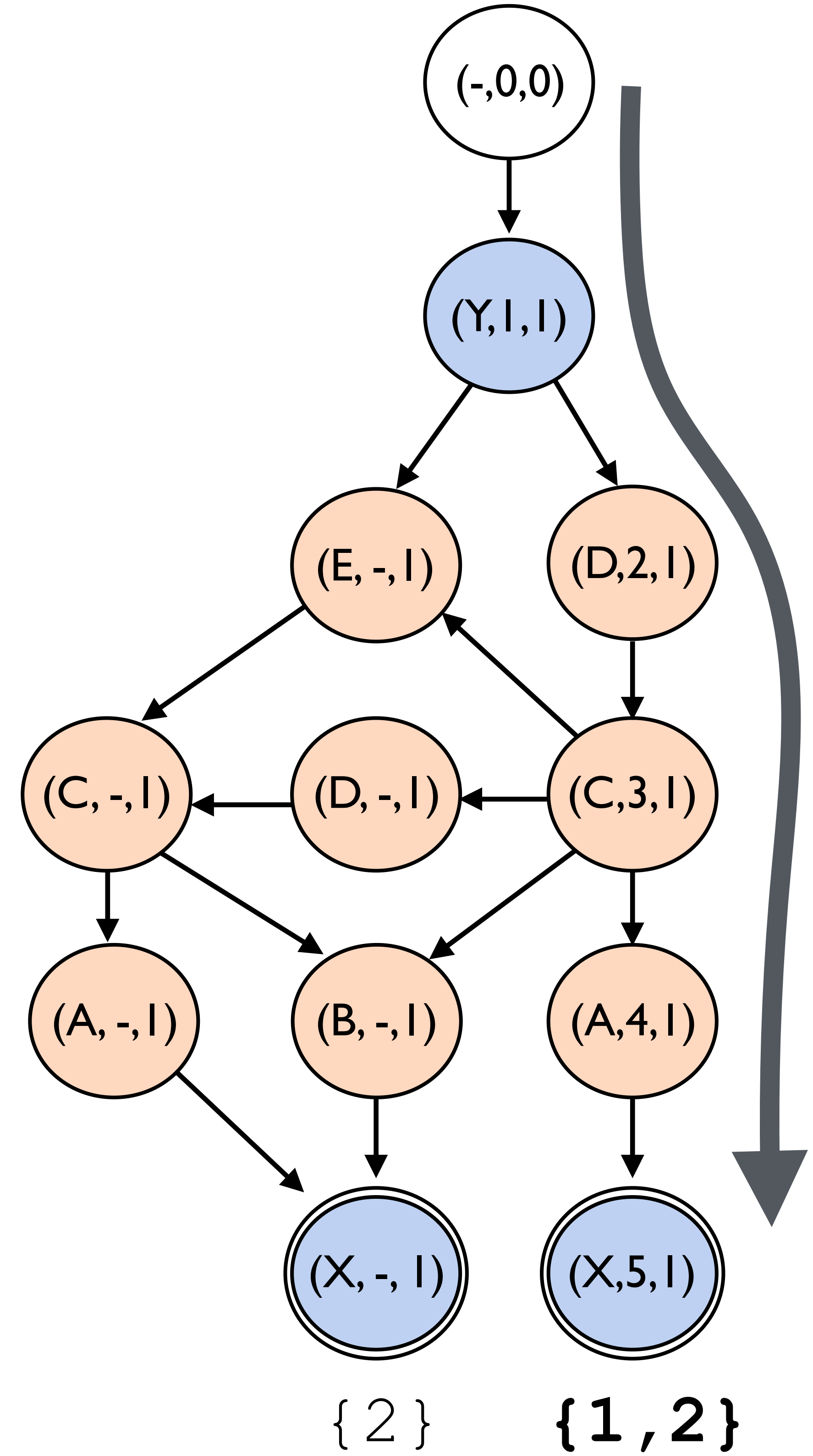


Preferences

Constructing the Product Graph (PG)

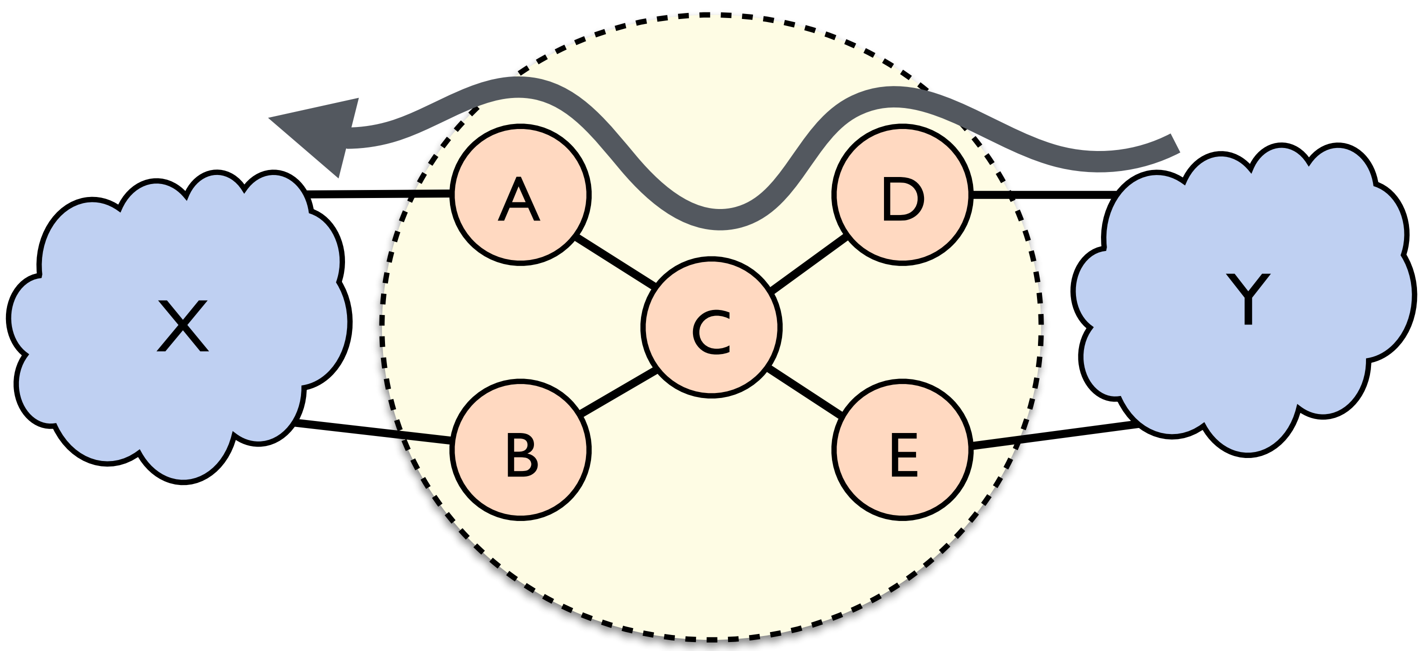


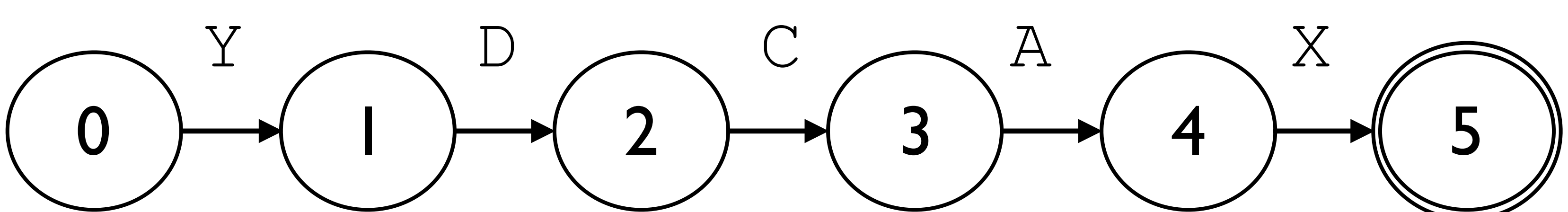


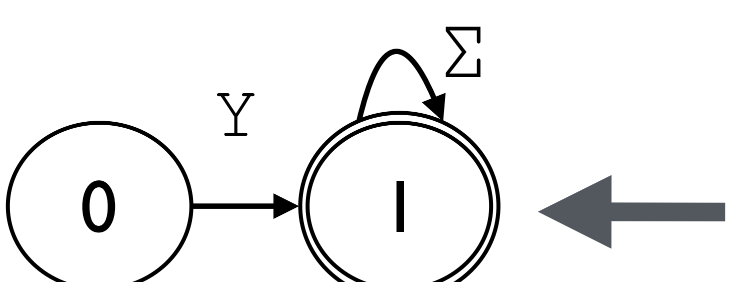


Preferences

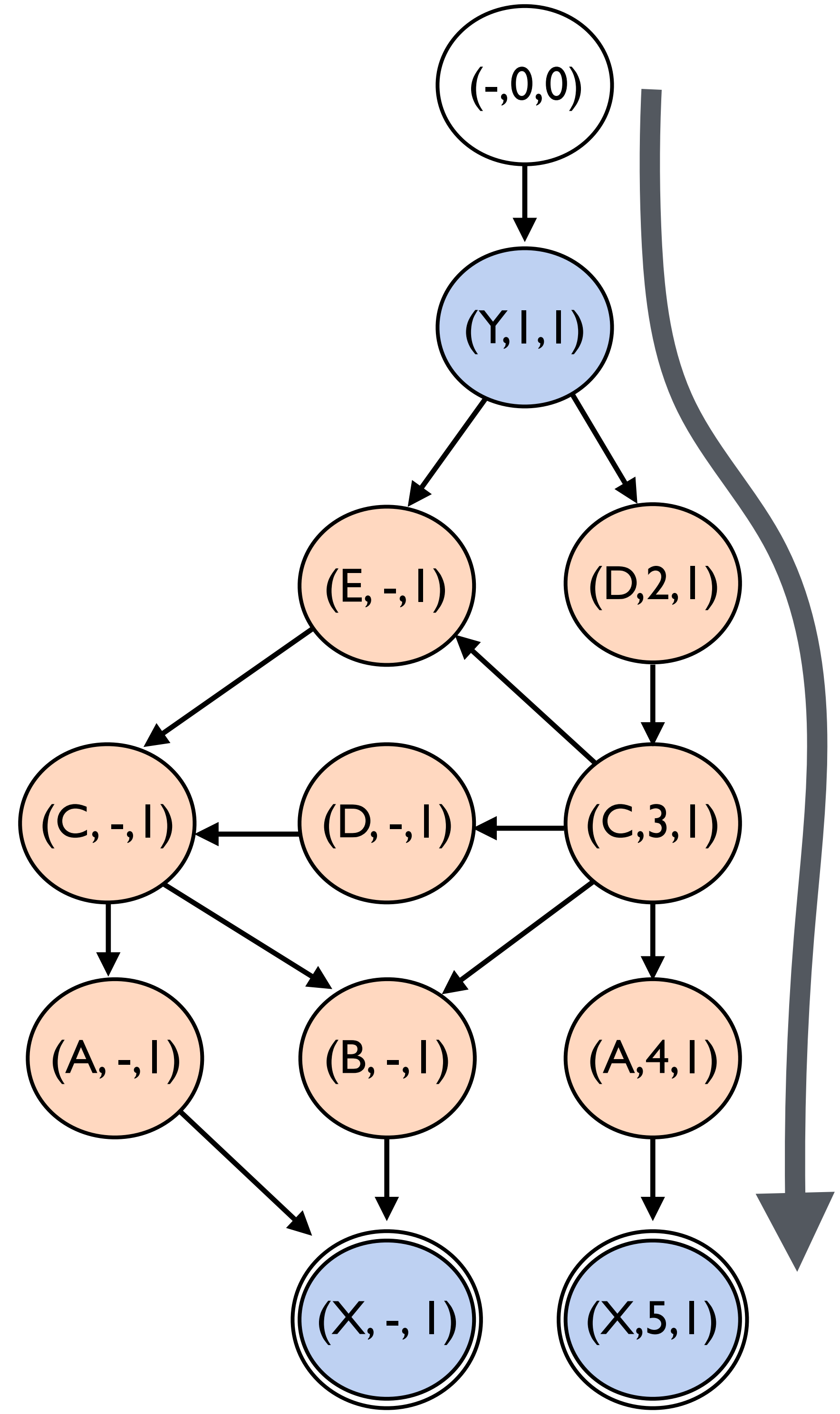
Constructing the Product Graph (PG)







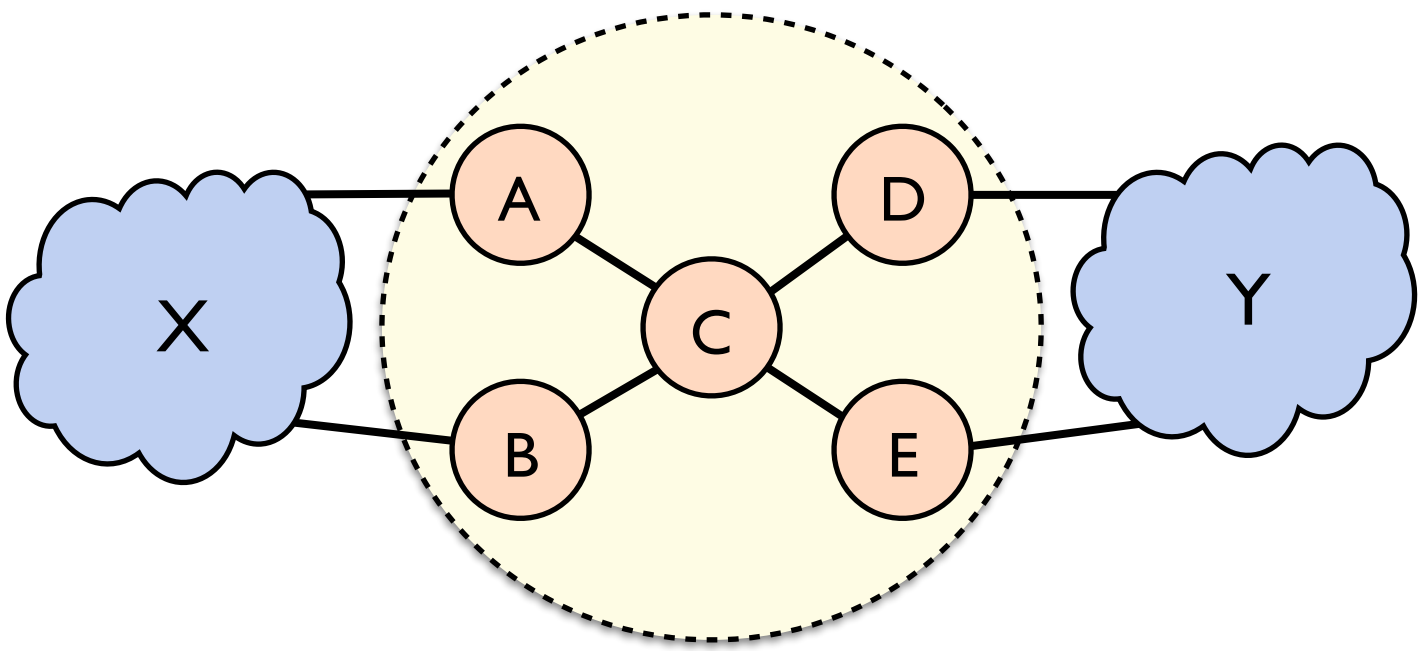
Accept: Y D C A X

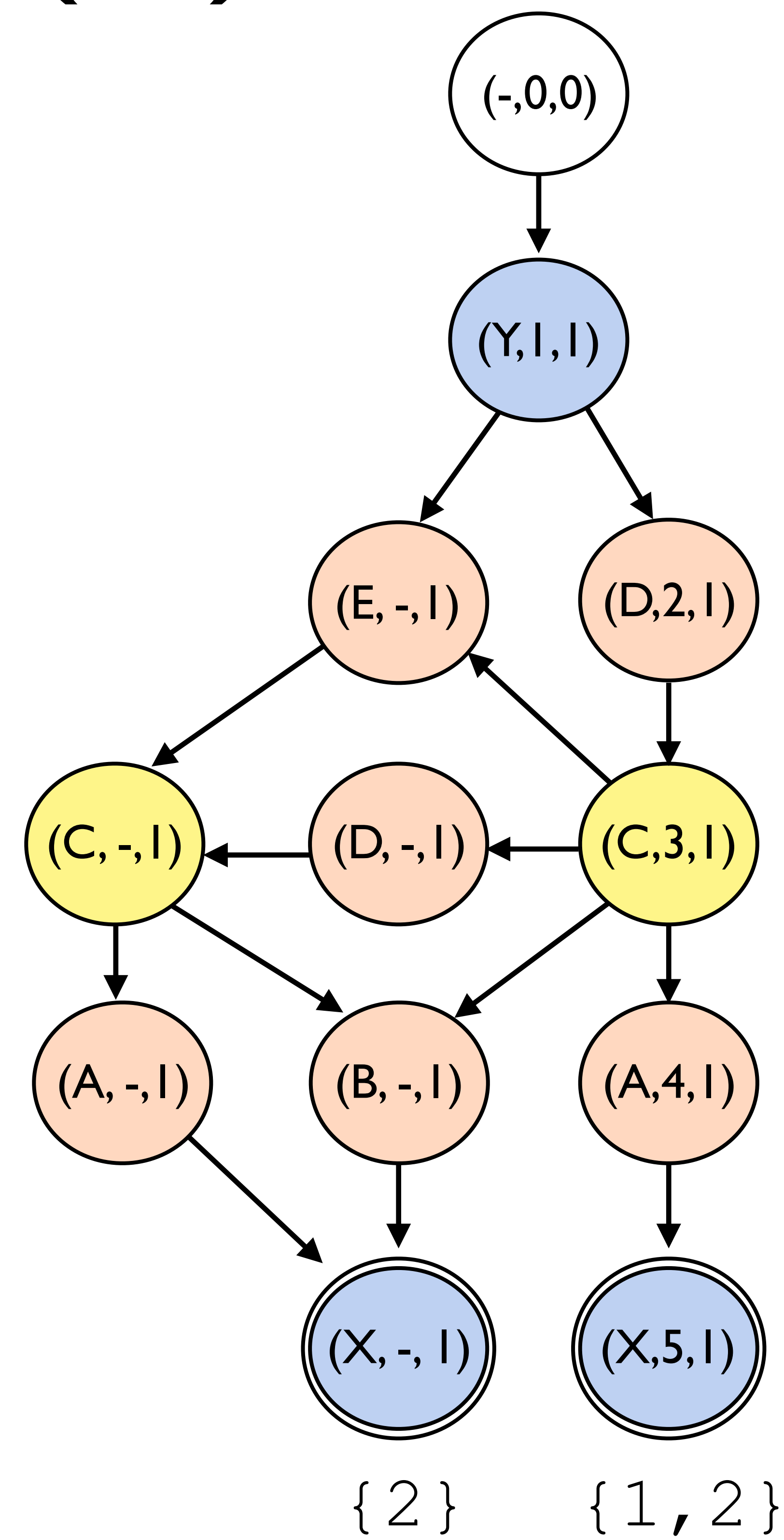


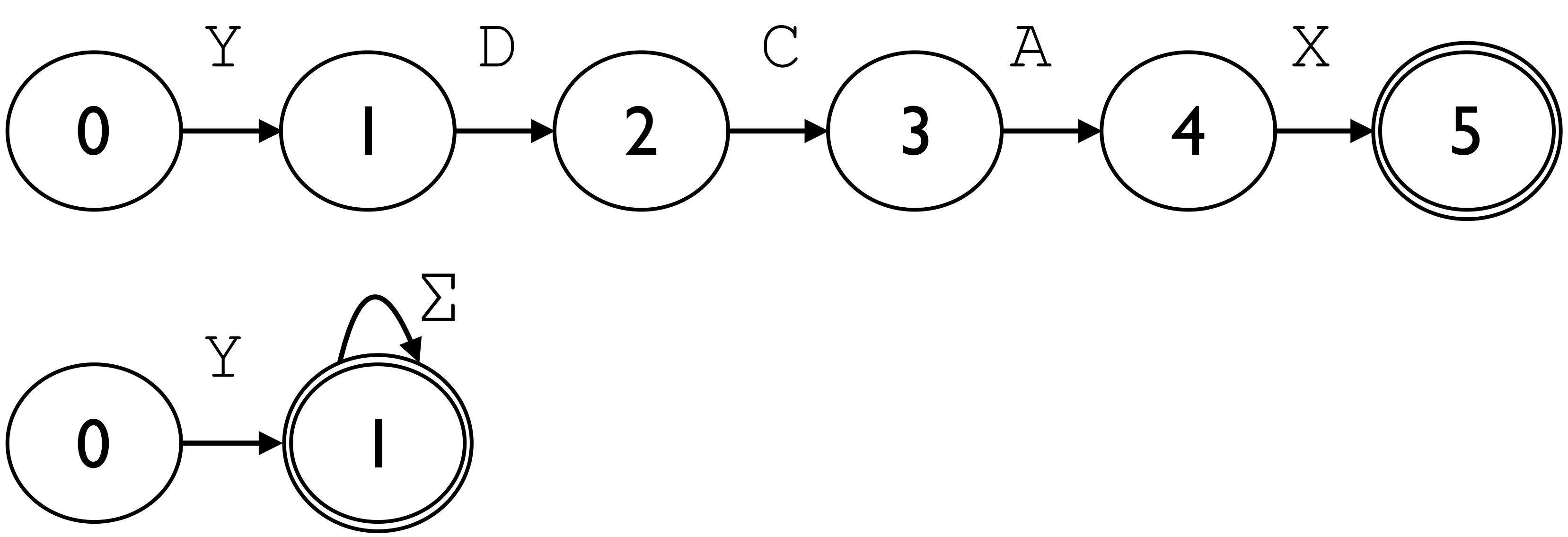
{2} **{1,2}**

Preferences

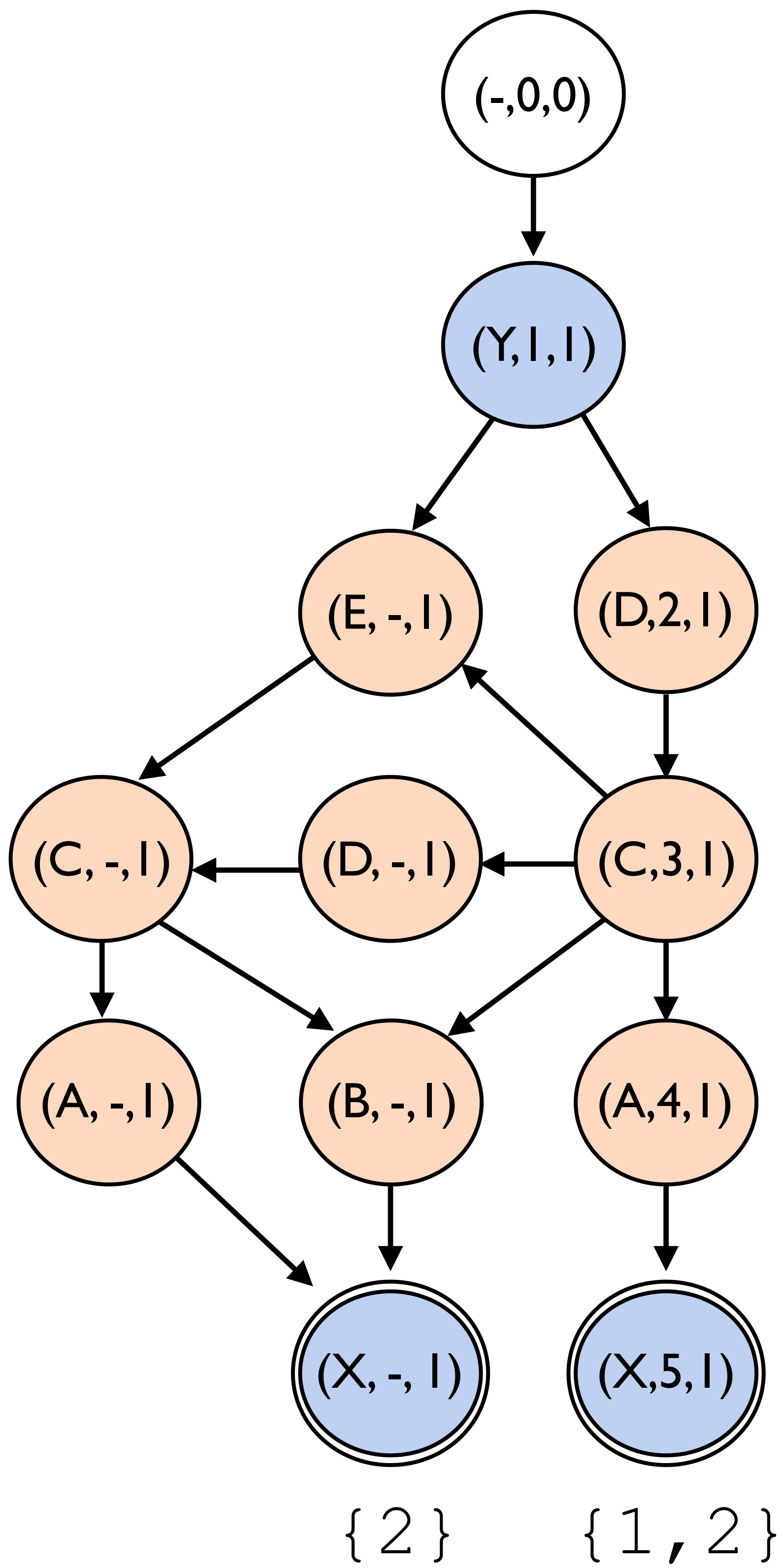
Constructing the Product Graph (PG)







Compilation to BGP:

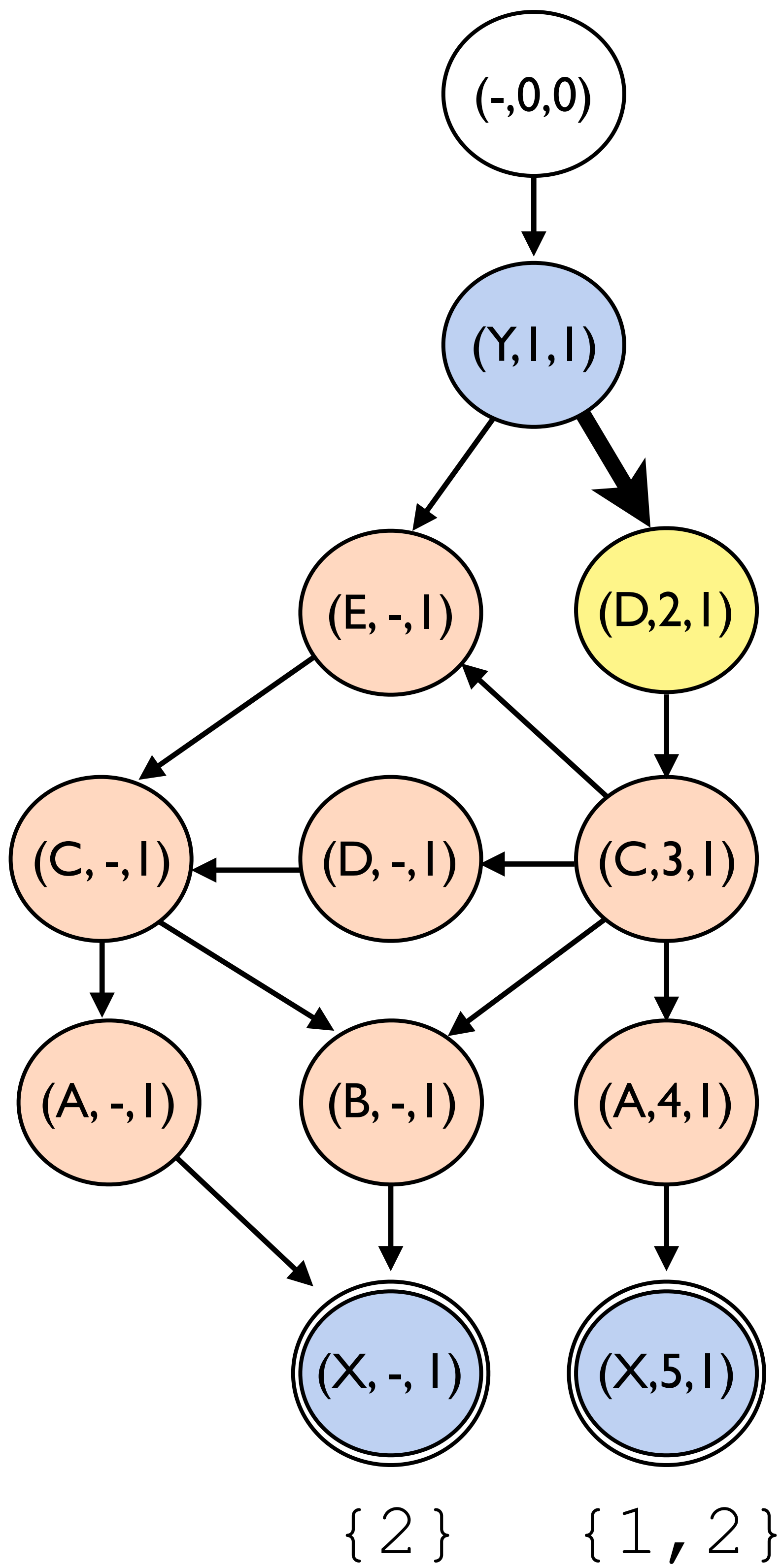


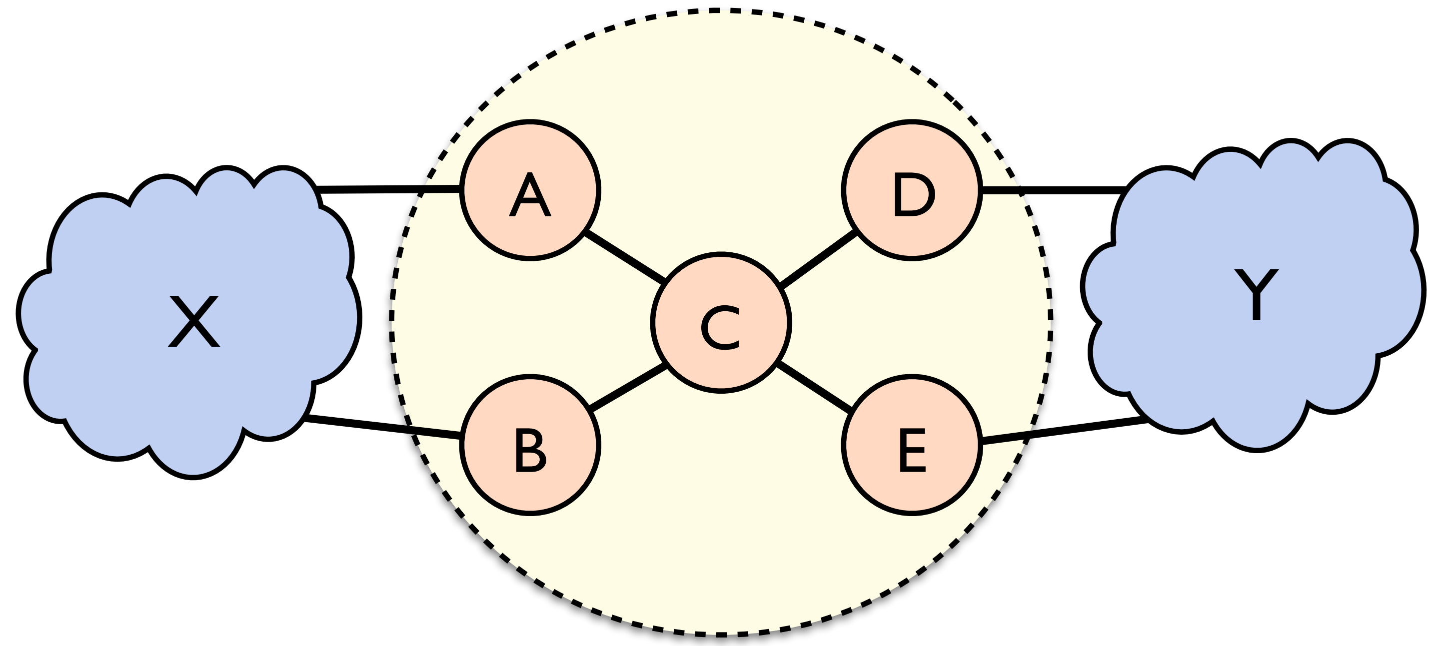
Idea 1: Restrict advertisements to edges

- Encode state in a BGP community tag
- Incoming edges — import filters
- Outgoing edges — export filters

Let BGP find **some allowed** path dynamically

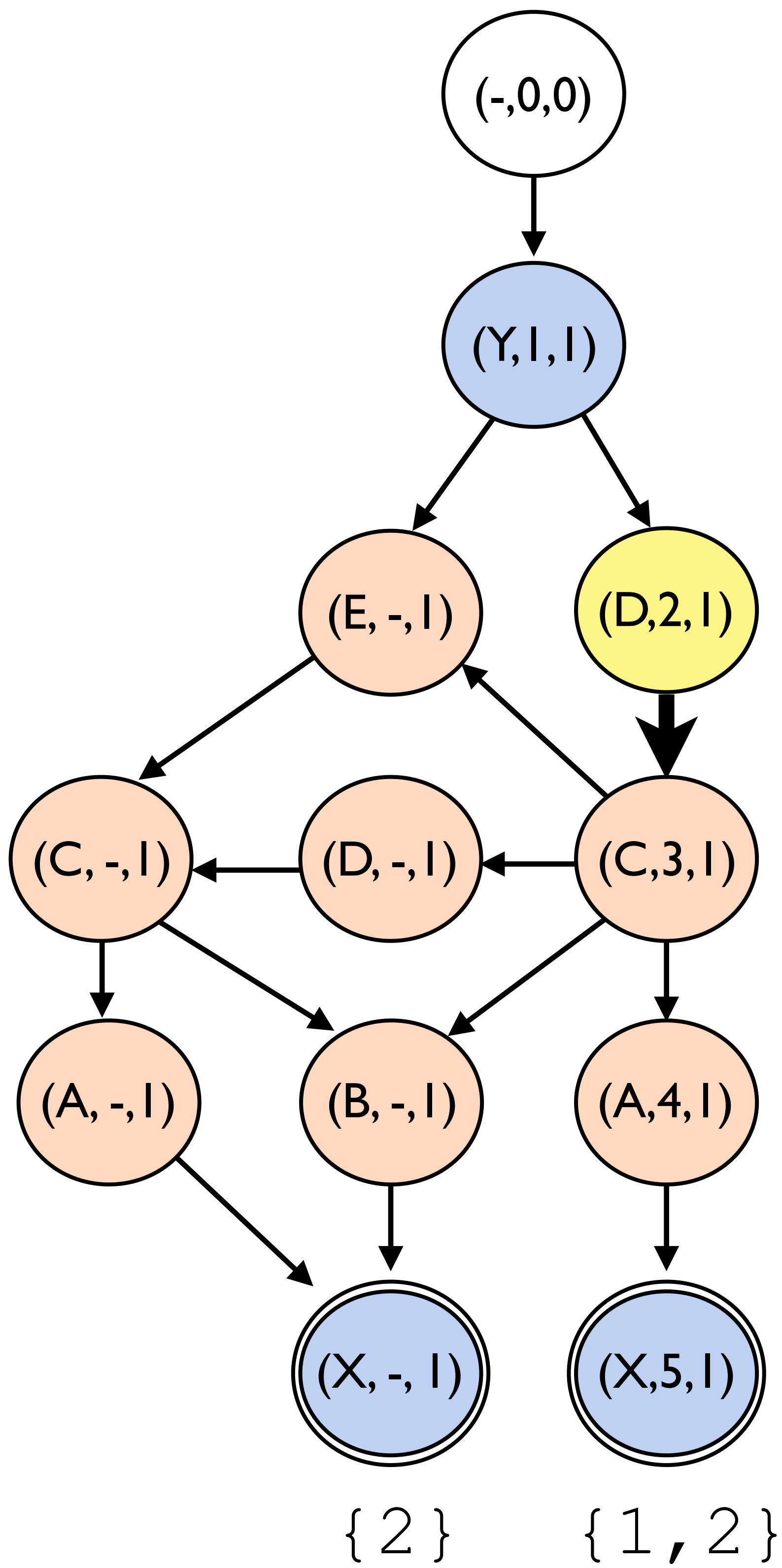
Compilation to BGP:

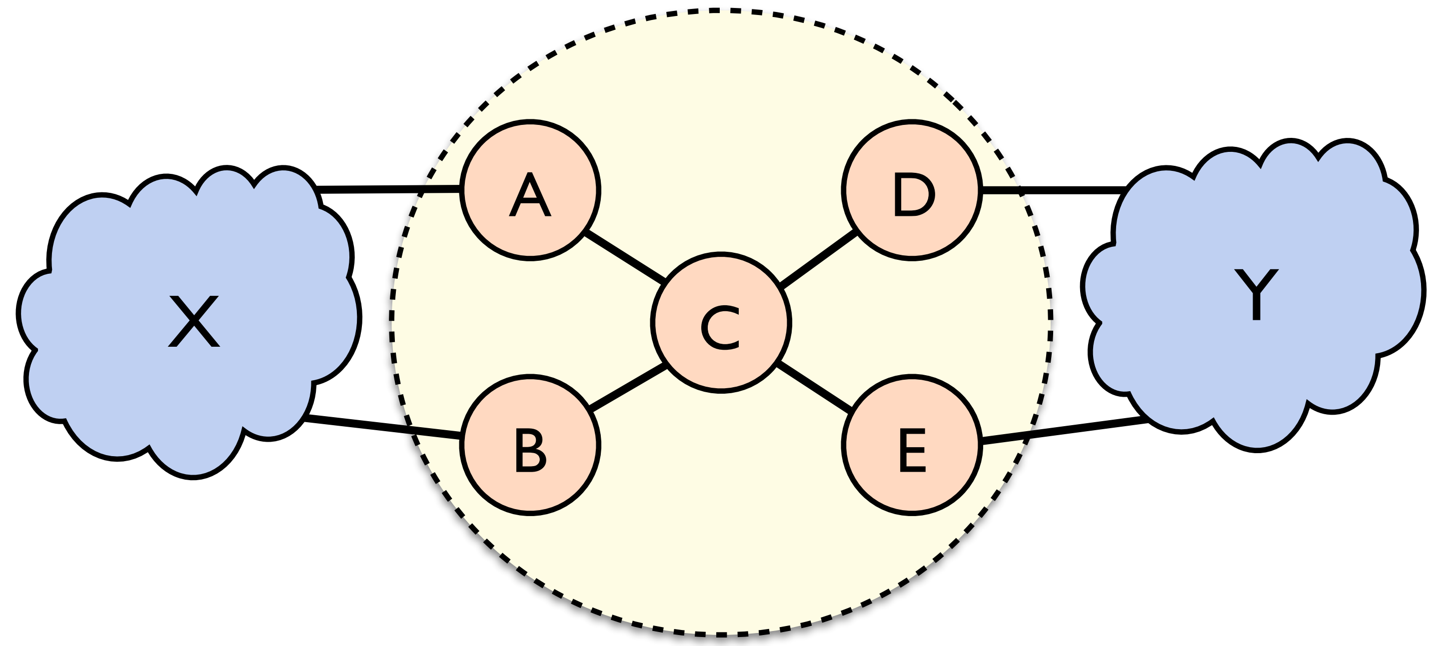




D allows import matching regex(Y)

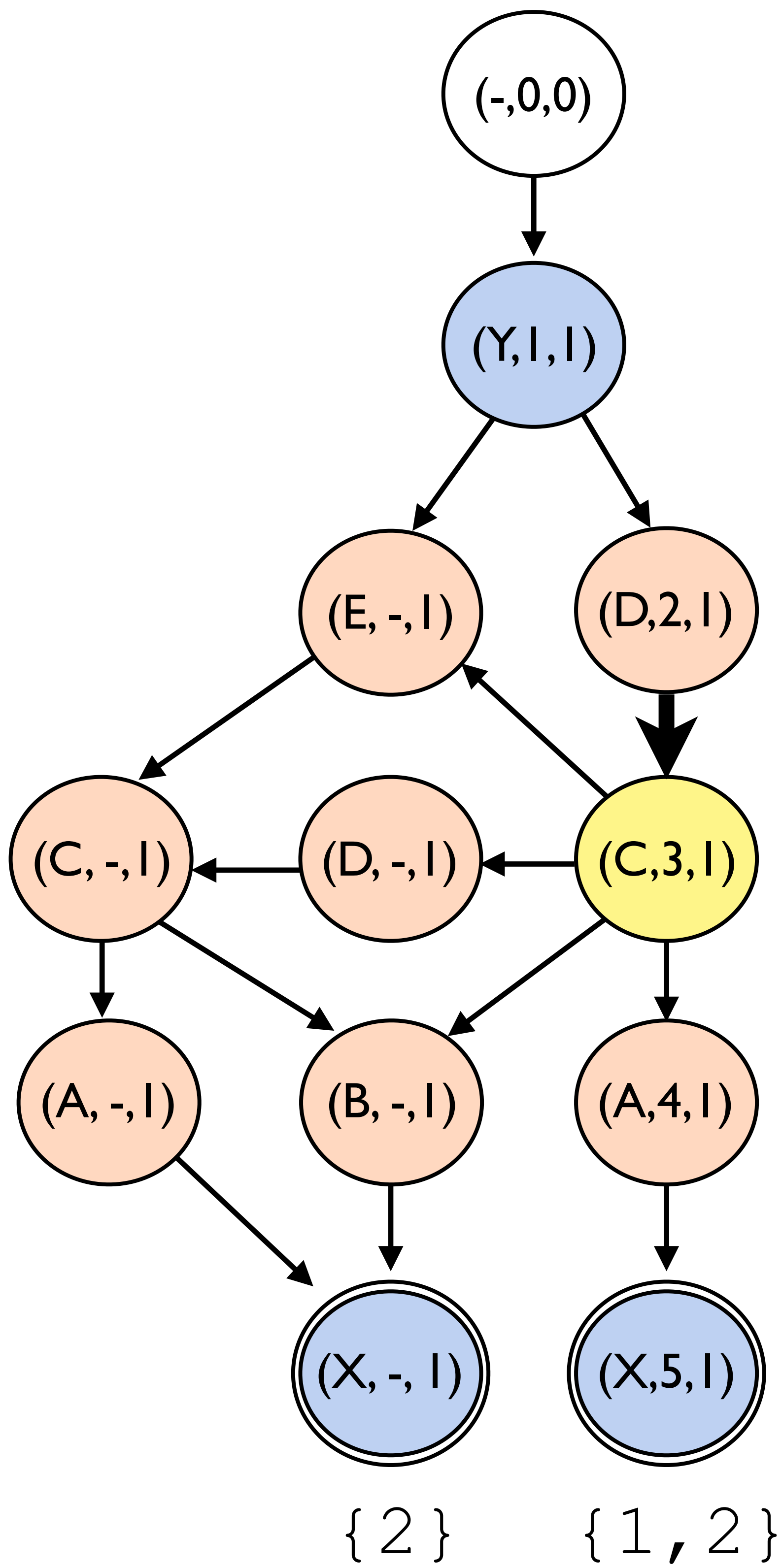
Compilation to BGP:

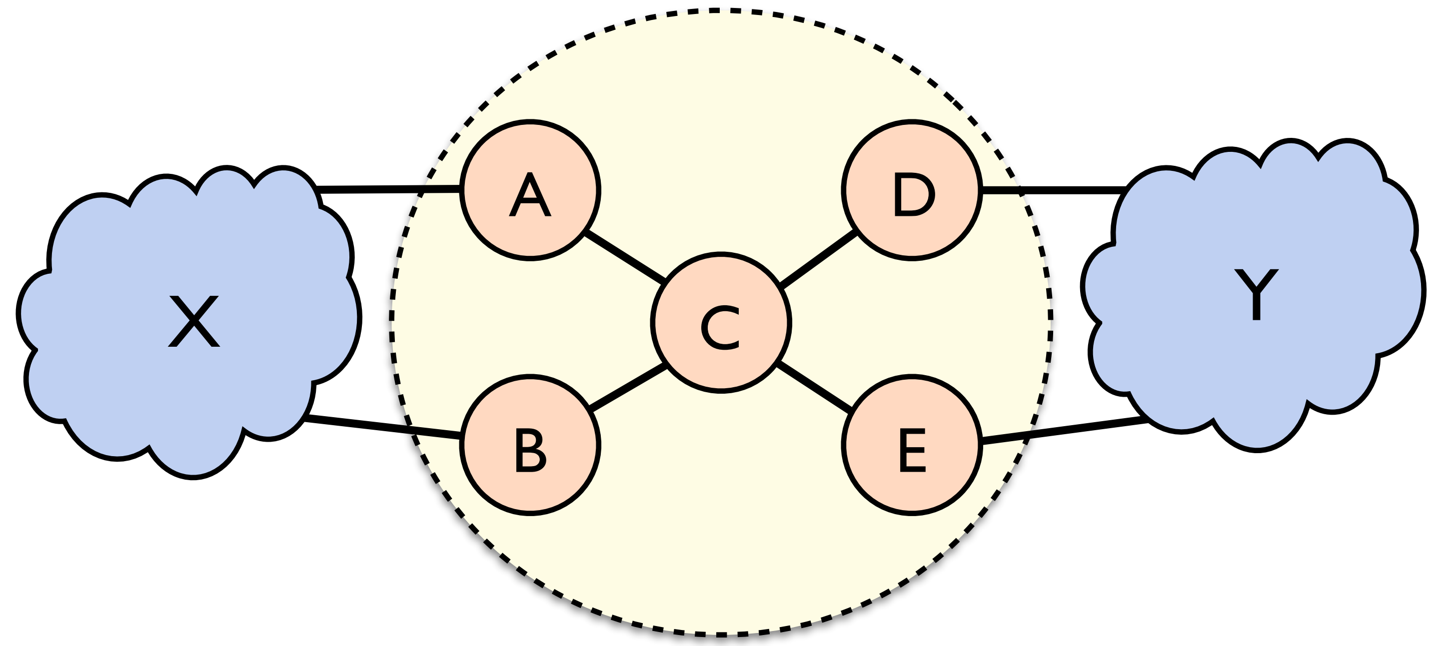




D exports to C with tag (2,1)

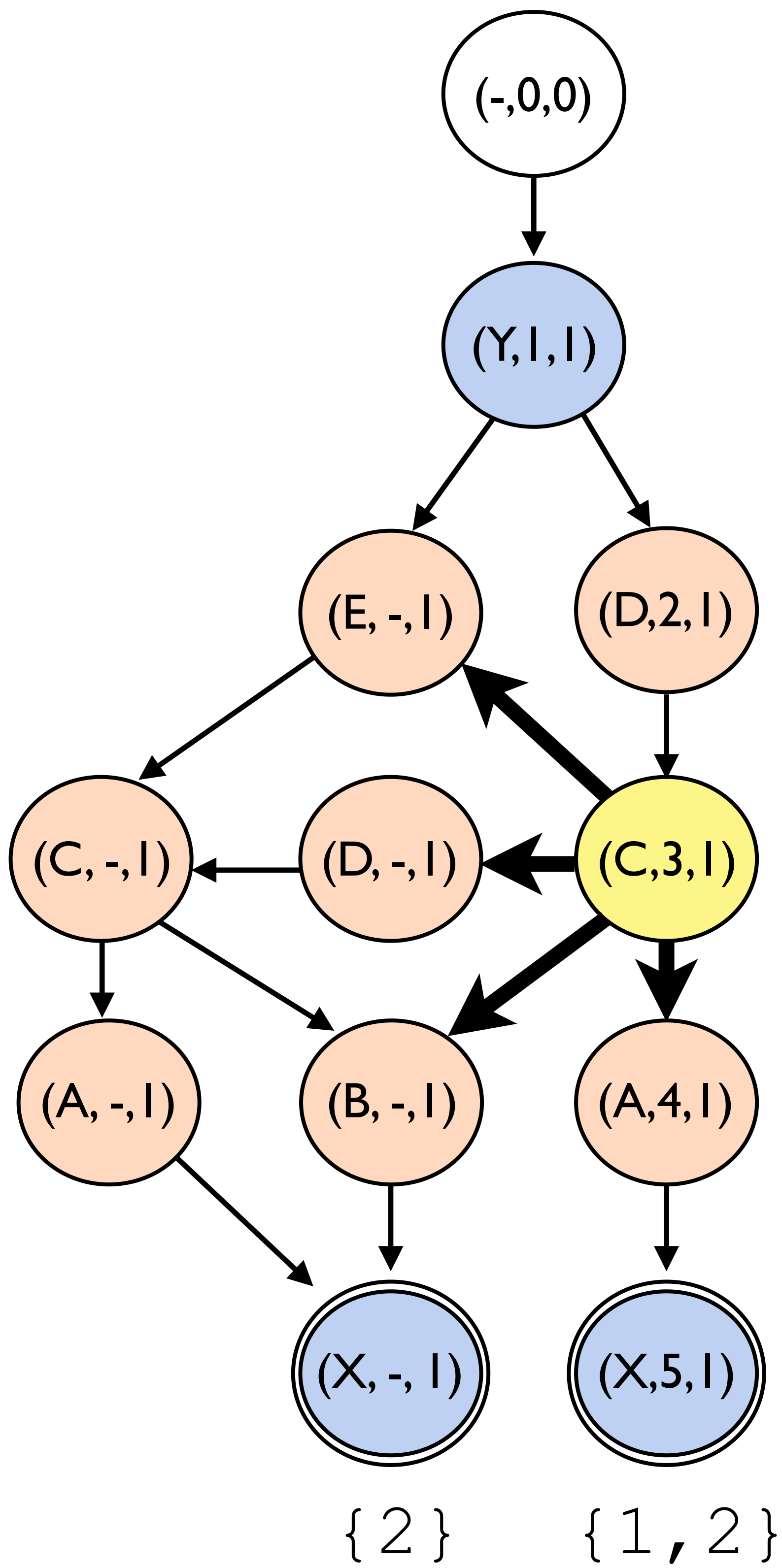
Compilation to BGP:

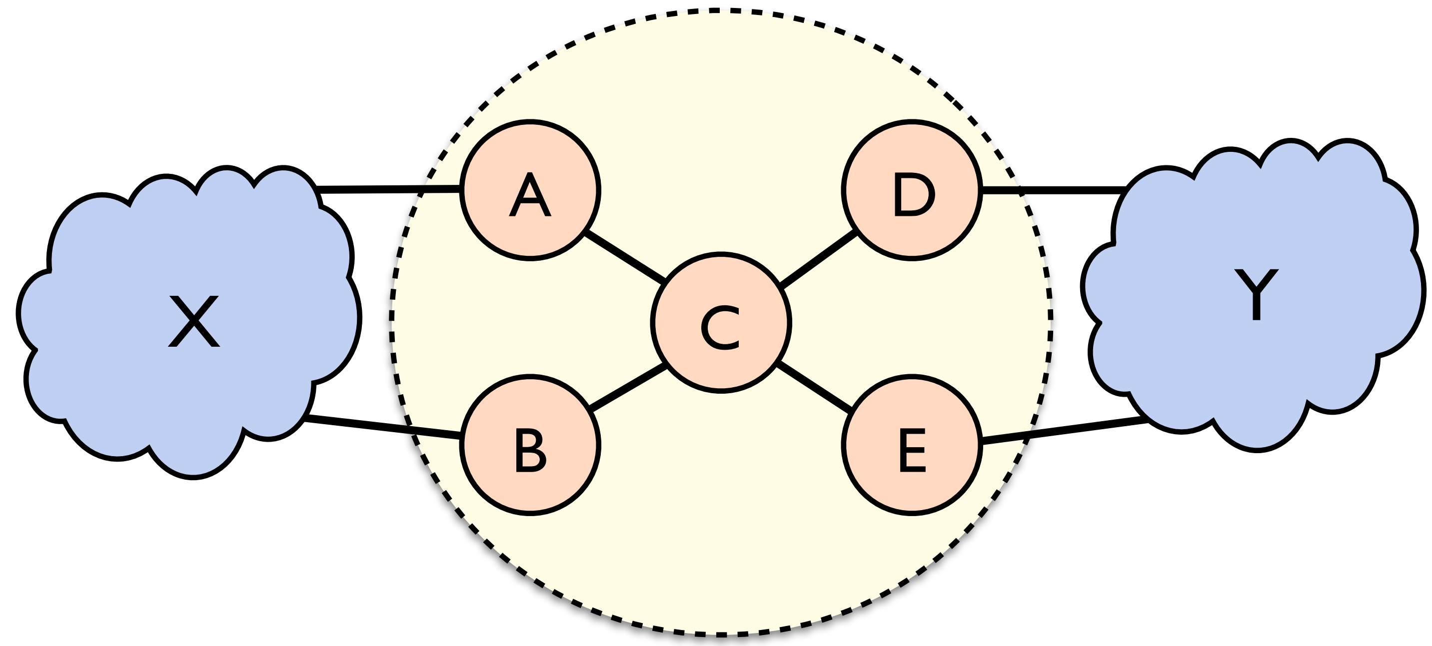




C allows import from D with tag (2,1)

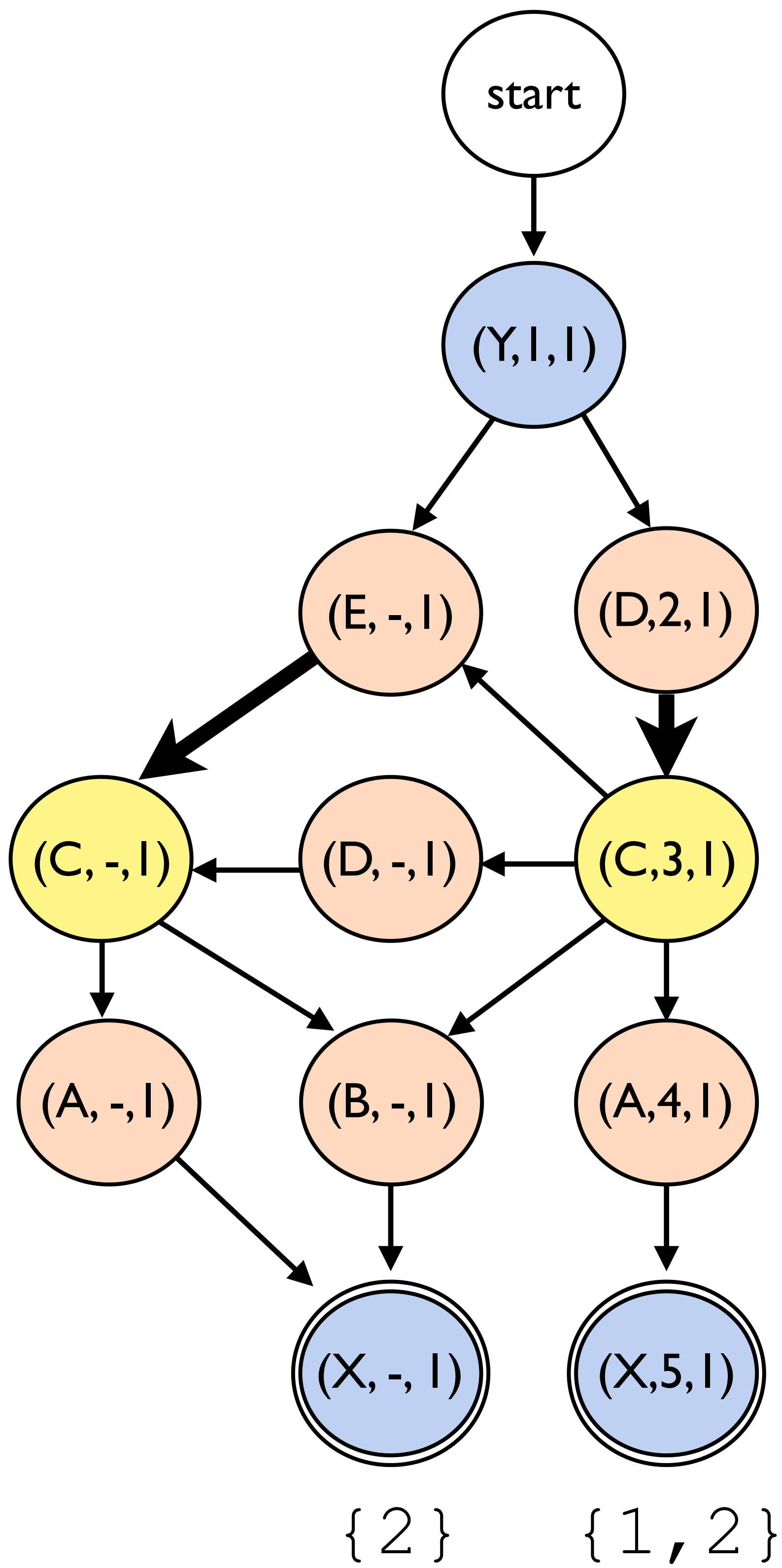
Compilation to BGP:





C exports to A,B,D,E with tag (3,1)

Compilation to BGP:

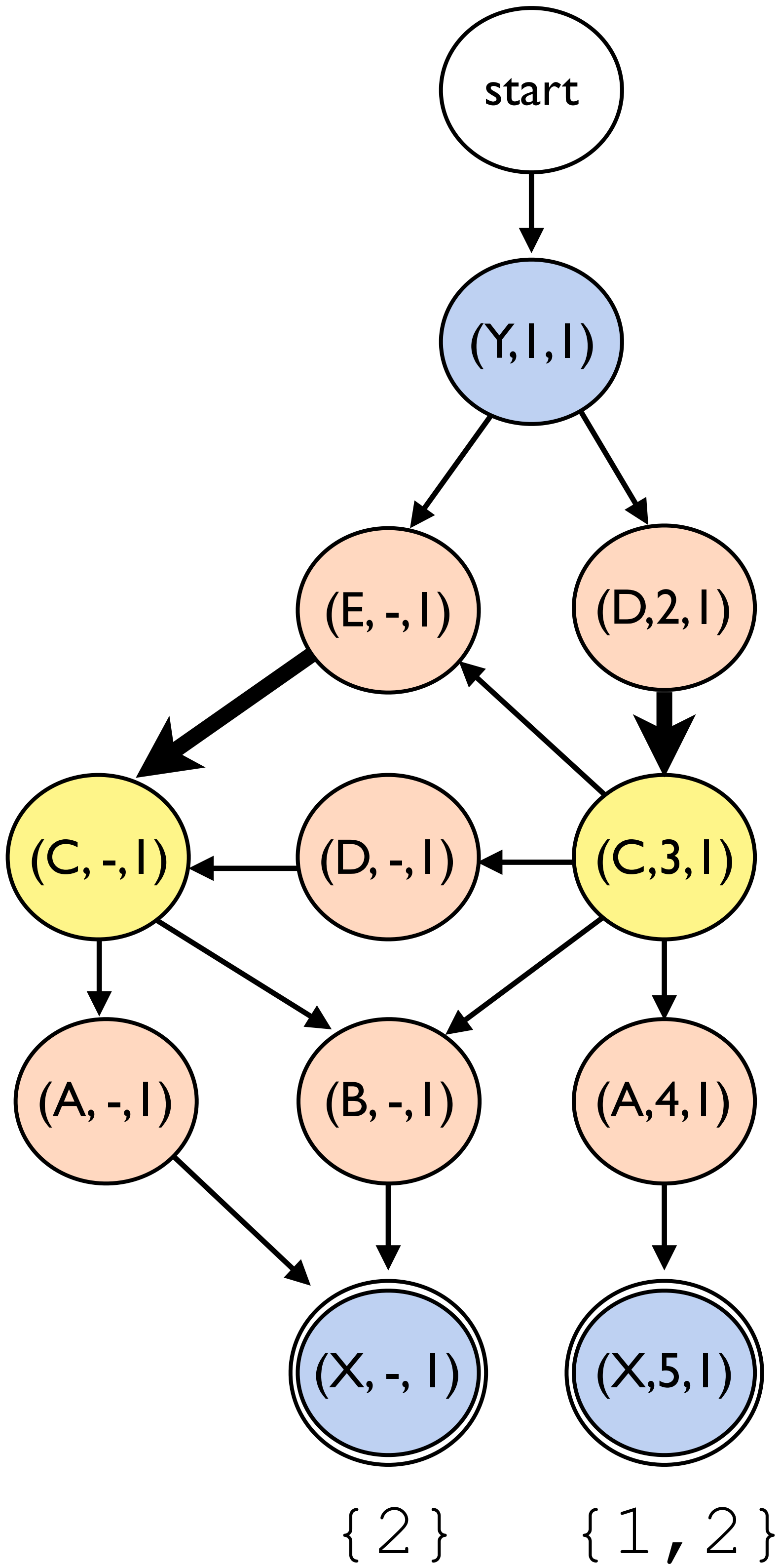


Idea 2: Find preferences

- Direct BGP towards best path
- Under all combinations of failures

Let BGP find **the best** path dynamically

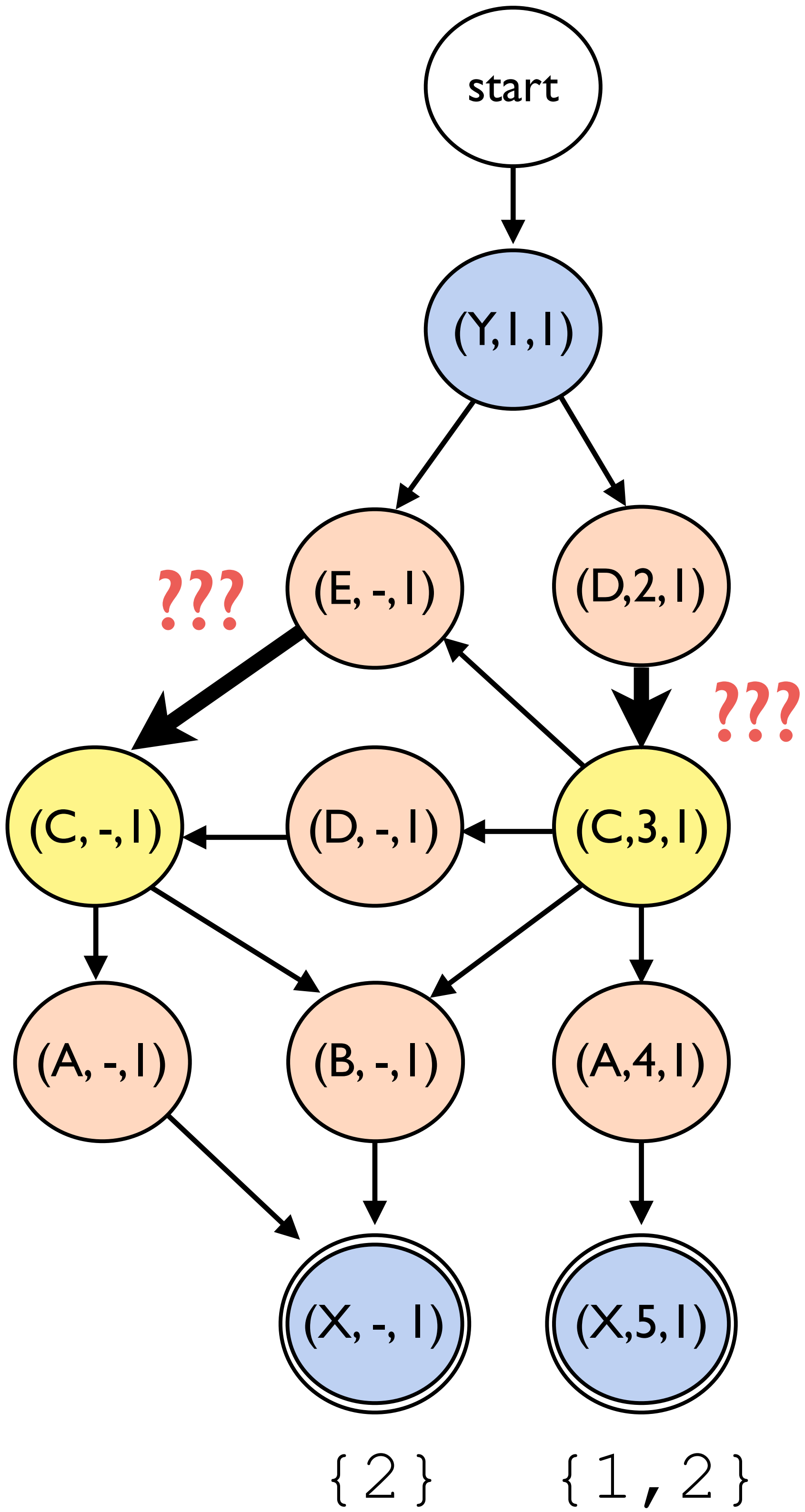
Compilation to BGP:



Router C

```
match peer = D ...

match peer = E ...
```

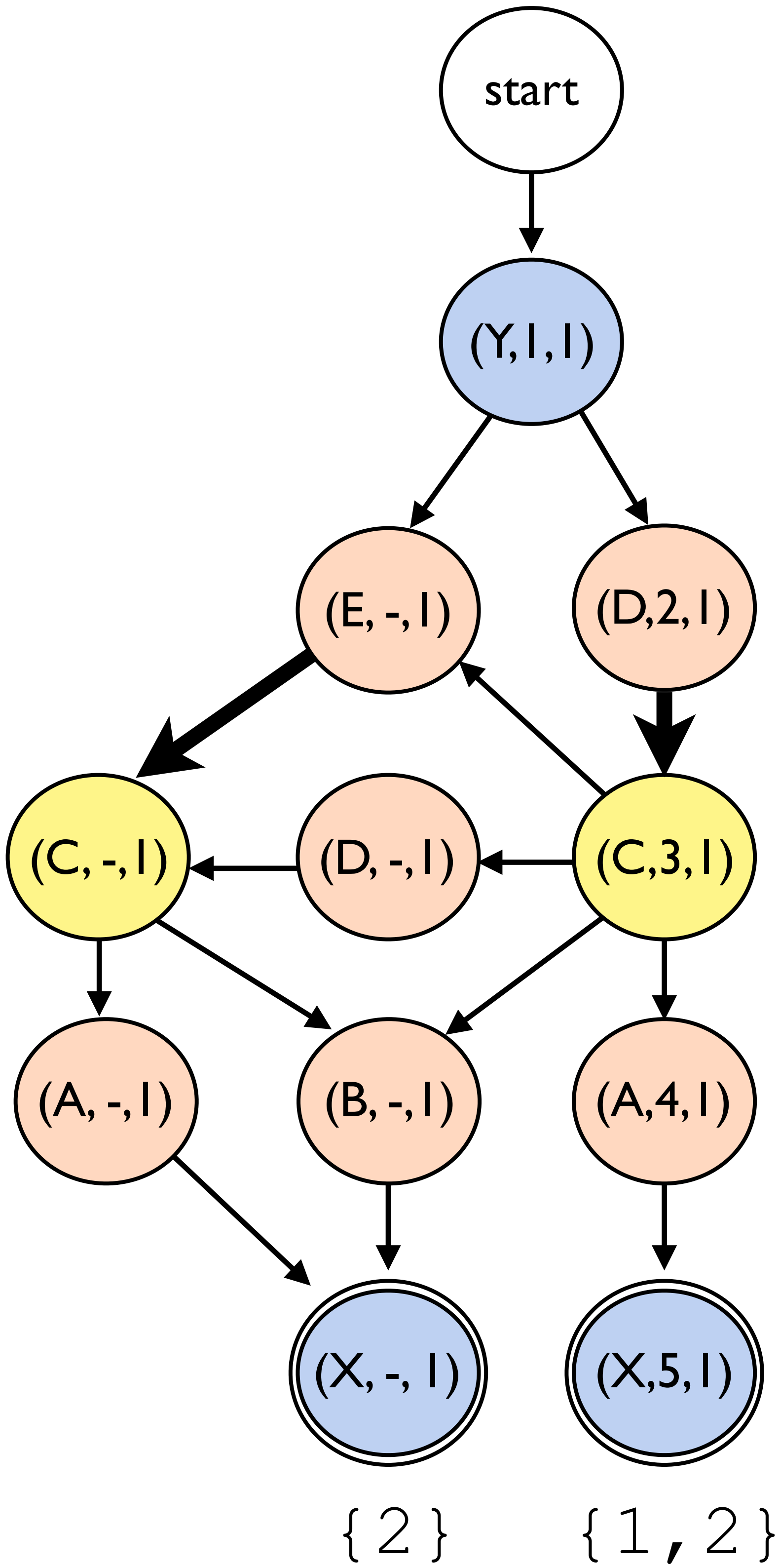
Compilation to BGP:



Router C

```
match peer = D …
  local-pref ← ???
match peer = E …
  local-pref ← ???
```

Compilation to BGP:



Router C

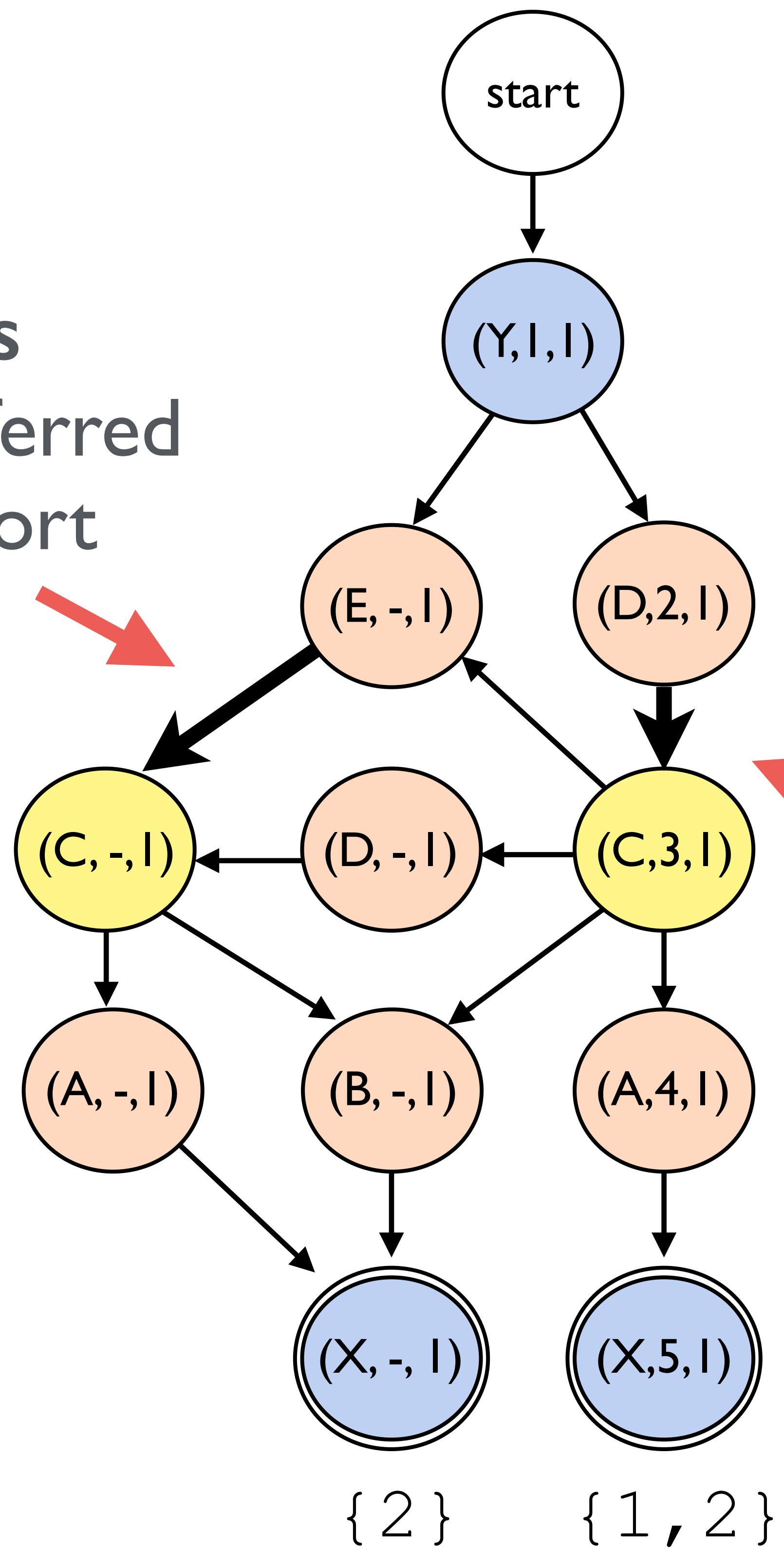
```
match peer = D …
  local-pref ← ???
match peer = E …
  local-pref ← ???
```

Efficient algorithm to assign preferences that forces BGP to find the **best paths** for **all possible failures**

See the paper for details!

Compilation to BGP:

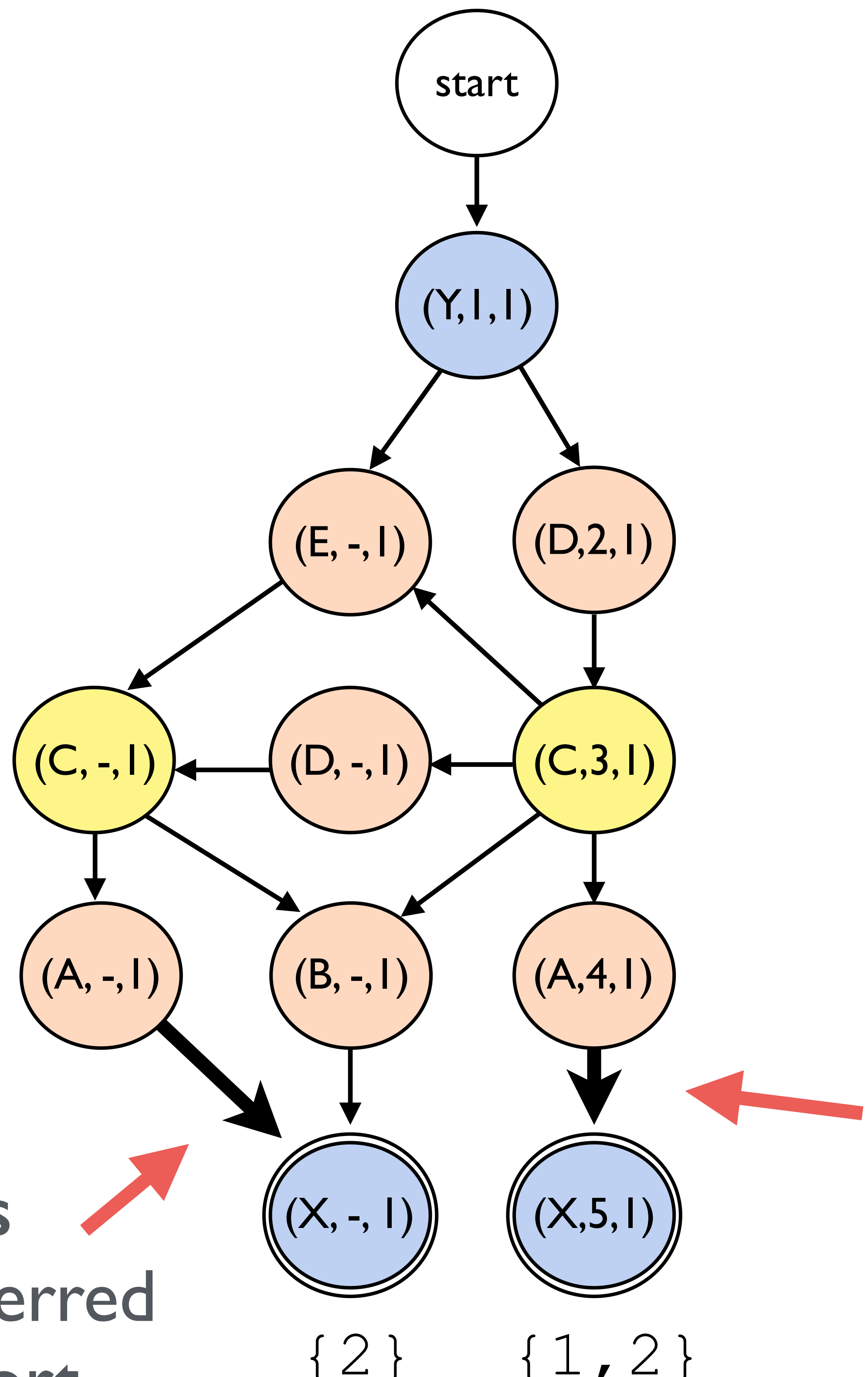
Less preferred import



More preferred import

Implementation: Local preference

Compilation to BGP:

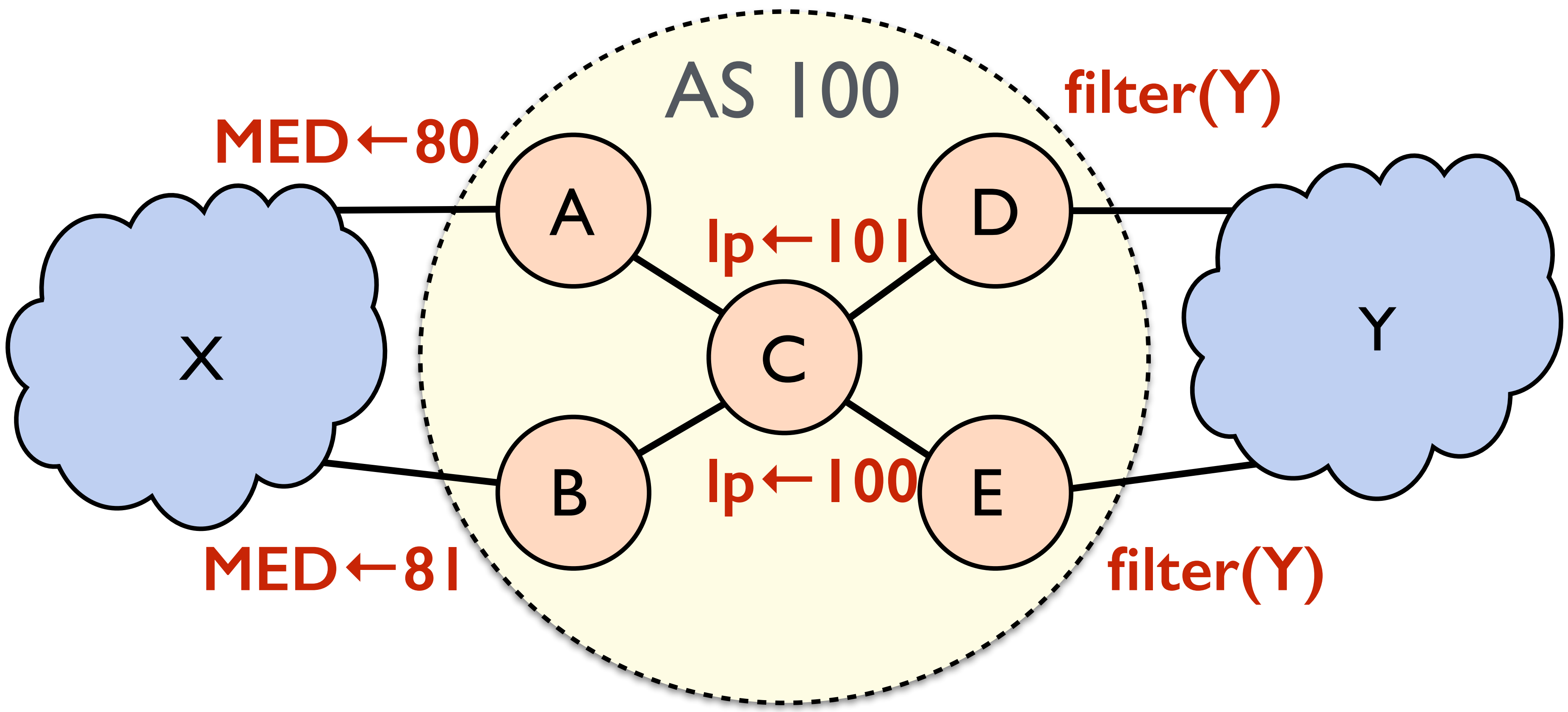


Implementation: MED/Prepending

More preferred import

Less preferred import

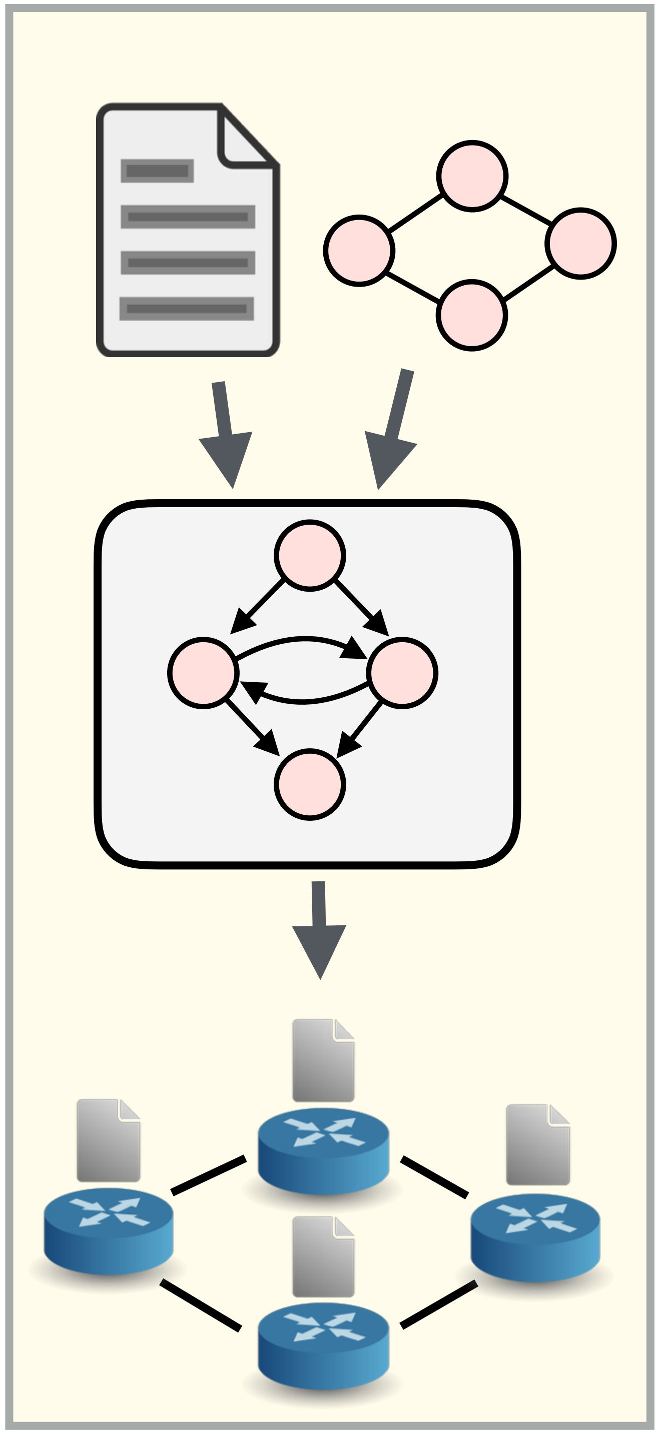
Compilation: A simple Example



`end(Y) & (path(A,C,D) >> any)`

Implementation

- Written in 7000 lines of F#
- Generates Quagga configurations
- A number of other analyses & features



Benchmarks

- Configurations from a large cloud provider
- Policy described in English documents
- Datacenter and Backbone policies

Policy Size

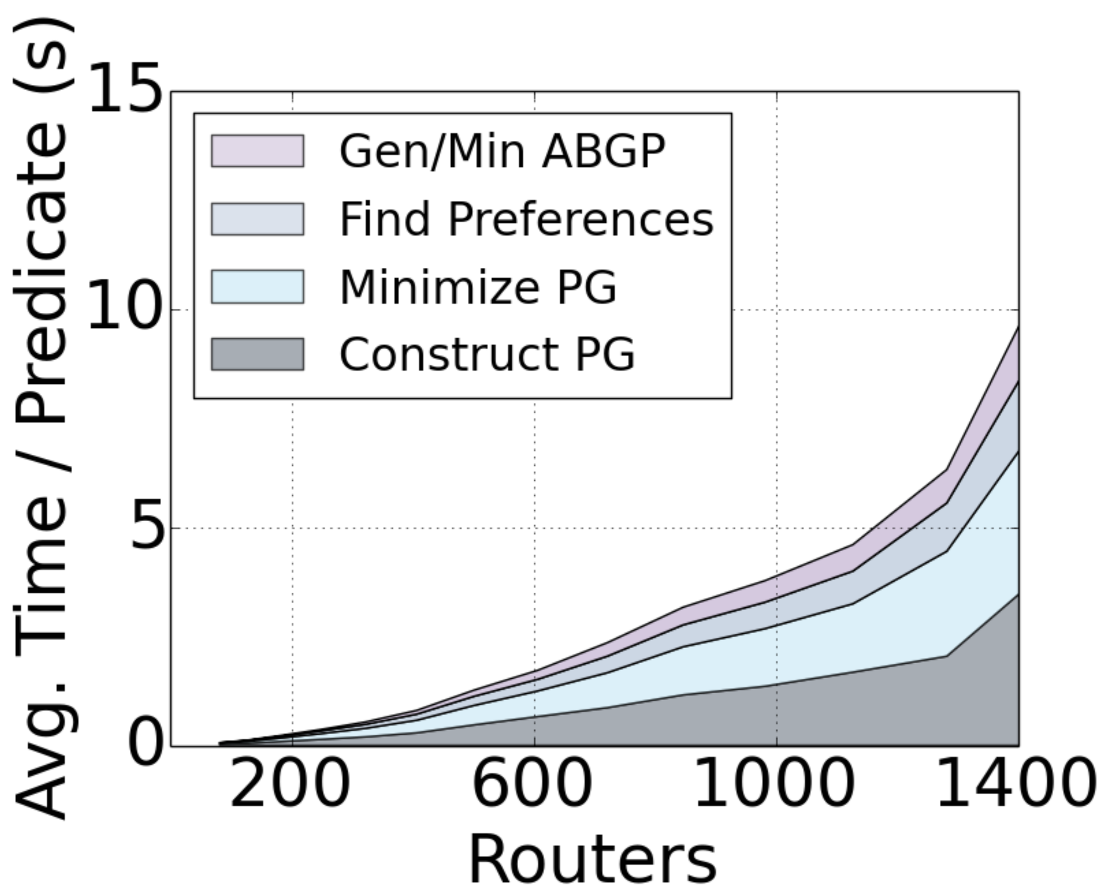
Without prefix/peer definitions

- Datacenter policy: 31 lines of Propane
- Backbone policy: 43 lines of Propane

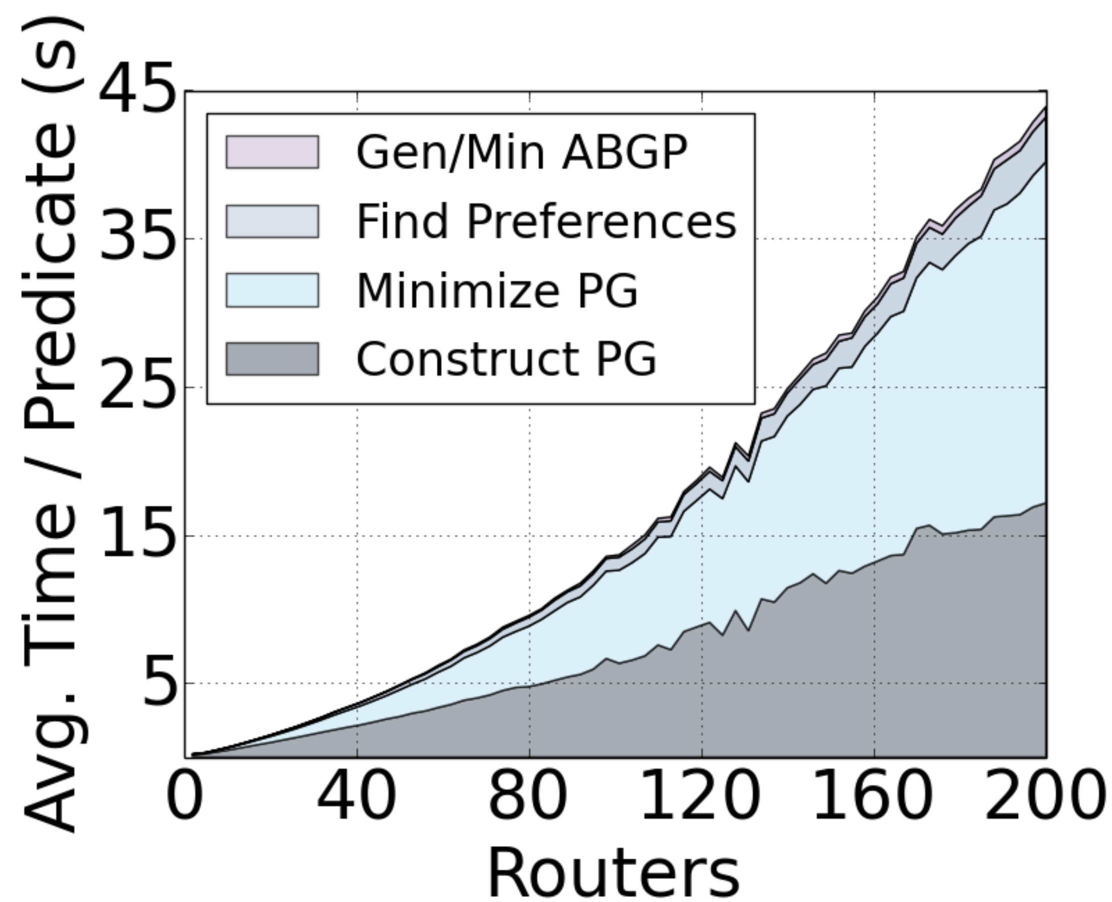
Conventional BGP configurations are 1000s of lines

Compilation Time

- Compile for each prefix ***equivalence class***
- Compile for each equivalence class in ***parallel***
- 8 core, 3.6 GHz Intel Xeon processor



Data center (< 9 min)



Backbone (< 3 min)

Configuration Size

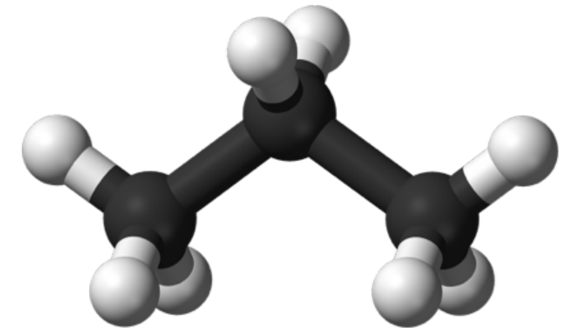
Optimizations

- Avoid using community tags when unambiguous
- Reuse community values across peers
- Merge import/export behaviors across peers

Results

- Optimizations yield **50-100x** decrease in config size
- Configurations **~1000-10000** lines per router

Propane: Summary



High-level language

- Centralized network programmability
- Constraints specify preferred paths and backups in case of failure
- Uniform abstractions for Inter- and Intra-domain routing
- Core policy in 30-50 lines of Propane vs. 1000s

Compiler

- Distributed implementation via BGP
- Static analysis guarantees policy compliance for all failures
- Scales to reasonably sized network topologies