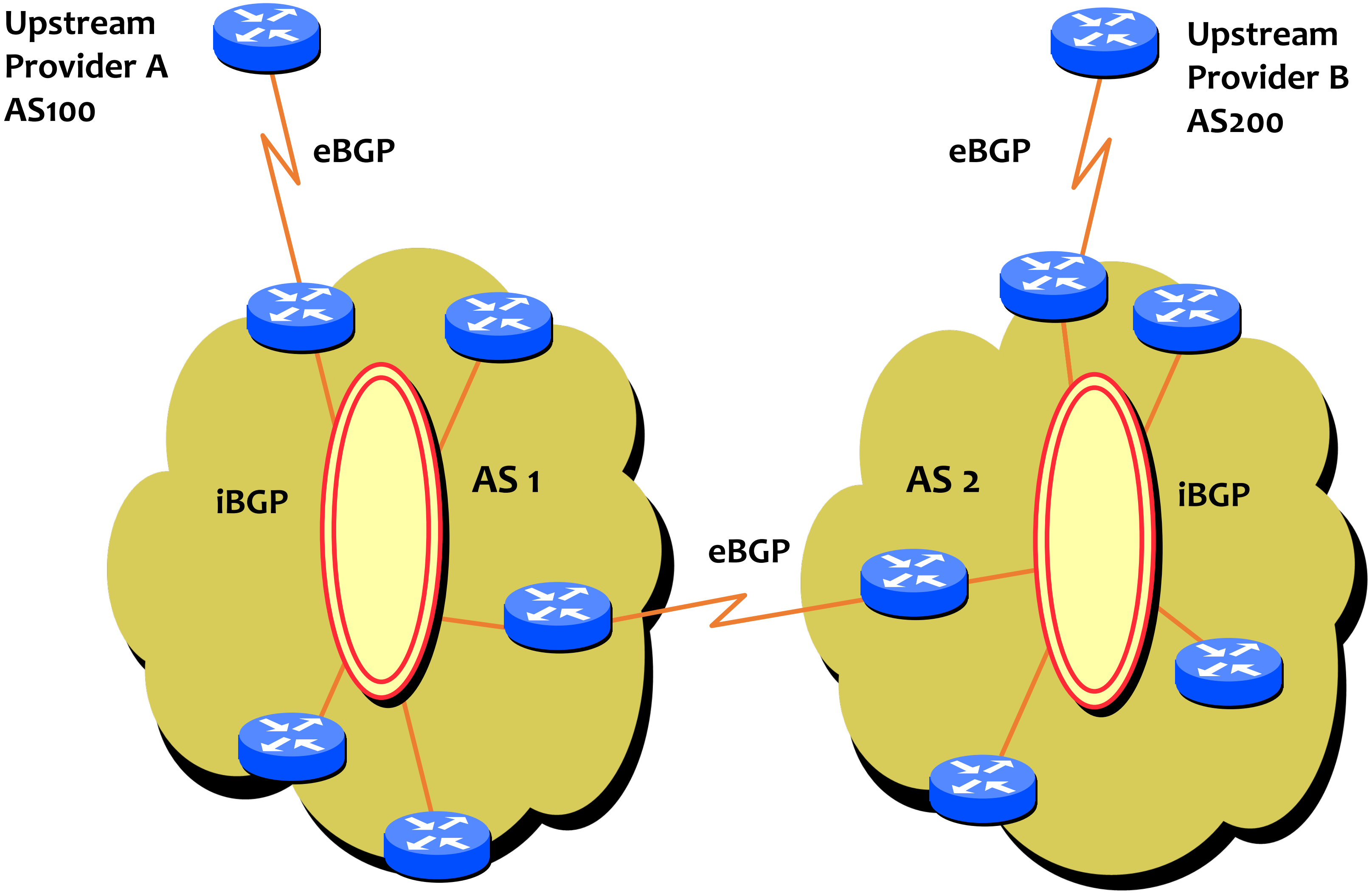
Routing: BGP Advanced topics

Prof. Anja Feldmann, Ph.D.

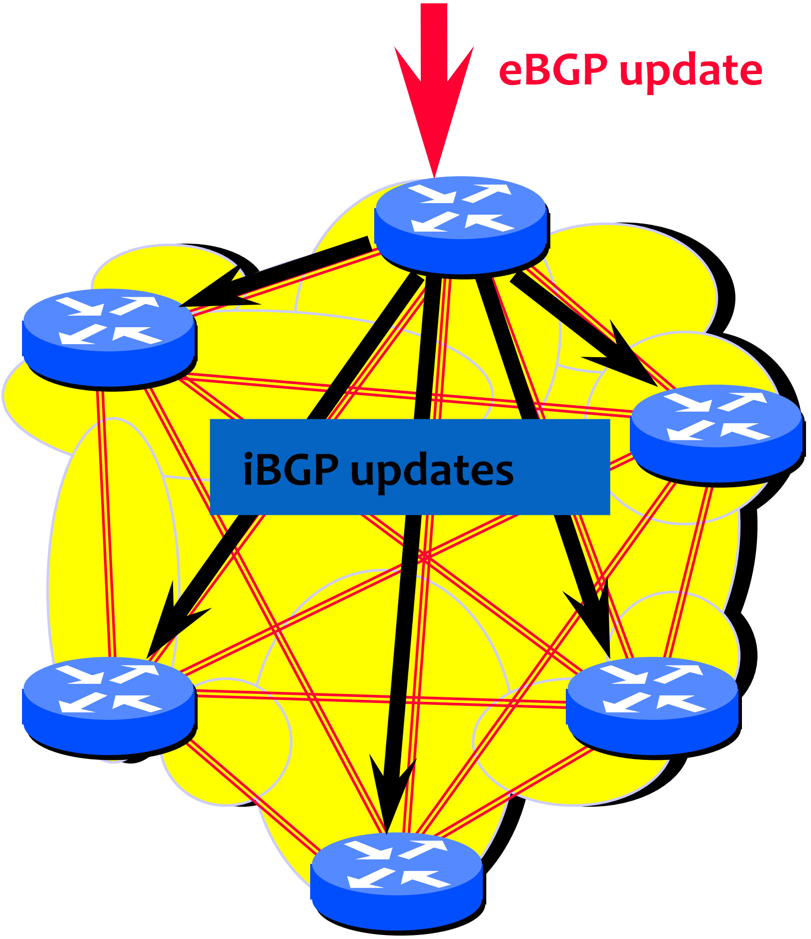
Balakrishnan Chandrasekaran, Ph.D.

Internal BGP (iBGP)

- Same routing protocol as BGP, different application
- iBGP should be used when AS_PATH information must remain intact between multiple eBGP peers
- All iBGP peers must be fully meshed, logically; An iBGP peer will not advertise a route learned by one iBGP peer to another iBGP peer (readvertisement restriction: To prevent looping)


Upstream
Provider A
AS100
eBGP
Upstream
Provider B
AS200
eBGP
iBGP
AS 1
eBGP
AS 2
iBGP

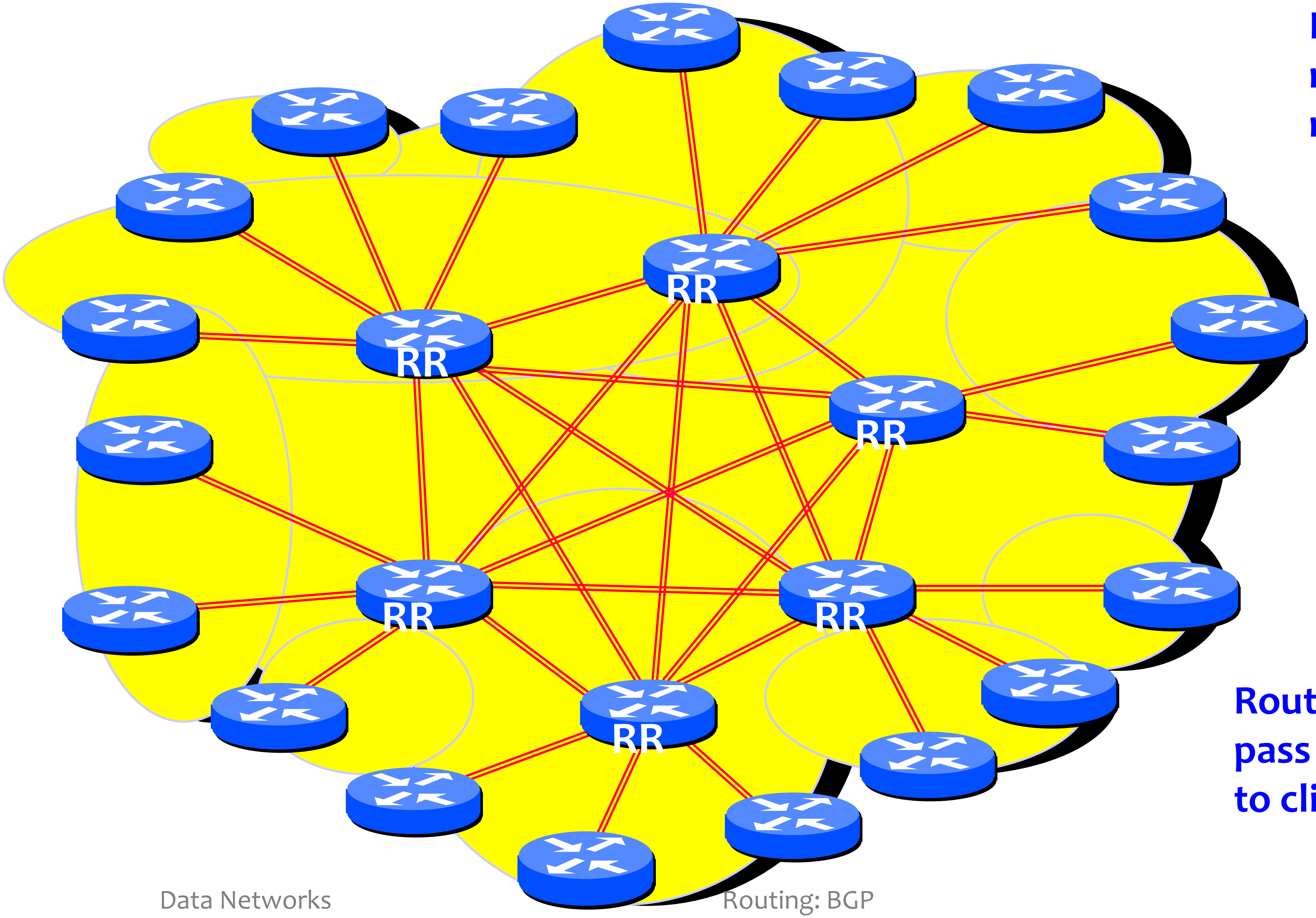
iBGP peers must be fully meshed



iBGP peers do not announce routes received via iBGP

- ***N*** **border routers means *N*(*N*-1)/2 peering sessions – this <u>does not scale</u>**
- **Currently three solutions:**
  - **Break an AS up into smaller Autonomous Systems**
  - **Route Reflectors**
  - **Confederations**

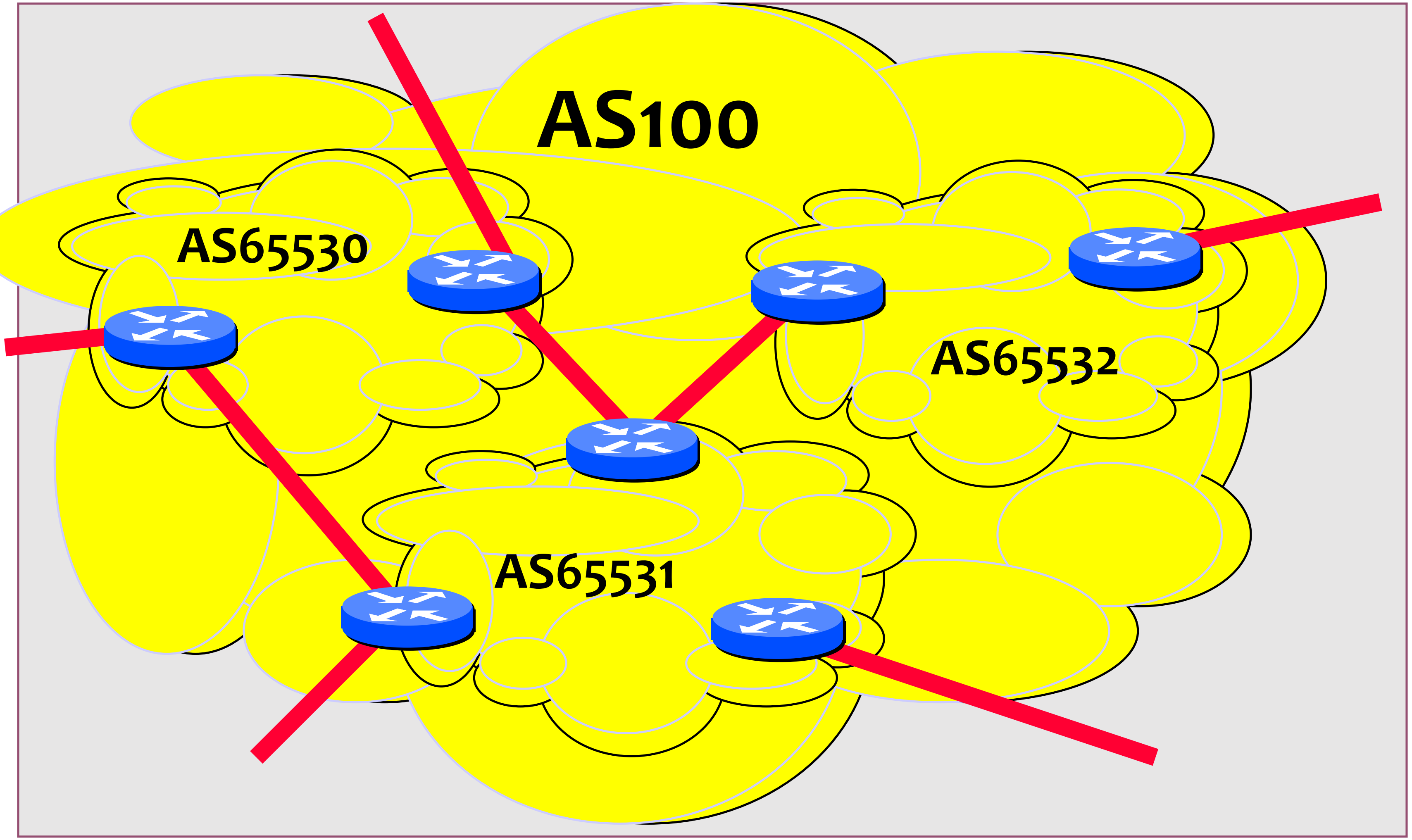
Route reflectors



Route Reflectors must be fully meshed

Route Reflectors pass along updates to client routers

Confederations



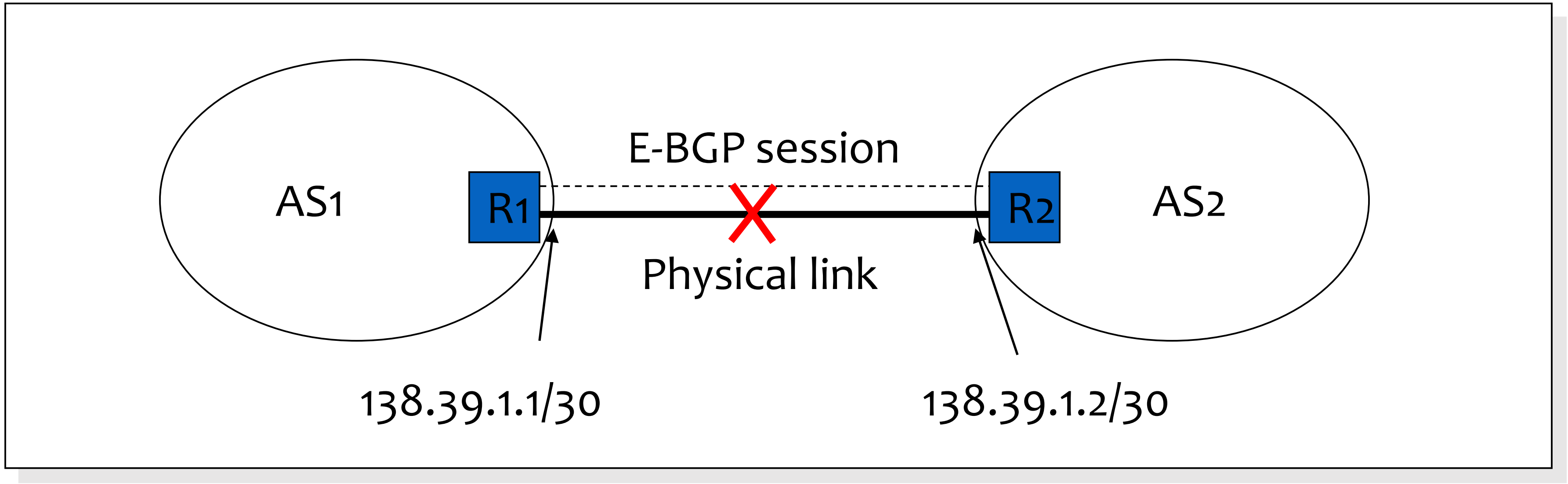
To the global internet, this looks just like AS100

Link failures

- Two types of link failures:
  - Failure on an E-BGP link
  - Failure on an I-BGP Link
- These failures are completely different in BGP
- Why?

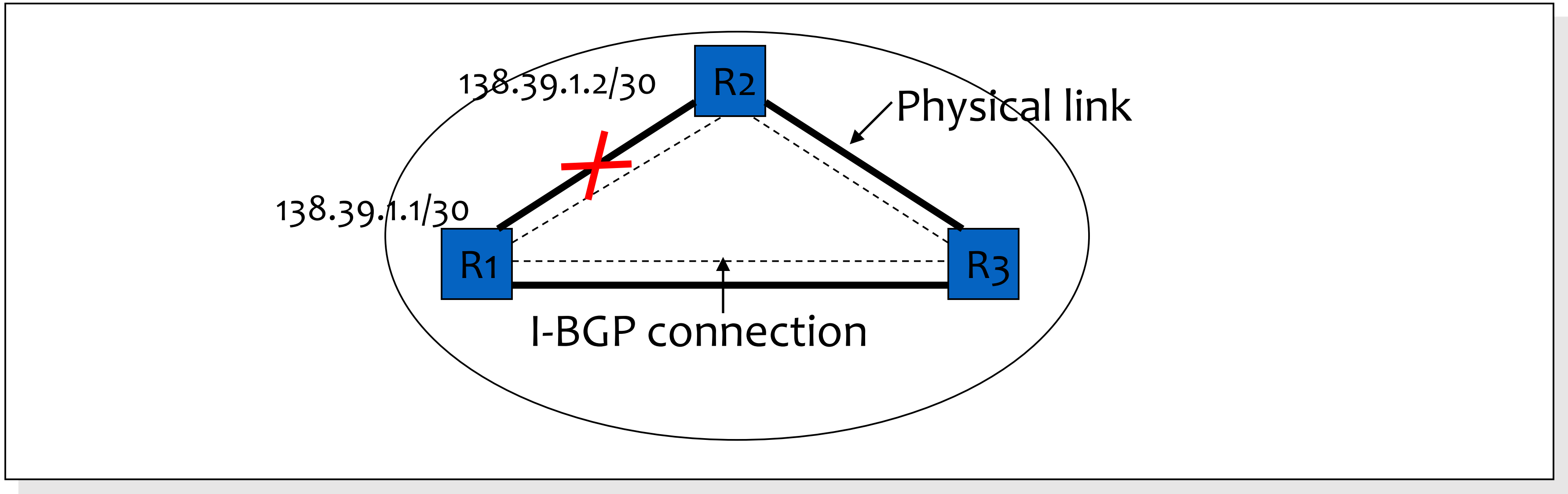
Failure of an E-BGP link

- If the link R1-R2 goes down
  - The TCP connection breaks
  - BGP routes are removed
- This is the desired behavior



Failure on an I-BGP link

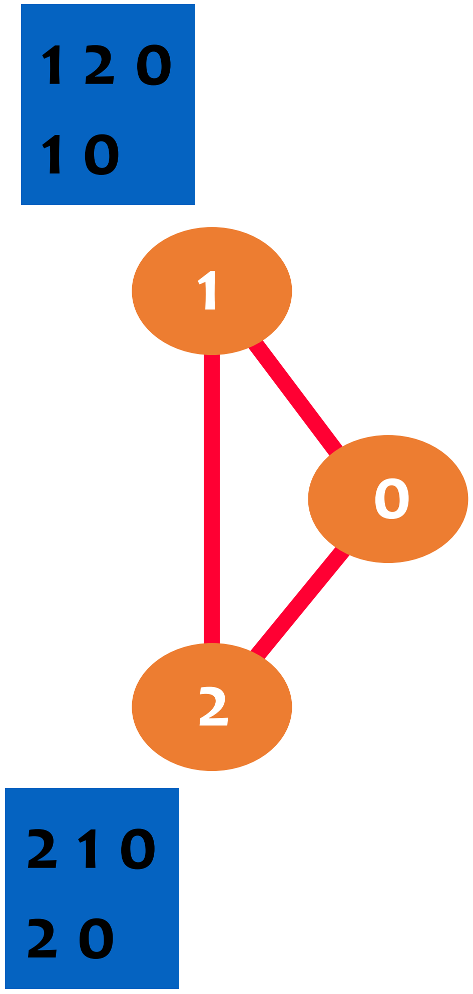
- Link R1-R2 down ➪ R1 and R2 can still exchange traffic
- The indirect path through R3 must be used
- E-BGP and I-BGP use different conventions with respect to TCP endpoints
    - E-BGP: no multihop – I-BGP: multihop OK



BGP is not guaranteed to converge!

- BGP is not guaranteed to converge to a stable routing
- Policy inconsistencies can lead to “livelock” protocol oscillations
- Goal:
  - Design a simple, tractable, and complete model of BGP modeling
  - Example application:
    Sufficient condition to guarantee convergence

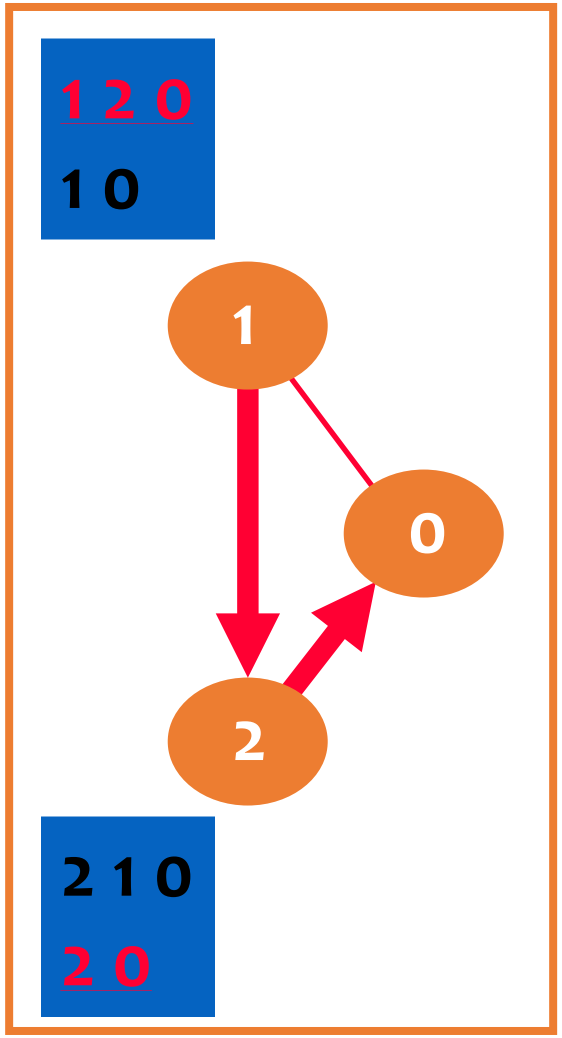
BGP can have multiple solutions



1 2 0
1 0

2 1 0
2 0

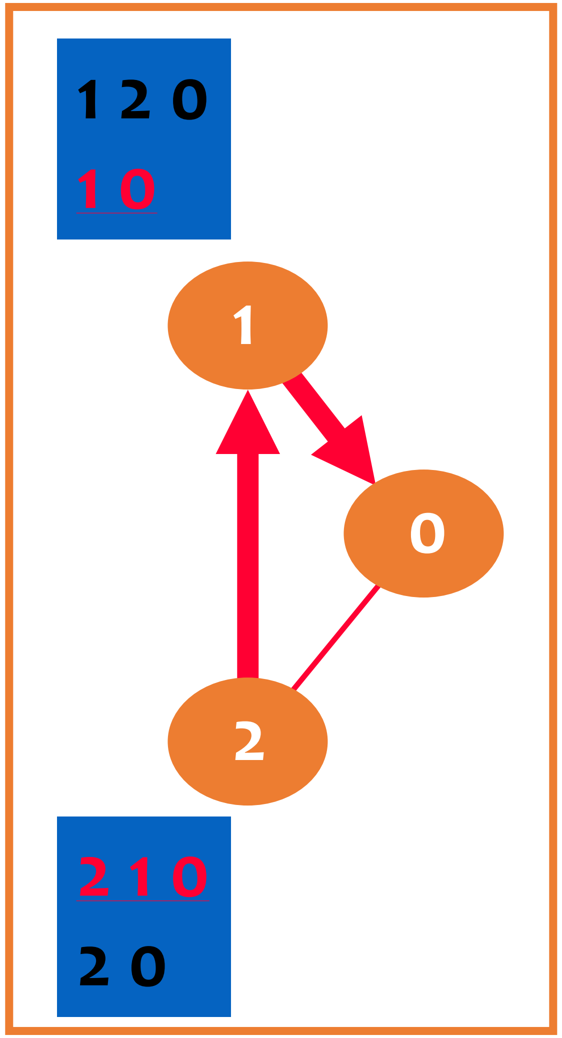
DISAGREE



1 2 0
1 0

2 1 0
2 0

First solution

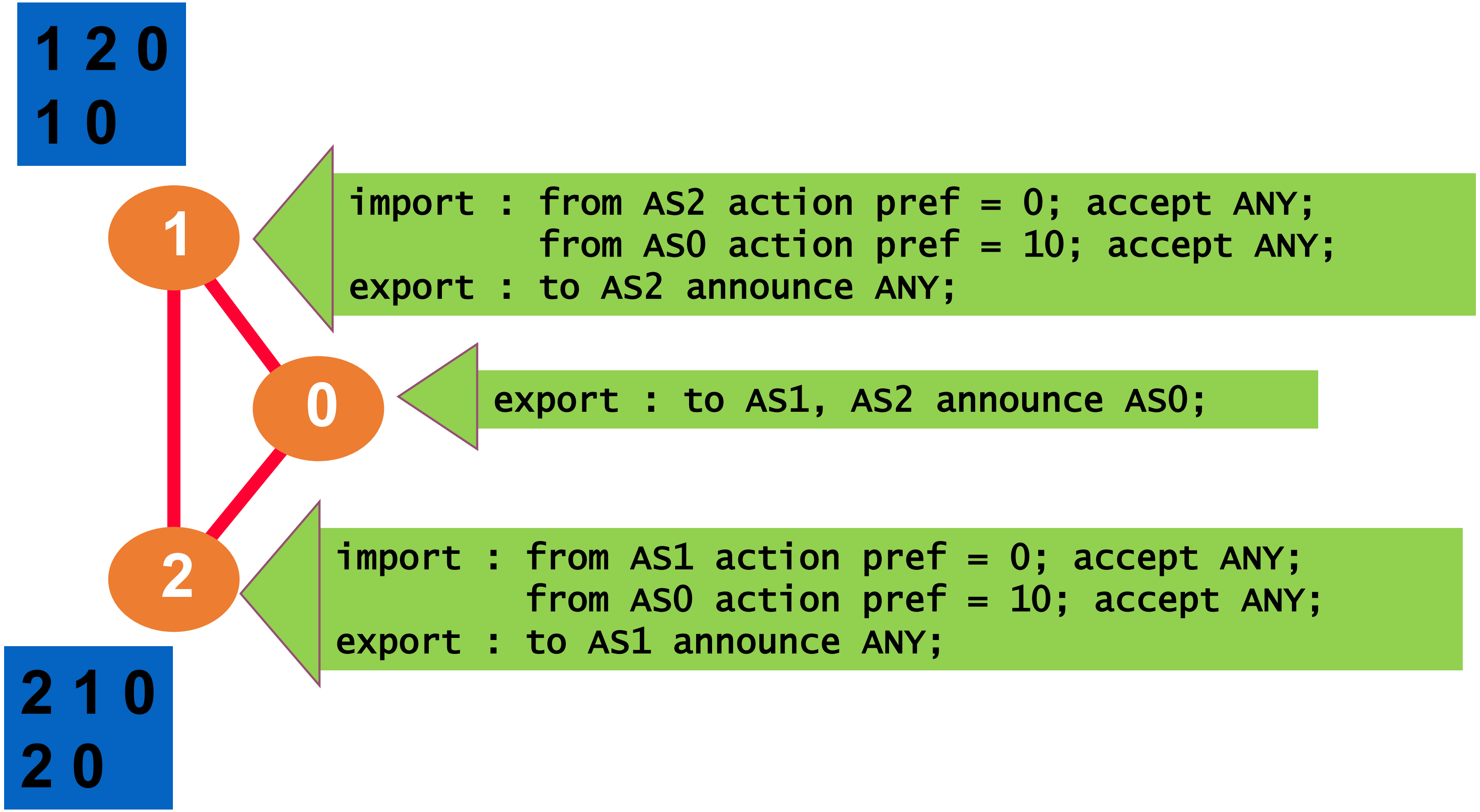


1 2 0
1 0

2 1 0
2 0

Second solution

BGP routing policies for DISAGREE



1 2 0
1 0

```
import : from AS2 action pref = 0; accept ANY;
         from AS0 action pref = 10; accept ANY;
export : to AS2 announce ANY;
```

```
export : to AS1, AS2 announce AS0;
```

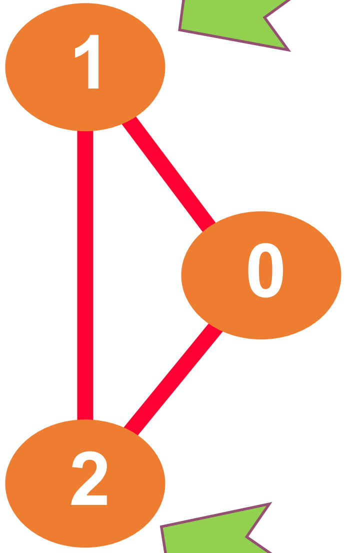
```
import : from AS1 action pref = 0; accept ANY;
         from AS0 action pref = 10; accept ANY;
export : to AS1 announce ANY;
```

2 1 0
2 0

BGP routing policies for DISAGREE (2)

```
1 2 0
1 0
```

```
import : from AS-ANY action pref = 0;
         accept community.contains(1:1);
         from AS-ANY action pref = 10; accept ANY;
export : to AS2 announce ANY;
```



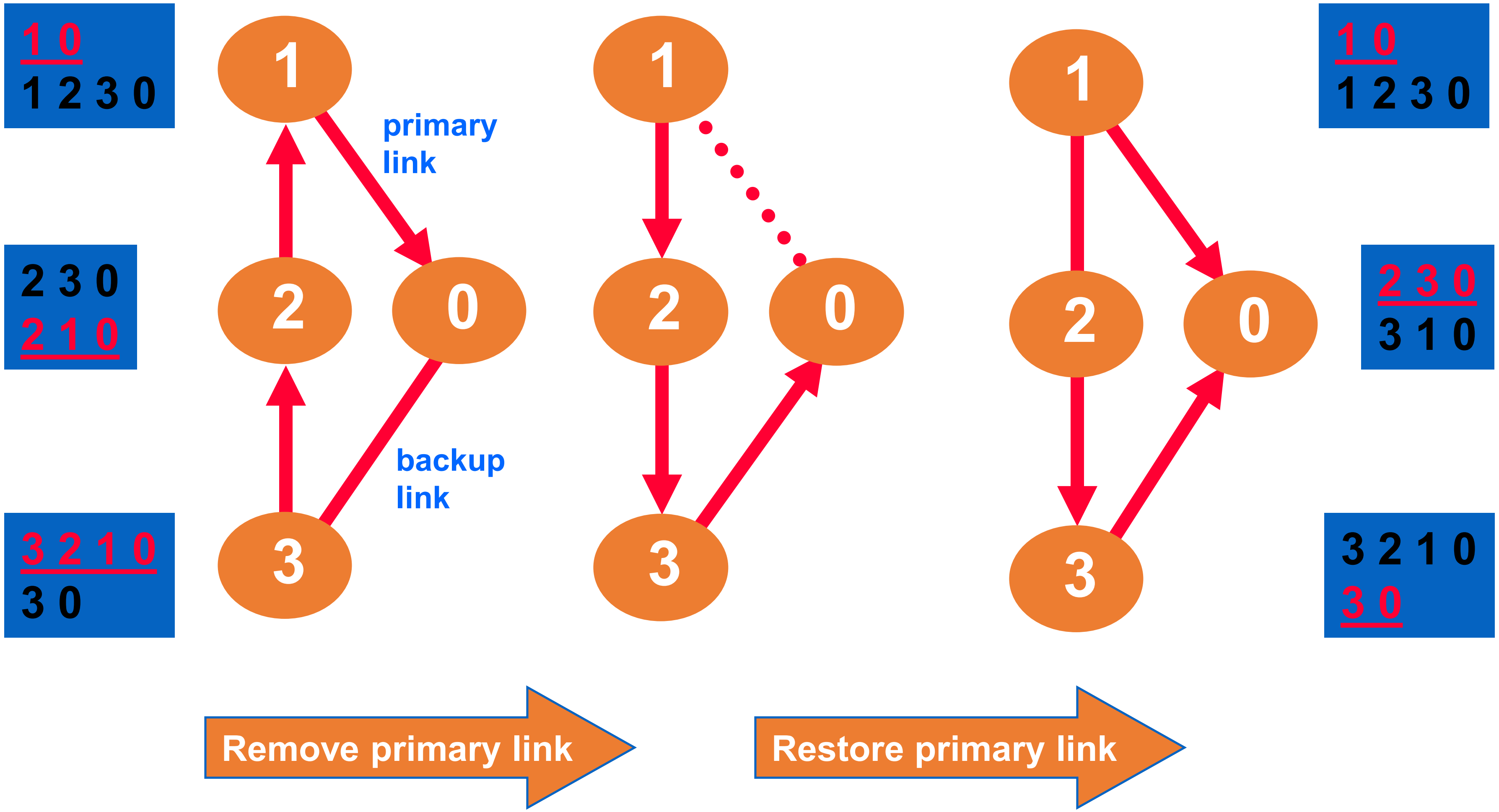
```
export : to AS1
         set community.append(2:1);
         announce AS0;
         to AS2
         set community.append(1:1);
         announce AS0
```

```
import : from AS-ANY action pref = 0;
         accept community.contains(2:1);
         from AS-ANY action pref = 10; accept ANY;
export : to AS1 announce ANY;
```

```
2 1 0
2 0
```

Assume AS1 and AS2 use “neighbor send-community” command ….

Multiple solutions => “Route Triggering”



1 0
1 2 3 0

2 3 0
2 1 0

3 2 1 0
3 0

1 0
1 2 3 0

2 3 0
3 1 0

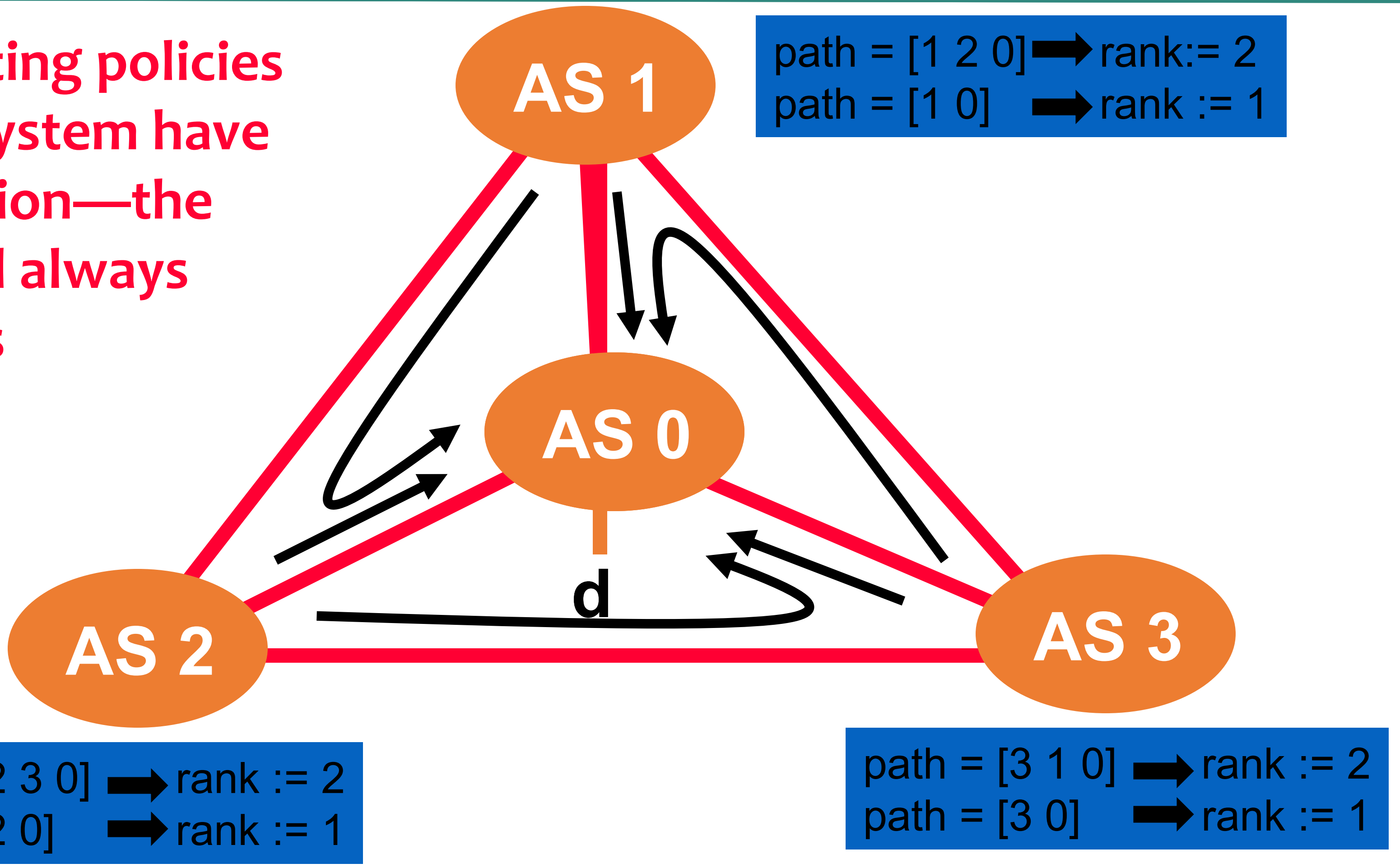
3 2 1 0
3 0

Remove primary link

Restore primary link

BAD GADGET: Always diverges

The routing policies of this system have no solution—the protocol always diverges



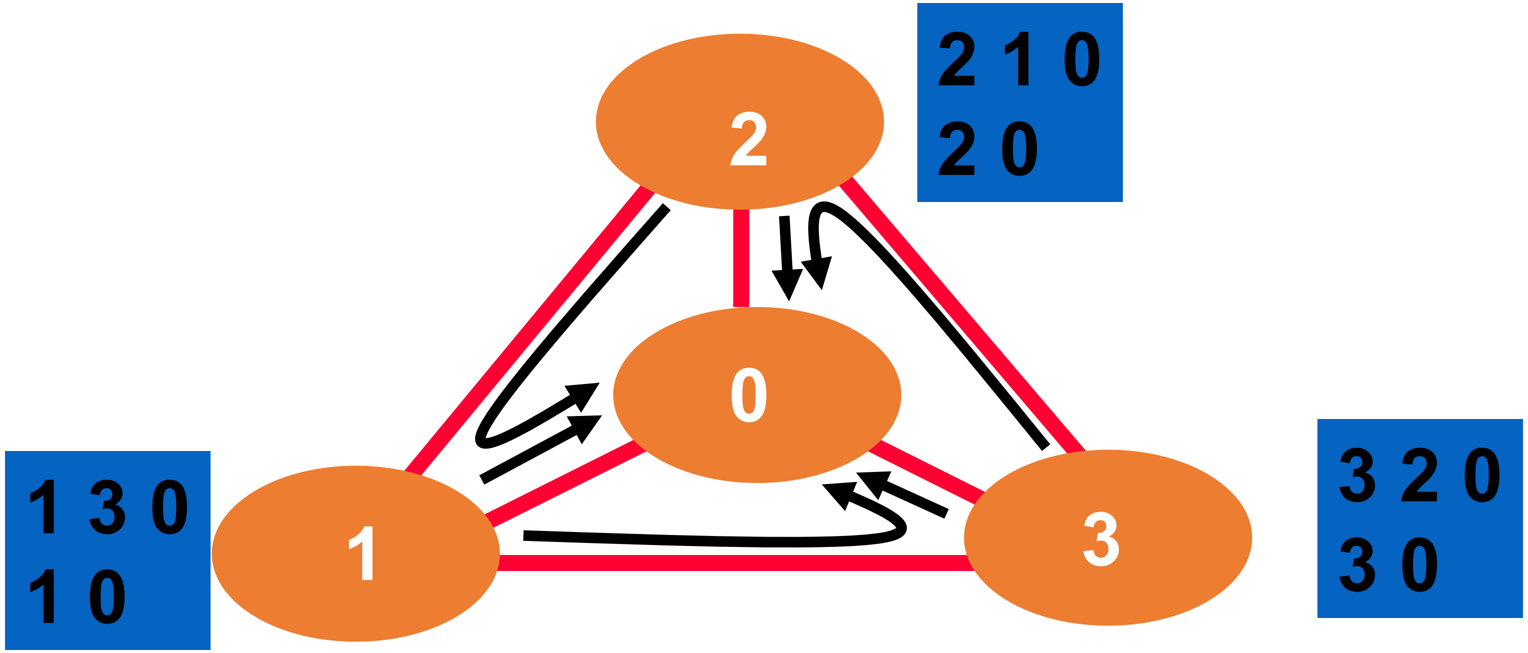
path = [1 2 0] ➡ rank:= 2
path = [1 0] ➡ rank := 1

path = [2 3 0] ➡ rank := 2
path = [2 0] ➡ rank := 1

path = [3 1 0] ➡ rank := 2
path = [3 0] ➡ rank := 1

See "Persistent Route Oscillations in Inter-domain Routing" by K. Varadhan, R. Govindan, and D. Estrin. ISI report, 1996

BAD GADGET



Bad Gadget: No solution

Stage 1:

1: [10]

2: [210]

3: [30]

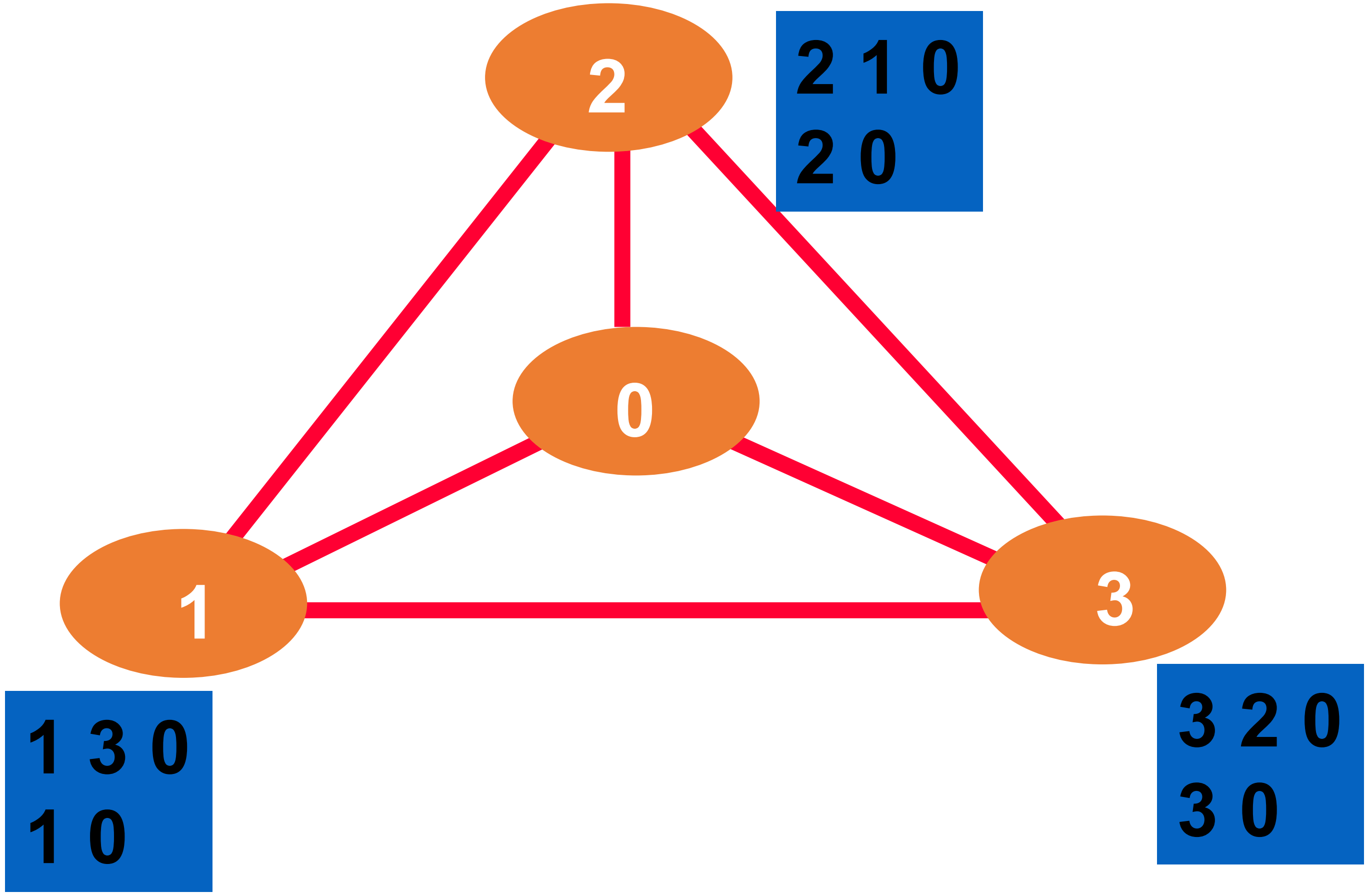
Stage 2:

1:[130]

2:[20]

3:[320]

Back to stage 1



Bad Gadget: No solution

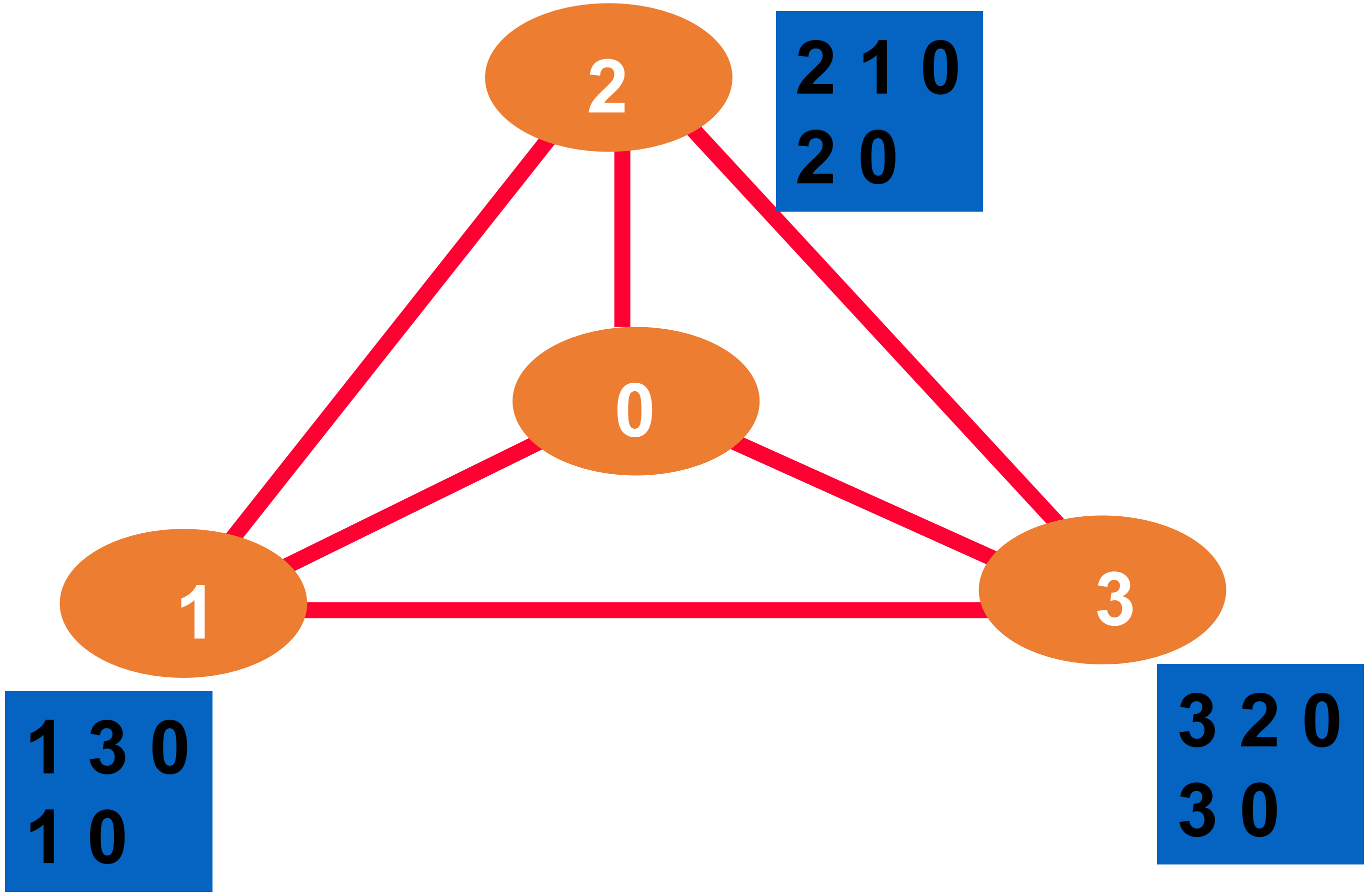
Stage 1:

- 1: [10]
- 2: [20]
- 3: [320]

Stage 2:

- 1:[130]
- 2:[210]
- 3:[30]

Back to stage 1



How to ensure no policy conflicts

Strawman Proposal: Perform Global Policy Check

- Require each AS to publish its policies
- Detect and resolve conflicts

Problems

- ASes typically unwilling to reveal policies
- Checking for convergence is NP-complete
- Failures may still cause oscillations

Think globally, act locally

- Key features of a good solution
  - Safety: Guaranteed convergence
  - Expressiveness: Allow diverse policies for each AS
  - Autonomy: Do not require revelation/coordination
  - Backwards-compatibility: No changes to BGP

- *Local* restrictions on configuration semantics
  - Ranking
  - Filtering

Gao and Rexford Scheme

Gao & Rexford, "Stable Internet Routing without Global Coordination", *IEEE/ACM ToN*, 2001

- Permit only two business arrangements
  - Customer-provider
  - Peering
- Constrain both filtering and ranking based on these arrangements to guarantee safety
- Surprising result: These arrangements correspond to today's most common behavior

Signs of routing instability

- Monitored BGP messages at major exchanges
- Orders of magnitude more updates than expected
  - Bulk: Duplicate withdrawals
    - Stateless implementation of BGP – did not keep track of information passed to peers
    - Impact of few implementations
  - Strong frequency (30/60 sec) components
    - Interaction with other local routing/links etc.

BGP summary

- Neighbors

discovery	configured
maintenance	keep-alives

- Database

granularity	prefix
maintenance	incremental updates & filter
synchronization	full exchange

- Routing table

metric	policies
calculation	route selection

Why different Intra-, Inter-AS routing ?

Policy:

- Inter-AS: admin wants control over how its traffic routed, who routes through its net.
- Intra-AS: single admin, so no policy decisions needed

Scale:

- Hierarchical routing saves table size, reduced update traffic

Performance:

- Intra-AS: can focus on performance
- Inter-AS: policy may dominate over performance

BGP: AS types and policies

- ***Providers***: Offer connectivity to direct customer offer transit to other ISPs
- ***Customers***: Buy connectivity from providers
- ***Peers***: Exchange customers traffic at no cost
- ***Siblings***: others

	Own routes	*Customer's routes*	*Sibling's routes*	*Provider's routes*	*Peer's routes*
Exporting to provider	✓	✓	✓	✗	✗
Exporting to customer	✓	✓	✓	✓	✓
Exporting to peer	✓	✓	✓	✗	✗

Why diff. intra-AS & inter-AS routing?

Policy:

- ***Inter-AS***: Admins want control over how its traffic is routed & who routes its net.
- ***Intra-AS***: Single admin, so no policy decisions needed

Scale:

- Hierarchica

Verdict?

We need both!

Performance:

- ***Intra-AS***: Can focus on performance
- ***Inter-AS***: Policy may dominate over performance