Improving network agility with seamless BGP reconfigurations

Laurent Vanbever
vanbever@cs.princeton.edu

IRTF Open Meeting, IETF87
July, 30 2013

Based on joint work with
Stefano Vissicchio, Luca Cittadini, Cristel Pelsser, Pierre François and Olivier Bonaventure
When you are changing the tires of a moving car

--Vijay Gill
“When you are changing the tires of a moving car, make sure one wheel is on the ground at all time.”

--Vijay Gill
Why does seamless BGP reconfigurations matter?

BGP is critical for ISPs
enforce business relationship, responsible for most of traffic

BGP configuration is often changed
On average, 400+ changes accounted per month in a Tier1

Changing a BGP configuration can impact availability
even if the initial and final configurations are safe
Improving network agility with seamless BGP reconfigurations

1. **BGP reconfiguration**
   A crash course

2. **Finding an ordering**
   Is it easy? Does it exist?

3. **Reconfiguration framework**
   Overcome complexity
Improving network agility with seamless BGP reconfigurations

1. BGP reconfiguration
   A crash course

   Finding an ordering
   Is it easy? Does it exist?

   Reconfiguration framework
   Overcome complexity
BGP is the only inter-domain routing protocol used today
BGP comes in two flavors
external BGP (eBGP) exchanges reachability information between ASes
internal BGP (iBGP) distributes externally learned routes within the AS
Plain iBGP mandates a full-mesh of iBGP sessions

O(n^2) iBGP sessions where \( n \) is the number of routers... quickly becomes totally *unmanageable*

Fair warning: some sessions are missing
With Route Reflection, iBGP routers are hierarchically organized.
Route Reflectors relay route updates between iBGP neighbors
Route Reflectors relay route updates between iBGP neighbors

Lower layers rely on upper layers to learn and propagate routing informations
iBGP and eBGP need to be carefully configured

A BGP configuration is composed of

iBGP

Clients sessions
Route-reflector sessions
Peer sessions

eBGP

External sessions
Routing policies
Each part of a BGP configuration can be changed

Typical reconfiguration scenarios consist in

iBGP
- Clients sessions
- Route-reflector sessions
- Peer sessions

Add sessions
Remove sessions
Change type

eBGP
- External sessions
- Routing policies
Each part of a BGP configuration can be changed

Typical reconfiguration scenarios consist in

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Session Type</th>
<th>Actions</th>
</tr>
</thead>
<tbody>
<tr>
<td>iBGP</td>
<td>Clients sessions</td>
<td>Add sessions</td>
</tr>
<tr>
<td></td>
<td>Route-reflector sessions</td>
<td>Remove sessions</td>
</tr>
<tr>
<td></td>
<td>Peer sessions</td>
<td>Change type</td>
</tr>
<tr>
<td>eBGP</td>
<td>External sessions</td>
<td>Add sessions</td>
</tr>
<tr>
<td></td>
<td>Routing policies</td>
<td>Remove sessions</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Modify policies</td>
</tr>
</tbody>
</table>
Reconfiguring BGP can be disruptive

BGP reconfigurations can create

- signaling anomalies  
  [Griffin, SIGCOMM02]

- dissemination anomalies  
  [Vissicchio, INFOCOM12]

- forwarding anomalies  
  [Griffin, SIGCOMM02]

or any combination of those
Reconfiguring BGP can be disruptive

BGP reconfigurations can create

- signaling anomalies
- dissemination anomalies
- forwarding anomalies

or any combination of those

- routing oscillations
- black holes
- forwarding loops
- traffic shifts
Reconfiguring BGP can be disruptive

BGP reconfigurations can create

- signaling anomalies
- dissemination anomalies
- forwarding anomalies

or any combination of those

How much?
Let’s migrate from a full-mesh to a RR topology
Let’s migrate from a full-mesh to a RR topology, following best practices.

Establish the RR sessions in a bottom-up manner, then remove the full-mesh sessions.

[Herrero10]
Best practices do not work

Tier 1 (50) experiments (cumul. frequency)

60% of the experiments were subject to loops for > 35% of the steps
Best practices do not work

Tier1 (50) experiments (cumul. frequency)

100% of the experiments were subject to traffic shifts for > 40% of the steps
Let’s tune BGP policies
AS1 learns a destination P via 5 egress points
Initially, each egress point is equally preferred.
Depending on its position, each egress receives a percentage of the traffic.
Let’s say that AS2 becomes more preferred
Let’s say that AS2 becomes more preferred.
Let’s say that AS2 becomes more preferred.

60% of the traffic experience a traffic shift.
Let’s say that AS2 becomes more preferred
Let’s say that AS2 becomes more preferred
Let’s say that AS2 becomes more preferred.

60% of the traffic experience a traffic shift.

33% of the traffic experience a traffic shift.
Let’s say that AS2 becomes more preferred

```
preference
```

```
usage
```

```
60
```

```
60
```

```
120
```

```
120
```

```
120
```

```
67%
```

```
33%
```

```
0%
```

```
AS2
```

```
AS1
```

```
AS3
```

```
AS4
```

```
E1
```

```
E2
```

```
E3
```

```
E4
```

```
E5
```

Let’s say that AS2 becomes more preferred

60% of the traffic experience a traffic shift

33% of the traffic experience a traffic shift

16% of the traffic experience a traffic shift
During the migration, 109% of the traffic has been shifted.

60% of the traffic experience a traffic shift.

33% of the traffic experience a traffic shift.

16% of the traffic experience a traffic shift.
Tuning eBGP policies can create huge traffic shifts

Tier1 experiments (cumul. frequency)

50% of the routers experience > 1 TS for each prefix
Improving network agility with seamless BGP reconfigurations

BGP reconfiguration
A crash course

2 Finding an ordering
Is it easy? Does it exist?

Reconfiguration framework
Overcome complexity
To avoid reconfiguration problems, a proper operational ordering must be enforced.

Given an initial & final, anomaly-free, BGP configuration.

Find a sequence of configuration changes such that:

- signaling anomalies
- dissemination anomalies
- forwarding anomalies

never occur, during any migration step.
Find a sequence of configuration changes
Find a sequence of configuration changes

Does it always exist?
Find a sequence of configuration changes

Does it always exist?

Is it easy to compute?
We model iBGP configurations by using extended Stable Path Problem instances
We model iBGP configurations by using extended Stable Path Problem instances.
We model iBGP configurations by using extended Stable Path Problem instances.
We model iBGP configurations by using extended Stable Path Problem instances.
A stable BGP configuration determines the forwarding paths being used.
A seamless migration ordering might not always exist

![Diagram](image-url)

Initial BGP configuration

Final BGP configuration
A seamless migration ordering might not always exist.
A seamless migration ordering might not always exist

Initial BGP configuration

Final BGP configuration
Path preferences

IGP configuration
The initial configuration is anomaly-free
The final configuration is anomaly-free
Let’s add the final session before removing the initial one
Let’s add the final session before removing the initial one.
R1 now learns and selects E2, forcing RR1 to use E2 as well
RR1 uses RR2 to reach E2, and RR2 uses RR1 to reach E1 ...
which creates a forwarding loop
Let’s remove the initial session before adding the final one
Let’s remove the initial session before adding the final one
When we remove the session, R2 and RR2 stop learning E1 and switch to E2.
R1 uses R2 to reach E1, and R2 uses R1 to reach E2
which creates a forwarding loop as well...
Find a sequence of configuration changes

Does it always exist? No.
Find a sequence of configuration changes

Does it always exist? \textbf{No.}

Is it easy to compute?
Finding a seamless migration ordering is computationally hard

Deciding if an ordering free from signaling anomalies exists is NP-hard

reduction in polynomial time from 3-SAT
Finding a seamless migration ordering is computationally hard

Deciding if an ordering free from signaling anomalies exists is NP-hard

reduction in polynomial time from 3-SAT

The same reduction applies for

- dissemination anomalies
- forwarding anomalies
- iBGP or eBGP reconfigurations
Find a sequence of configuration changes

Does it always exist? No.

Is it easy to compute? No.
Find a sequence of configuration changes

Does it always exist? No.

Is it easy to compute? No.

An algorithmic approach is not viable
Improving network agility with seamless BGP reconfigurations

BGP reconfiguration
A crash course

Finding an ordering
Is it easy? Does it exist?

Reconfiguration framework
Overcome complexity
Why is BGP reconfiguration so complex?

Local reconfiguration can have global impact in an unpredictable manner.
Why is BGP reconfiguration so complex?

Local reconfiguration can have global impact in an unpredictable manner.

To avoid that, we could run each configuration in an independent routing plane.

Similar to

- IGP reconfiguration [Vanbever, SIGCOMM11]
- Shadow configuration [Alimi, SIGCOMM08]
The reconfiguration framework leverages Ships-In-The-Night (SITN) migration for BGP

SITNs migrations consists in

1. running multiple BGP routing planes
2. waiting for each plane to converge
3. modifying the plane responsible for forwarding
The reconfiguration framework leverages Ships-In-The-Night (SITN) migration for BGP

SITNs migrations consists in:

1. running multiple BGP routing planes
2. waiting for each plane to converge
3. modifying the plane responsible for forwarding
The reconfiguration framework leverages Ships-In-The-Night (SITN) migration for BGP

SITNs migrations consists in

1. running multiple BGP routing planes
2. waiting for each plane to converge
3. modifying the plane responsible for forwarding

Abstract model of a router
Control-plane

init BGP  final BGP

init forwarding paths
Data-plane
The reconfiguration framework leverages Ships-In-The-Night (SITN) migration for BGP

SITNs migrations consists in

1. running multiple BGP routing planes
2. waiting for each plane to converge
3. modifying the plane responsible for forwarding
The reconfiguration framework leverages Ships-In-The-Night (SITN) migration for BGP.

SITNs migrations consists in:

1. running multiple BGP routing planes
2. waiting for each plane to converge
3. modifying the plane responsible for forwarding

BGP SITN can be deployed on today’s routers using BGP/MPLS VPNs technology.
Let’s reconfigure a network from an iBGP full-mesh ...

GEANT
European research network
36 routers (virtualized)
53 links
Let’s reconfigure a network from an iBGP full-mesh to an iBGP hierarchy

GEANT

European research network

36 routers (virtualized)

53 links

iBGP hierarchy

- Top
- Middle
- Bottom

Planned Backbone Topology by the end of 2010. GEANT is operated by DANTE on behalf of Europe’s NRENs.
Following best practices, traffic was **lost** for **30% of the process**

Average results (30 repetitions) computed on 120+ pings per step from every router to 16 summary prefixes
Following our approach, **lossless** reconfiguration was achieved.

Average results (30 repetitions) computed on 120+ pings per step from every router to 16 summary prefixes.

No loss occurred with our approach.
Improving network agility with seamless BGP reconfigurations

BGP reconfiguration
A crash course

Finding an ordering
Is it easy? Does it exist?

Reconfiguration framework
Overcome complexity
Contributions

1. Study BGP reconfiguration, both practically and theoretically

2. Show that a (seamless) operational ordering
   - might be needed
   - might not exist
   - is computationally hard to find

3. Implement and validate a BGP reconfiguration framework
Improving network agility with seamless BGP reconfigurations

Laurent Vanbever
http://vanbever.eu

IRTF Open Meeting, IETF87
July, 30 2013