About the Correctness of Routing Configurations

Stefano Vissicchio

Université de Strasbourg, 11/12/2013
Why Network Management?

- Network management is necessary
  - Design from scratch is expensive, rare
  - Networks need to be managed and evolved
    - For most of their lifecycle
    - With iterative methodologies [Oppenheimer04, Teare07]
Why Network Management?

- network management is necessary, and crucial
  - large businesses lose 3.6% (on avg.) of annual revenue due to network downtime [Infonetics04]
  - downtime costs several millions of USD/h for critical apps [YankeeGroup04]
  - almost 80% of IT budget is reserved to network management [YankeeGroup04]
How Hard Can it Be?
How Hard Can it Be?

- heterogeneous devices
  - in type and vendors
How Hard Can it Be?

- lots of heterogeneous devices
  - > 1,000 routers in backbones
How Hard Can it Be?

- low-level device configuration
- few automation
How Hard Can it Be?

- distributed protocols
  - that interact between them
How Hard Can it Be?

- strict end to end requirements
  - connectivity
  - availability
  - performance
  - security
  - ...

How Hard Can it Be?

- strict end to end requirements
  - connectivity
  - availability
  - performance
  - security
  - ...

Which Problems?

- Consistency and predictability in a **distributed system**
- Resource allocation boils down to be an **optimization problem**
- **Security** requirements are fundamental
- Configuration management is similar to **software management**
  - configurations == software for networks

...
Which Problems?

- Consistency and predictability in a distributed system
- Resource allocation boils down to be an optimization problem
- Security requirements are fundamental
- Configuration management is similar to software management
  - configurations == software for networks
  - with focus on routing
- ...

Routing Correctness
(BGP, IGP and their problems)
(IP) Routing in the Internet
(IP) Routing in the Internet
(IP) Routing in the Internet

- eBGP selects the ISP path
- basically, a path on a big graph
It Isn’t Simple

- eBGP is **policy-based**
  - not all path information is propagated
  - path preference is locally defined
- Policies are **autonomously** set by ISPs
  - no global coordination (by design) → conflicting policies → routing inconsistencies
Restricting to a Single ISP

- configuration is under control of the same entity
Still Not Simple!

- IGP+iBGP show the same issues of eBGP [Griffin02]
  - possibly conflicting decisions between routers
- Two concurrent reasons
  - partial path visibility
    - scalability $\rightarrow$ information hiding
  - interaction between iBGP and IGP
    - BGP decision is (partially) based on IGP metrics
    - IGP decides the path between two BGP hops
Routing (In)Correctness

- problem: ensure consistent routing
  - convergence to a stable state [Griffin02]
  - correct route propagation *
  - no forwarding loop [Griffin02]

Routing (In)Correctness

- problem: ensure consistent routing
  - convergence to a stable state [Griffin02]
  - correct route propagation *
  - no forwarding loop [Griffin02]
- avoid never-ending synchronization attempts
  - can and do [Berger01] occur

Fixing Routing

- patch the protocol (e.g., [Flavel09])
- configuration guidelines (e.g., [Gao00])
- configuration test and tweak
Fixing Routing

- patch the protocol (e.g., [Flavel09])
  - too much inertia
- configuration guidelines (e.g., [Gao00])
  - not always applicable
- configuration test and tweak
Fixing Routing

- patch the protocol (e.g., [Flavel09])
  - too much inertia
- configuration guidelines (e.g., [Gao00])
  - not always applicable
- configuration test and tweak
Routing Configuration Testing

- routing depends on device configurations
- *problem*: how to check configurations ... ?
  - statically
  - for dynamic routing correctness
  - similarly to software unit testing
- use cases
  - pre-deployment configuration assessment
  - what-if analyses
The Research Perspective

(problem formalization and algorithmic proposal)
Theories for BGP

- **Routing algebras** [Griffin05]
  - semi-rings with non-distributive properties
  - local optima differ from global optimum
- **Game theoretical approaches** [Nisan07]
  - ISPs == players with different strategies
  - BGP steady state== Nash equilibrium
- **Graph-based models** [Griffin99]
  - that take into account protocol peculiarities
Modeling BGP Networks

- graph-based model (SPP [Griffin02])
  - nodes are BGP routers
  - node 0 is the destination
  - links are BGP communication channels (peerings)
Modeling BGP Networks

- a list of **permitted paths** is attached to each node
  - paths that do not appear in a list have been **filtered**
  - the list is ordered according to **local path preference**
Convergence Problems

- in some cases, BGP never converges
  - under specific message timings
Convergence Problems

- in some cases, BGP never converges
  - under specific message timings
Convergence Problems

- in some cases, BGP never converges
  - under specific message timings
Convergence Problems

- in some cases, BGP never converges
  - under specific message timings
Convergence Problems

- in some cases, BGP never converges
  - under specific message timings
Convergence Problems

- in some cases, BGP never converges
  - *under specific* message timings
Convergence Problems

- in some cases, BGP never converges
  - under specific message timings
- even worse, BGP may not be able to converge for any message timing !!!
  - no stable state [Griffin99]
**Structural Inconsistencies**

- cyclic structure of preferences are the root cause of convergence [Griffin02]
  - such a structure is called Dispute Wheel (DW)
  - in a DW, each node prefers its clockwise neighbor
  - No DW $\rightarrow$ convergence

- note that **DWs are static structures**
Did I say “easy”? 

- spoke and rim paths can arbitrarily intersect
Did I say “easy”?

- spoke and rim paths can arbitrarily intersect
Testing BGP Convergence

HAS A STABLE STATE (NP-hard)
Testing BGP Convergence *

HAS A STABLE STATE (NP-hard)  
SAFE

SUF  NO DISPUTE REEL

Testing BGP Convergence **

The Greedy+ Heuristic ***

- Greedy+ can check SPP instances for safety
  - iteratively grows a set of stable nodes
  - until it cannot stabilize any other node
- desirable properties
  - efficient (P-time)
  - correct (but not complete)
  - pinpoints potentially troublesome nodes

From Theory to Practice ***


---

**translation to a generic model**

**efficient and correct heuristic**
From Theory to Practice


flexibility
scalability
iBGP Testing in Subseconds****

Conclusions

- routing testing is trickier than what it looks like
  - poses new problems
    - optimality problems in presence of local preferences
  - asks for new theories
    - semi-rings algebraic structures without distributivity
    - models for asynchronous message exchange and state inconsistencies
  - spurs new algorithms and tools
    - for static checking of dynamic consistency properties
Open Problems

- Theory
  - models for more general routing schemes
  - algorithmic improvements
  - extension to other configuration management tasks
- Practice
  - defeat low-level implementation using abstractions
  - overcome misconfigurations via automation
A Side Note about SDN

- SDN is emerging as a new paradigm
  - logic centralization
  - management simplification
- My point of view
  - not all networks might adopt SDN
    - research on current protocols is still needed
  - not everything should be centralized
    - new distributed protocols could be deployed
    - we need to learn from previous protocol issues
Thank You!

- Questions?