Putting BGP on the Right Path: A Case for Next-Hop Routing

Michael Schapira*
Yale University, CT, and UC Berkeley, CA, USA
michael.schapira@yale.edu

Yaping Zhu and Jennifer Rexford
Princeton University, NJ, USA
yapingz@cs.princeton.edu,
jrex@cs.princeton.edu

ABSTRACT

BGP is plagued by many serious problems, ranging from protocol divergence and software bugs to misconfigurations and attacks. Rather than continuing to add mechanisms to an already complex protocol, or redesigning interdomain routing from scratch, we propose making BGP *simpler*. We argue that the AS-PATH, which lists the sequence of ASes that propagated the route, is the root of many of BGP's problems. We propose a transition from today's *path-based routing* to a solution where ASes select and export routes based *only* on neighboring ASes. We discuss the merits and limitations of *next-hop routing*. We argue that next-hop routing is sufficiently expressive to realize network operator's goals while side-stepping major problems with today's BGP. Specifically, we show that next-hop routing simplifies router implementation and configuration, reduces BGP's attack surface, makes it easier to support multipath routing, and provably achieves faster convergence and incentive compatibility. Our simulations show that next-hop routing significantly reduces the number of update messages and routing changes, and is especially effective at preventing the most serious convergence problems.

Categories and Subject Descriptors

C.2.2 [**Network Protocols**]: Routing protocols

General Terms

Design, Management

Keywords

BGP, AS-PATH

1. INTRODUCTION

Interdomain routing is far too complicated. We ask whether BGP can be made much simpler.

1.1 How BGP Got on the Wrong Path

Routing in the early days of the Internet was substantially simpler than it is today. Shortest-path routing was sufficient for a relatively small network dominated by a single administrative entity. The birth of the commercial Internet, and its metamorphosis into a vast network of privately-owned Autonomous Systems (ASes), raised the need for greater flexibility in selecting routes, for economic, operational, and even political reasons. The Border Gateway Protocol (BGP) was designed to fulfill these needs. BGP announcements include an ordered list of the ASes on the path to the destination (*i.e.*, the AS-PATH attribute). The AS-PATH was originally meant merely to avoid the "count-to-infinity" problems that plagued earlier distance-vector protocols by enabling faster loop detection.

Over time, however, *path-based routing* with BGP has become increasingly more complex, as ASes' routing policies became more and more intricate. Today's routers have a bewildering array of configuration options for selecting and exporting routes. For example, network operators specify policies that prefer shorter paths, prepend their AS number to influence the incoming traffic, apply complex regular expressions to the AS-PATH to balance load over multiple paths or avoid undesirable ASes or countries. And that's just the "good guys." Misbehaving ASes may forge the AS-PATH to hijack prefixes they do not own, or intercept traffic, forcing operators to defensively filter BGP announcements without ever really ensuring that their networks are secure.

The growing complexity of BGP is *not* free. Today's BGP is plagued by configuration errors [1], software bugs [2], slow convergence [3], risks of persistent route oscillations [4, 5], security vulnerabilities [6, 7], economically-driven manipulations [8], and mismatches between the AS-PATH and the path data traffic actually travels. In response to BGP's many problems, the

*Supported by NSF grant 0331548.

Hotnets '10, October 20–21, 2010, Monterey, CA, USA.

research and standards communities have proposed numerous enhancements to BGP, as well as alternative routing architectures. However, these proposals face serious practical obstacles, including poor incremental deployability and limited benefits to early adopters. Instead, we advocate a different approach to "fixing" interdomain routing—reigning in complexity by *constraining* how paths are selected and exported.

1.2 Internet Routing We Can Believe In

Addressing the ills of today's BGP cannot come at the expense of compromising on the problem the routing system is solving—stitching a single global network together out of ASes with diverse policy objectives. We believe the routing system should satisfy the following (possibly conflicting) goals:

- **Loop-freedom** (to ensure traffic delivery);
- **Realization of business policies** (ASes should control which neighbors carry their traffic and direct traffic through them);
- **Fast convergence** (to prevent performance disruptions);
- **Security** (to ensure delivery, avoid dissemination of erroneous information, *etc.*);
- **Incentive compatibility** (ASes should have an incentive to participate honestly in the routing protocol and forward traffic as advertised);
- **Good performance** (ASes should be able to select paths that offer better performance);
- **Traffic engineering** (ASes should be able to balance load and circumvent congestion);
- **Scalability** (the information an AS must store and disseminate should be minimized, and the scope of routing changes limited);
- **Simplicity** (to minimize software complexity for vendors and configuration complexity for operators, and avoid unnecessary outages).

The AS-PATH is arguably more of a hindrance than a help in achieving most of these goals. BGP policies that consider the entire AS-PATH lead to longer convergence times and convoluted, error-prone approaches to traffic engineering. Knowing that other ASes make decisions based on the AS-PATH gives malicious or even rational ASes a reason to lie. And so on. Rather than trying to address these issues, we argue for relegating the AS-PATH to (at most) its original role in loop detection, and considering other ways of achieving our goals for interdomain routing.

1.3 Next Hop(e) for Interdomain Routing

We propose to *constrain* routing policy in ways that are beneficial both *globally* (achieve better convergence, prevent many attacks, remove incentives to lie, and more) and *locally* (simplify both BGP configuration and implementation, reduce BGP's attack surface, make multipath routing much simpler, and more). In particular, we argue that an AS should rank paths *solely* based on the next-hop AS en route to each destination prefix, and apply a simple "consistent filtering" rule [9] when exporting routes. We discuss both the merits and the limitations of *next-hop routing.* We argue that next-hop routing is sufficiently expressive to realize network operators' goals while side-stepping major problems with today's BGP.

In designing and evaluating next-hop routing, we grapple with a wide array of issues, ranging from theoretical results (for convergence and incentive compatibility), simulations (for convergence time and path lengths), protocol design (to extend next-hop routing to the multipath setting), router configuration and operational practices (to enable backwards compatibility), and qualitative arguments (for which design goals should be handled *outside* of BGP).

Whether or not the Internet really moves to next-hop routing, we believe there is value in conducting a thought experiment to understand whether next-hop routing could achieve the goals set for interdomain routing. Perhaps next-hop routing is an alternate way BGP could have evolved, a direction the community should nudge BGP in the future, or even a candidate routing architecture for a future Internet.

2. NEXT-HOP ROUTING RULES!

We define next-hop routing as three simple rules that constrain how routes are selected and exported. We then discuss how next-hop routing simplifies router implementation and configuration.

2.1 Rankings and Export

Rule I: Use next-hop rankings. Configure rankings of routes based *only* on the immediate next-hop AS en route to each destination (*e.g.*, to prefer customer-learned routes over provider-learned routes). Ties in the rankings of next-hops are permitted.

Rule II: Prioritize old routes. To minimize path exploration, when faced with a choice between the "old" (current) route and an equally-good (in terms of next-hop) new one, re-select the old route.

Rule III: Consistently export [9]. If a route P is exportable to a neighboring AS i, then so must be all routes that are more highly ranked than P. Intuitively, Consistent Export prevents undesirable phenomena as in Figure 1, where an AS disconnects a neighbor from a destination by selecting a better route for itself.

2.2 Simple to Realize in Practice

Next-hop routing can be deployed incrementally by

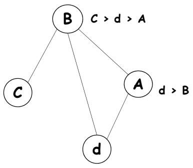


Figure 1: *B* **is willing to export route** *BAd***, but not the more preferred route** *Bd***, to** *C***. Hence, if** *B* **changes its route from** *BAd* **to the** *Bd***, it will disconnect** *C* **from the destination** *d***.**

individual ASes without changing BGP and without any support from neighboring ASes. We envision three main deployment scenarios that offer increasing benefits: (1) configuring today's routers to obey the next-hop routing rules; (2) creating a simpler router configuration interface; and (3) building router software that only supports next-hop routing. In all three deployment scenarios, the AS-PATH is used solely for the purpose of loop detection.

Configuring today's router to obey the next-hop routing rules. Operators can easily switch to next-hop routing by locally configuring their routers to rank routes only based on the next-hop (and disable the AS-PATH length step), to prioritize old routes, and to obey Consistent Export.

Creating a simpler router configuration interface or simpler router software. Simple router configuration interface that *enforces* next-hop routing would require less training for operators and would lead to fewer configuration errors. In addition, the router software would be simpler, and have fewer bugs, since the policy configuration would be easier to parse; this is important, as many bugs in routing software lie in the complex configuration parsing code [2]. Creating router software that only supports next-hop routing will have additional benefits; it would not only have much fewer configuration options, but also fewer execution paths (for applying routing policy) and a faster/shorter decision process.

3. PROS AND CONS

We now discuss the pros and cons of next-hop routing in light of the desiderata listed in the Introduction. We argue that the advantages of next-hop outweigh its disadvantages.

3.1 Merits of Next-Hop Routing

The following four merits are immediate:

Prevents loops. BGP's loop-detection mechanism is still enabled;

Reduces BGP's attack surface. BGP is notoriously vulnerable to AS-PATH-length related attacks. Indeed, major Internet outages have resulted from the announcement of (extremely) long routes. Under next-hop routing, ASes do not consider the AS-PATH when making routing decisions—beyond the first hop, which cannot be forged!—and so an AS no longer benefits from such attacks.

Realizes business policies. ASes sign business contracts with *immediate* neighbors and so next-hop routing is sufficient to realize ASes' business policies;

Simple. See Section 2.2.

We now present three non-trivial advantages of next-hop routing over path-based routing: good convergence, scalability and incentive compatibility.

Good convergence and scalability. In Section 4 we prove and give experimental evidence that next-hop routing converges quickly. We also show that it involves the transmission of significantly fewer BGP update messages than path-based routing, and also necessitates fewer forwarding-table changes. In Section 5, we show that it also has many scalability benefits in the multipath routing context;

Incentive compatibility. If the next-hop routing rules hold then all ASes have an incentive to participate honestly in the routing protocol and forward traffic as advertised [9, 10]. This removes incentives for economically-driven attacks and so has important implications for security.

3.2 Limitations of Next-Hop Routing

We present many good qualities of next-hop routing. However, these come at the cost of limitations on ASes' expressiveness. We now discuss these limitations. We elaborate on the severity of these limitations and on ways to handle them in Section 3.3.

AS-PATH length. Unlike BGP, next-hop routing does not favor routes with shorter AS-PATH lengths.

AS-avoiding policies. Under next-hop routing, an AS can no longer specify BGP routing policies that avoid remote undesirable ASes or countries.

AS-number prepending. Under BGP, ASes sometimes prepend their AS number to make routes through them longer and so less desirable to others. This becomes ineffective with next-hop routing, where the AS-PATH length plays no role in routing decisions.

Expressions on the AS-PATH for traffic engineering. Today, regular expressions on the entire AS-PATH sometimes play a role in traffic engineering to avoid congestion and improve performance.

3.3 Getting Off the AS-PATH

We now discuss how operators can achieve performance, security, and traffic-engineering goals without relying on the AS-PATH. These three topics are themselves open research questions, and serious challenges that exist in today's BGP as well. As such, we do not hope to completely solve these, but rather to argue that next-hop routing makes the situation mostly better.

Performance. We show both analytically and experimentally that next-hop routing leads to much fewer update messages, routing changes and forwarding changes, and is especially effective at preventing the most serious convergence problems. While AS-PATH length is (at best) loosely correlated with end-to-end propagation delay[1], let alone metrics like throughput, clearly a significant increase in path lengths is undesirable. Our simulation results establish that next-hop routing achieves path lengths similar to today's BGP.

Security. We have already discussed the advantages, in terms of security, of next-hop routing over path-based routing (smaller attack surface, incentive compatibility). While next-hop routing renders "AS-avoiding policies" impossible, these come with no guarantees anyway; the AS-PATH lists the sequence of ASes that propagated the BGP announcement, not the path the *data packets* necessarily traverse (and these can differ even for benign reasons). We believe that relying on BGP for data-plane security is misguided. It is precisely when issues like confidentiality and integrity are involved that the matter should not be left to chance, or misplaced trust. Instead, we believe these guarantees should be assured in other (end-to-end) ways, such as encryption and authentication, as suggested in [11].

Traffic engineering. We argue that using existing mechanisms that do not rely on the AS-PATH—replacing regular expressions on the AS-PATH with the use of different next-hop rankings for different (groups of) prefixes, the BGP communities attribute—is as effective, and no more clumsy, than the existing techniques.

We also point out that all three goals (performance, security and traffic engineering) can benefit from leveraging two new mechanisms—multipath routing and end-to-end monitoring— as proposed in, *e.g.*, [12, 13]. Today's BGP provides neither of these mechanisms; next-hop routing lowers the barrier for making multipath routing a reality (see Section 5).

4. FAST CONVERGENCE

BGP gives network operators significant freedom in expressing local routing policies at the risk of *persistent route oscillations* [4]. Even when BGP convergence is guaranteed, this can entail a large number (potentially *exponential* in the size of the network [14]) of forwarding changes and, consequently, also a large number of BGP updates.

We argue that, intuitively, this is due to two main reasons: (1) **small and faraway routing changes can lead an AS to select a new next-hop**, thus leading to a chain reaction of subsequent routing changes; and (2) **inconsistencies between path rankings and route export policies** can lead an AS to disconnect other ASes from a destination when selecting a better route for itself, pushing them to seek alternate routes (see Figure 1).

Intuitively, our next-hop routing rules prevent these scenarios; next-hop rankings guarantee that remote routing changes do not drive an AS to select a new next-hop; Consistent Export guarantees that when bettering its own route an AS *never* disconnects other ASes from a destination. We prove, and give experimental evidence, that next-hop routing converges *quickly* to a "stable" routing configuration.

4.1 Theoretical Results

We prove our results within the model for analyzing BGP dynamics in [5]. We first show that next-hop routing implies the existence of a stable routing state to which BGP can potentially converge.

THEOREM 4.1. *If all ASes obey the next-hop routing rules then a stable state exists in the network.*

Our main theoretical result is proving that under next-hop routing the number of forwarding changes and BGP update messages sent during convergence is at most *polynomial* in the size of the network. We prove our result within the commercial framework of interdomain routing of Gao and Rexford [15], that captures ASes' common business practices.

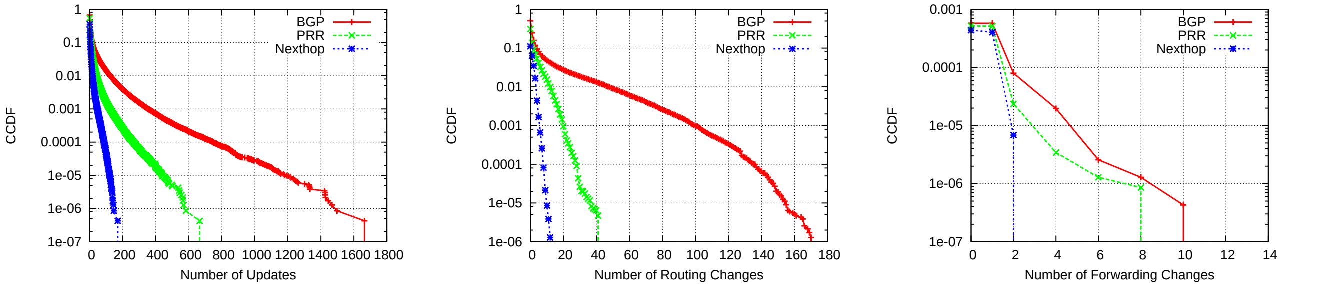
THEOREM 4.2. *If ASes use next-hop routing rules, and the Gao-Rexford conditions hold, then BGP convergence to a stable state requires at most $O(L^2)$ forwarding changes (in total, across all routers), and at most $O(L^3)$ BGP update messages, where L is the number of links in the network. This holds for all initial states of the system and for all timings of router activations/update message arrivals.*

4.2 Simulation Results

Our experiments show that next-hop routing significantly reduces the number of update messages, routing changes, and forwarding changes, under various network events and vantage points. We also evaluate AS-path lengths under next-hop routing.

Topology and metrics. We use the Cyclops [16] AS-level Internet topology on Jan 01, 2010. Each AS is represented by one router, and the links between ASes

[1] A route with short AS-PATH can have many *router*-hops, or be long in terms of *physical* distance.



(a) # updates (b) # routing changes (c) # forwarding changes

Figure 2: Fraction of non-stub ASes experiencing more than x update messages/routing changes/forwarding changes after a link failure

are annotated with business relationship. The topology contains a total of 33,976 ASes, 54,786 provider-customer links and 43,888 peer-peer links. We observe the number of update messages, routing changes, and forwarding changes from all 4,670 non-stub ASes and randomly chosen 5,000 stub ASes as vantage points. We also evaluate AS-path lengths.

Protocols. We evaluate BGP (standard decision process), PRR (Prefer Recent Route) [17], which contains one extra tie-breaking step to prefer current best route over new best routes, and next-hop routing. For all these protocols, we follow the Gao-Rexford conditions [15] for route import and export.

Events. We consider four events: prefix announcement, link failure, link recovery, and prefix withdrawal. We first inject a prefix from a randomly selected multi-homed stub AS, next randomly fail a link between the stub AS and one of its providers, recover the failed link, and then withdraw the prefix. We repeat this experiment for 500 randomly-chosen multi-homed stub ASes.

Results: updates messages, routing changes and forwarding changes. Figure 2(a) plots the distribution of the number of update messages seen at non-stub ASes. Since many ASes on the Internet see little or no effects after any event, we plot the complementary cumulative distribution function (CCDF) to focus on ASes that experience many update messages.

Under BGP, some non-stub ASes receive thousands of update messages (stubs can receive hundreds). PRR greatly reduces this number (middle curve in Figure 2(a)). Next-hop routing leads to even more significant improvement (bottom curve). Next-hop routing also greatly reduces the number of routing and forwarding changes (see Figure 2(b)(c)). Furthermore, next-hop routing performs significantly better across a range of network events and is especially effective at preventing the most serious convergence problems—where an AS experiences thousands of update messages, hundreds of routing changes and tens of forwarding changes. This not only reduces the performance disruptions experienced by the data traffic, but also significantly reduces the overhead for disseminating BGP update messages.

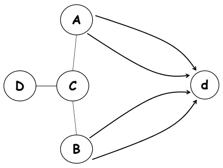


Figure 3: If ASes A and B both announce 2 routes to AS C, and C then announces 4 routes to AS D, and so on. Clearly, this can result in state explosion.

Results: AS-PATH length. We also evaluate in our experiments the AS path lengths during prefix injection, link failure, and link recovery. Our results establish that the vast majority of ASes (97%-99% of ASes, depending on the event) have the same path length under BGP and next-hop routing. In most of the remaining cases (about 1%-2% of ASes), the increase in path length under next-hop routing is 1-2 hops, and only for few ASes (about 0.1%) the increase in path length is significant (a stretch of 2-11). We argue that the small, in general, increase in path length (that is, at best, loosely correlated with performance) is outweighed by the many advantages of next-hop routing in terms of performance. We envision using traffic engineering (change in next-hop rankings) guided, in some cases, by direct monitoring of path performance, to address poor path performance (*e.g.*, poor latency due to extra hops).

5. NEXT-HOP MEETS MULTIPATH

Recently, there has been a surge of interest in multipath routing (see, *e.g.*, [18, 19, 20]). Unfortunately, naive BGP-based multipath routing schemes can be unscalable, due to the need to disseminate and store multiple routes. Consider the example described in Figure 3, that shows that a naive implementation of multipath routing can easily result in state explosion and in excessive transmission of update messages.

We show that next-hop routing is more amenable to multipath than path-based routing. The key observation is that under next-hop routing, a node need not learn a neighboring node's multiple paths, but merely learn enough to avoid loops. If AS C in Figure 3 has a next-hop ranking of routes then, to enable C to detect loops, AS A (and B) can merely send C an (unordered) list of all the ASes its (multiple) routes traverse. BGP allows the aggregation of routes into one such AS-SET [21], that summarizes the AS-PATH attributes of all the individual routes. Thus, BGP route aggregation, used to keep BGP routing tables manageable in other contexts, can also be used to greatly mitigate the cost of multipath next-hop routing.

Hence, next-hop routing lowers the barrier for making multipath routing a reality. Capitalizing on multipath routing can yield the following benefits:

Availability: Multipath routing increases the likelihood that an AS have at least one working path.

Failure recovery: An AS with multiple next-hops can react *immediately* to a failure in one outgoing link by sending traffic along another (and not wait for the protocol to re-converge).

Performance: An AS could have multiple next-hops (with equal local preference) and decide whether and how much traffic to direct through each next-hop (*e.g.*, based on data-plane monitoring). Conventional techniques, such as hashing on fields in the IP header, can ensure that successive packets of the same flow traverse the same path, to prevent out-of-order packet delivery.

Customized route selection: An AS may want to select different routes for different neighboring ASes for economic reasons (*e.g.*, an ISP could offer different services to customers [20]) or operational reasons (*e.g.*, to increase resiliency to failures [19]). Several BGP-based multipath protocols that customize route selection have been proposed [19, 20]. However, these BGP extensions naturally share BGP's plight. We explore next-hop routing in this context and prove that the theoretical results for BGP convergence time, scalability and incentive compatibility extend to this setting.

6. CONCLUSION

Like so many protocols in today's Internet, BGP has grown far too complicated. We believe it is time for the research community to understand the cost of this complexity and identify acceptable ways to *simplify* the Internet infrastructure. Interdomain routing can be made much simpler by reducing our reliance on path-based routing, and solving important security, availability, and performance problems where they belong. We would like to gain a deeper understanding of how much simpler the software and configuration complexity of BGP could be under next-hop routing. We also plan to extend our simulations and to explore techniques for removing the AS-PATH attribute entirely.

7. REFERENCES

[1] R. Mahajan, D. Wetherall, and T. Anderson, "Understanding BGP misconfiguration," in *Proc. ACM SIGCOMM*, pp. 3–16, 2002.

[2] Z. Yin, M. Caesar, and Y. Zhou, "Towards understanding bugs in open source router software," *ACM SIGCOMM CCR*, 2010.

[3] C. Labovitz, A. Ahuja, A. Bose, and F. Jahanian, "Delayed Internet routing convergence," *IEEE/ACM Trans. Networking*, vol. 9, no. 3, pp. 293–306, 2001.

[4] K. Varadhan, R. Govindan, and D. Estrin, "Persistent route oscillations in inter-domain routing," *Computer Networks*, vol. 32, pp. 1–16, March 2000.

[5] T. G. Griffin, F. B. Shepherd, and G. Wilfong, "The stable paths problem and interdomain routing," *Trans. Netw.*, vol. 10, no. 2, pp. 232–243, 2002.

[6] H. Ballani, P. Francis, and X. Zhang, "A study of prefix hijacking and interception in the internet," *ACM SIGCOMM CCR*, vol. 37, no. 4, pp. 265–276, 2007.

[7] S. Goldberg, S. Halevi, A. D. Jaggard, V. Ramachandran, and R. N. Wright, "Rationality and traffic attraction: Incentives for honest path announcements in BGP," in *Proc. ACM SIGCOMM*, pp. 267–278, 2008.

[8] H. Levin, M. Schapira, and A. Zohar, "Interdomain routing and games," in *Proc. ACM Symposium on Theory of Computing*, pp. 57–66, May 2008.

[9] J. Feigenbaum, V. Ramachandran, and M. Schapira, "Incentive-compatible interdomain routing," in *Proc. ACM Electronic Commerce*, pp. 130–139, 2006.

[10] J. Feigenbaum, M. Schapira, and S. Shenker, "Distributed algorithmic mechanism design," in *Algorithmic Game Theory* (N. Nisan, T. Roughgarden, E. Tardos, and V. Vazirani, eds.), ch. 14, Cambridge University Press, 2007.

[11] D. Wendlandt, I. Avramopoulos, D. Andersen, and J. Rexford, "Don't Secure Routing Protocols, Secure Data Delivery," in *Proc. Hotnets V*, 2006.

[12] D. G. Andersen, H. Balakrishnan, M. F. Kaashoek, and R. Rao, "Improving Web availability for clients with MONET," in *Proc. USENIX NSDI*, 2005.

[13] M. Motiwala, M. Elmore, N. Feamster, and S. Vempala, "Path Splicing," in *Proc. ACM SIGCOMM*, Aug. 2008.

[14] U. Syed and J. Rexford, "Some results on BGP convergence," Jan. 2010. working paper, `http://www.cs.princeton.edu/~usyed/bgp_convergence_time.pdf`.

[15] L. Gao and J. Rexford, "Stable Internet routing without global coordination," *IEEE/ACM Trans. on Networking*, vol. 9, no. 6, pp. 681–692, 2001.

[16] B. Zhang, R. Liu, D. Massey, and L. Zhang, "Collecting the internet AS-level topology," *ACM SIGCOMM CCR, special issue on Internet Vital Statistics*, 2005.

[17] "BGP best path selection algorithm." `http://www.cisco.com/en/US/tech/tk365/technologies_tech_note09186a0080094431.shtml`.

[18] P. B. Godfrey, I. Ganichev, S. Shenker, and I. Stoica, "Pathlet routing," in *Proc. ACM SIGCOMM*, 2009.

[19] N. Kushman, S. Kandula, D. Katabi, and B. Maggs, "R-BGP: Staying connected in a connected world," in *Proc. USENIX NSDI*, 2007.

[20] Y. Wang, M. Schapira, and J. Rexford, "Neighbor-Specific BGP: More flexible routing policies while improving global stability," in *Proc. ACM SIGMETRICS*, 2009.

[21] "Understanding route aggregation in BGP." `http://www.cisco.com/en/US/tech/tk365/technologies_tech_note09186a0080094826.shtml`.