OPCA: Fault-Tolerant Routing and Load-Balancing

Sharad Agarwal†
Computer Science Division
University of California, Berkeley
sagarwal@cs.berkeley.edu

Chen-Nee Chuah†
Dept. of Electrical & Computer Engineering
University of California, Davis
chuah@ece.ucdavis.edu

Randy H. Katz
Computer Science Division
University of California, Berkeley
randy@cs.berkeley.edu

Abstract—**The number of stub ASes that participate in the inter-domain BGP peering sessions has grown. Most of these ASes are multi-homed to multiple upstream providers. We believe that this behavior exhibits a widespread intent to achieve fault tolerant and load balanced connectivity to the Internet. However, BGP today offers route fail-over times as long as 15 minutes, and very limited control over incoming traffic across multiple wide area paths. We propose an policy control architecture, OPCA, that runs as an overlay network on top of BGP. OPCA allows an AS to make route change requests at other, remote ASes to achieve faster route fail-over and provide capabilities to control traffic entering the local AS. The proposed architecture and protocol will co-exist and interact with the existing routing infrastructure.**

I. Introduction

The Border Gateway Protocol (BGP) [1] is the de-facto inter-domain routing protocol between Autonomous Systems (ASes) that achieves global connectivity while shielding intra-domain routing details and fluctuations from the external view. Recent studies of BGP [2], [3] have indicated a significant growth in BGP routing tables, an increase in route flapping and unnecessarily specific route announcements. The large growth in the number of ASes that participate in BGP peering sessions has been fueled by stub ASes. Our analysis of the BGP dumps from Routeviews [4] reveals that 60% of these stub ASes are *multi-homed* to two or more providers, i.e., they announce BGP routes via multiple upstream ASes. Multi-homing is intended as a solution to achieve two goals: fault tolerance and load balancing on inter-domain routes.

As an illustration, Figure 1 compares two scenarios where a stub AS is (a) single-homed and (b) multi-homed to three providers. The stub AS, W, in Figure 1(b) can choose to have its traffic go primarily through ISP X. If the link to ISP X fails, or when there are failures along the path through ISP X, W can failover to ISP Y or Z. If it were singly homed, as in case (a), it could only protect itself against upstream link failures by purchasing multiple redundant links to ISP X. In addition, W can load-balance its outgoing traffic by selecting the best route to the destinations via one of the three providers. Companies such as Routescience [5] provide devices that automate outgoing traffic balancing by selecting specific BGP announcements heard from different providers.

Achieving connectivity by subscribing to multiple providers is likely to be expensive, but Mortimer's study [6] suggests that reliability is a deciding factor. However, the effectiveness of multi-homing is limited by the slow convergence behavior of BGP. Inter-domain routes can take as long as 15 minutes or more [7] to fail-over. For companies that rely on Internet connectivity to conduct online transactions, such a long outage can have a severe financial impact. Furthermore, BGP allows an AS very little control over how the incoming traffic enters its network.

Instead of overloading BGP with protocol extensions, we propose to address these problems by developing an Overlay Policy Control Architecture (OPCA) running on top of BGP to facilitate policy exchanges. Our architecture relies on knowing AS relationships and the AS level hierarchy. The two main goals of OPCA are:

- to support fast, fine grained management of incoming traffic across multiple incoming paths, and
- to reduce the fail-over time of inter-domain paths.

Together, these goals serve to improve routing and traffic in the current inter-domain routing structure, and allow it to scale better to the growing number of multi-homed ASes.

OPCA consists of a set of intelligent Policy Agents (PAs) that can be incrementally deployed in all the participating AS domains. The PAs are responsible for processing external policy announcements or route-change requests while adhering to local AS policies, and enforcing necessary changes to local BGP routers. These PAs communicate with one another via a new Overlay Policy Protocol (OPP). Such an overlay architecture allows ASes to negotiate the selection of inter-domain paths for incoming traffic with remote ASes, leading to more predictable load-balancing performance. In addition, an AS can request routing changes to other ASes to expedite fail-over. Our architecture does not require any modifications to the BGP protocol or to existing BGP routers.

We will review related work in the next section. Section III describes the design of our architecture and discusses how it improves route fail-over and achieves incoming traffic balancing. We outline some design decisions related to scaling and deployment in Section IV, and conclude the paper in Section V.

II. Related Work

In [8], Huston suggests multiple ways to address the problems of load balancing and route fail-over. There are two main classes of approach: (a) extending the current BGP protocol/implementations or (b) use alternate routing through overlay networks or to replace BGP.

Various BGP-based solutions have proposed to limit the advertisement scope of route announcements [9], [10], [11]. For example, BGP can be modified to allow bundling of routes or to specify aggregation scopes. These proposals may limit the ill-effects of multi-homing but do not solve the issues of fast fail-over and inbound load balancing that we are concerned with. In addition, they require a new version of BGP to be deployed or many BGP routers to be reconfigured. This is difficult to ac-

†Sharad Agarwal and Chen-Nee Chuah are also affiliated with the IP & Interworking Division of the Sprint Advanced Technology Laboratories

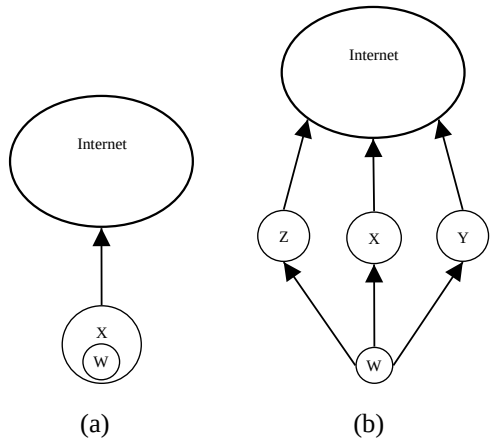

Fig. 1. (a) Company W with ISP X. (b) Company W with ISP's X, Y and Z

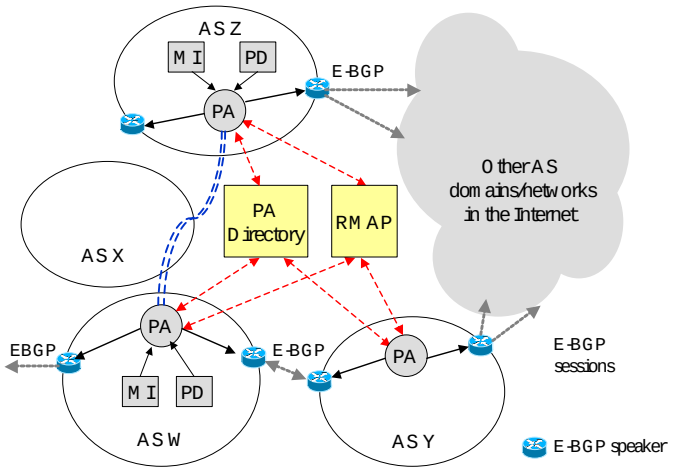

Fig. 2. Overlay Policy Control Architecture

complish given the widespread use of BGP. In the early days of BGP, Estrin [12] proposed a centralized routing arbiter that collects all routing entries, centrally computes the "best" routes, and re-distributes the final routing entries. Such an architecture is not deployed in the Internet today due to complexity and scalability issues.

Alternative routing architectures have been proposed, such as RON [13], Nimrod [14], and BANANAS [15]. RON is an overlay network that uses *active probing* and *global link state* in a fully *meshed* network to customize routing between overlay nodes. RON is designed for applications with a small number of participating nodes and cannot scale to the number of ASes that exist today. Nimrod [14] was proposed in 1996 as an alternative inter-domain routing architecture. It distributes link-state information and supports three forms of routing: MPLS-like flow routing, BGP-like hop by hop routing and data packet specified routing. BANANAS [15] also distributes link state information and allows a sender to specify the full path for each packet. Although packet specified routing can help achieve load balancing and fault tolerance, it introduces other problems such as the difficulty of timely link-state propagation.

Frameworks and protocols for distributing policies within a domain that are based on MPLS, DiffServ or IntServ have been proposed, e.g., COPS and Bandwidth Broker [16], [17], [18], [19]. MPLS Fast-Reroute maintains backup paths and switching entries for every link computed from link state information flooded through the local network. In our solution, we focus on the *inter*-domain case and do not rely on the widespread deployment of DiffServ or MPLS.

III. Overlay Policy Control Architecture (OPCA)

A. Overview

The Overlay Policy Control Architecture (OPCA) is designed to support fault-tolerant and efficient wide-area routing. Our approach leverages intra- and inter-AS measurement and monitoring systems that are commonly deployed in ASes to obtain simple SNMP style data on traffic load changes and network performance. As shown in Figure 2, OPCA consists of five components: Policy Agents (PAs), Policy Databases (PDs), Measurement Infrastructures (MIs), the PA directory and the AS Topology and Relationship Mapper (RMAP). Policy Agents (PAs) are intelligent proxies that reside in each AS domain that agrees to participate in the overlay policy distribution network. Most Internet Service Providers (ISPs) today have deployed a measurement infrastructure to monitor the performance of their network. They also maintain a database of service level agreements (SLAs) and peering arrangements. We assume that OPCA can reuse any of such existing PDs and MIs within an AS domain. The PA directory and RMAP are new query-services introduced by OPCA.

The next section describes each of these components in detail.

B. Components of OPCA

B.1 Policy Agent (PA)

Intelligent Policy Agent (PAs) are introduced in each AS domain that employs OPCA to form an overlay policy distribution network. The PAs are responsible for processing external policy announcements, processing local AS policies, and enforcing these policies at border routers participating in external BGP sessions. This implies that PAs should have administrative control over the E-BGP speakers within their respective AS domains. The E-BGPs are dynamically (re)configured to reflect policy changes, and continue to perform route selection based on these policies. A centralized or distributed PA directory is needed to allow distant PAs (PAs belonging to different ASes) to communicate with each other. Each PA should be accessible at an IP address and port that can be reached from distant ASes.

A routing policy will be influenced by traffic measurements between the local AS and one or more distant ASes that are important, such as application level customers. To impose a change on the routing between these sites, a local PA will have to negotiate with the remote PA, and possibly other intermediate policy agents. Contacting the appropriate intermediate PAs will be important, since conflicting routing and filtering policies in other ASes along the route can severely impact the routing change. Having an understanding of the relationships between ASes along prospective routes [20] will be important to ensure

the effectiveness of protocol.

The protocol that the PAs use to communicate with one another is the new overlay policy protocol (OPP). The design of PAs and the OPP protocol is subject to certain constraints:

- The PAs should communicate with BGP speakers via conventional means, and should not require any modifications to the routers. This is important to the acceptability and deployment of this protocol.
- The PAs should detect and identify policy conflicts at runtime, and avoid potential BGP oscillations or divergence.
- OPP should co-exist with the widely deployed IGP/EGP today such as BGP, IBGP, IS-IS or OSPF.
- The use OPP should not increase BGP route flapping and the number of routing table entries.
- The correctness and effectiveness of OPCA should not rely on every AS employing PAs. We strive to support incremental deployment, i.e., early adopters of the new PAs should not be at a disadvantage compared to those continuing to use only BGP, even though the utility of the new system may increase as more ASes subscribe to OPCA.

B.2 Policy Database (PD)

The policy database is a local repository of information that will help the local PA decide how to change routing for its domain. The PD should provide the following data:

- *Ordered list of remote ASes containing the local domain's main customers.* This list identifies the target ASes that the PA should focus its load balancing efforts at. This list can easily be formed by examining the logs of the service provided by the local domain [21].
- *List of local application servers.* The PA needs to know which set of IP addresses serve content or any other service to the remote customers. The majority of traffic will likely be destined for or originate from these local servers. The PA will be concerned with the load balancing and fail-over of routes to these addresses.
- *Pricing constraints and SLAs.* The PA will try to balance traffic across multiple ISP links evenly weighted by actual link capacity. However, the local domain may prefer to weight traffic by pricing structures imposed by the SLAs it is operating under. If this is the case, the PA will need to be informed about these price constraints.

B.3 Measurement Infrastructure (MI)

Most ISPs and customer bases already employ some form of a measurement infrastructure (MI). Some may use it to verify Service Level Agreement (SLA) specifications with a third party, or may use it to manage their network. In our architecture, we assume that such an infrastructure already exists in each domain employing OPCA. The MI helps the PA keep track of the effects of the PAs alterations to routes and decide when such an alteration is necessary. The MI should provide the following data:

- *E-BGP link characteristics.* The PA needs data on each of the links connecting the local AS to the Internet. This data should include actual link capacity and current bandwidth load. This allows the PA to decide which links are underutilized and to verify the effect of policy changes that it imposes. PA control traffic will also go over these links.

TABLE I
INFERRED RELATIONSHIPS FOR 23,935 AS PAIRS

Relationship	# AS pairs	Percentage
Provider-customer	22,621	94.51%
Peer-peer	1,136	4.75%
Unknown	178	0.74%

TABLE II
DISTRIBUTION OF ASES IN THE HIERARCHY

Level	# of ASes
Dense core (0)	20
Transit core (1)	129
Outer core (2)	897
Small regional ISPs (3)	971
Customers (4)	8898

- *Customer-server traffic characterization.* The MI should also provide data outlining traffic characteristics (bandwidth, average latency, jitter, etc.) that each main customer sends/receives to/from each local server. This helps the PA understand the current traffic distribution and how to best redistribute this load.

B.4 PA Directory

The PA directory is a means by which a local domain's PA can locate the address for a distant AS's PA. This is necessary because some ASes may not have PAs, since we do not rely on immediate wide scale deployment of OPCA, and those that do have PAs can place them anywhere inside their network. The design of the directory is not a goal of our work. The system can use one or multiple directories, which may or may not be coherent with each other. From the architecture's point of view, there is logically one PA directory. A simple solution such as using the already deployed DNS system can be used.

B.5 AS Topology & Relationship Mapper (RMAP)

The RMAP is a repository of the inter-AS relationships and Internet hierarchy from our prior work [20]. These relationships determine how routing and traffic flows on the Internet as governed by route export rules. In the RMAP, we deduce these relationships from multiple BGP routing tables, and then infer the hierarchical structure of the AS topology that exists today.

We apply heuristics based on commonly followed route export rules to the multiple BGP routing tables that we collect. We infer whether an AS-AS link represents a peer-peer or a provider-customer relationship. The results are summarized in Table I. Using pruning, greedy ordering and weak cuts designed to expose the different business classes of Internet service providers, we also formulate the 5 level hierarchy shown in Table II.

Using our hierarchy characterization, we can study the growth of multihoming. We use our current list of customer ASes and identify multihomed customer ASes as those making route announcements via multiple upstream ASes. Note that these values are lower bounds as some ASes and links may not be visible from the BGP tables that we use. We use routing tables from many time periods [4] to show the change in multihomed

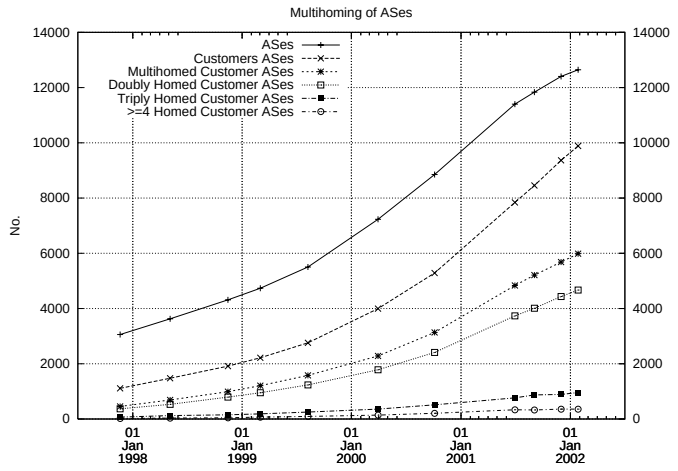

Fig. 3.

customer ASes over time. In Figure 3, we show that the large growth in the number of ASes has been fueled by customer ASes, most of which are multihomed.

The RMAP is an essential component to OPCA. The architecture is indifferent to whether there is only one global RMAP or if multiple RMAPs exist for different ASes. The only requirement is that the RMAP have access to enough diverse BGP routing table dumps as to be mostly accurate. We will revisit this issue in Section III. PAs need the RMAP to find the likely path a distant AS uses to reach it and the intermediate AS relationships that determine how routes propagate to the distant AS. This also helps to determine if a policy request will conflict with a distant AS's local routing policy.

C. Applications of OPCA

In this section, we briefly describe how OPCA achieves fast fail-over of inter-domain routes and inbound load-balancing.

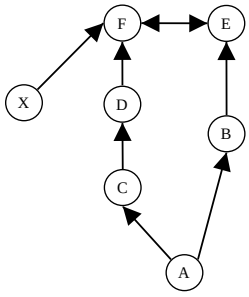

Fig. 4. Routing Scenario

C.1 Improving Route Fail-Over Time

From Labovitz [7], we know that the fail-over time for BGP routes can be as long as 15 minutes or more and that this depends on the length of the longest path. Consider the trivial example shown in Figure 4. An arrow from AS A to B indicates that B is A's provider. A double-ended arrow indicates a peer-peer relationship. Suppose D is using the path via F to reach A. If the A-B link fails, A would want D to quickly switch over to using the route through C, because most of A's important customers reside in D. Slow BGP fail-over would involve B timing out and sending a route withdrawal to E, which then after some delay sends a withdrawal to F and then to D, at which point D will re-run route selection. Instead, in OPCA, A's PA can contact D's PA directly and request the change, thereby shortcutting the long BGP convergence. Alternatively, if D does not have a PA, A can contact F. This is a very simple example. A more complicated scenario will be more common, where various routing pathologies can cause normal BGP route convergence to be especially delayed.

The advantage of our architecture is that PAs can directly communicate with other PAs when necessary instead of using a multi-hop, dampened protocol like BGP. This allows the benefits of a fully meshed communication architecture, without the pitfalls because the actual routes still propagate through the underlying hierarchical BGP structure. It is important to note here the importance of the RMAP component. It is important for A's PA to know what the topology, as shown in Figure 4, is in order to contact the relevant remote PAs. Also, it is important to know if alternate routes are feasible. In this scenario, both the D-F-E-B-A and D-C-A paths are valid. However, if D and F were peers instead of customer and provider, then the D-F-E-B-A path would be invalid based on commonly followed route export rules [20].

C.2 Incoming Traffic Load Balancing

Direct PA to PA communication also helps to achieve incoming traffic balancing. Consider again the example in Figure 4 where AS A is load balancing its links to C and B, primarily for some application level customer X. With respect to X, F is the aggregation point for routes to A via either C or B. Pathological route updates need not be sent in BGP by A to blindly "game" route selection at F. Instead, A's PA will contact F's PA directly and request specific route selection at F. Again, A's PA has to first determine the structure of this graph and that F is the aggregation point. It also has to determine whether its route selection requests will conflict with F's local route selection policies based on F's peering arrangements. The RMAP helps the PA in these respects.

C.3 Other Applications

OPCA can be applied to solve other problems beyond our stated goals of fast fail-over and traffic balancing. It can be used for querying the BGP table at remote ASes. This can be used to diagnose router misconfigurations or find the source of malicious BGP advertisements. It can potentially be used to also arrange per address prefix micro-peering [22].

D. Evaluation

We plan on evaluating our architecture in an emulation testbed. We will use about 50 Linux PCs connected via a high speed LAN. We will run multiple software Zebra BGP routers [23] and PAs on each machine which we can connect over TCP into any arbitrary AS-level topology, extracted from real, sample scenarios we have observed in [20]. We will inject the BGP routing tables that we have collected into the software

routers. We can then induce faults by shutting down specific routers and measure the convergence time. We plan on using NIST Net [24] to allow us to emulate wide area packet delays. We will use fictitious traffic traces to evaluate OPCA's load balancing capabilities.

One main metric of success is the reduction in the time to fail-over from a primary route to a backup route for a destination prefix. The other metric is the time it takes to shift incoming traffic from one route to another. We will also quantify how effective OPCA can be given the current AS-level structure of the Internet. That is, the current deployment of ASes and inter-AS links will determine how many diverse paths exist for each stub AS and how many upstream aggregation ASes exist. This will also determine how well OPCA will scale, as we will describe in Section IV.

IV. OPCA Design Choices

In this section, we discuss our design rationale and address several deployment and scalability issues. We begin by examining why using a separate control path from BGP helps us achieve our goals. We also discuss the use of multiple BGP views to improve the accuracy of OPCA.

A. Policy Control Path v. Routing Path

In theory, we could propose our own BGP community attribute and distribute policy control information along with BGP routes. However, the advertiser will not know a priori which remote BGP speakers adhere to the new community specification, which diminishes the reliability of such approach. Further, the system would suffer from the same BGP convergence times.

OPP carries inter-PA messages directly over TCP between the PAs by rendezvousing through the PA directory. Therefore, PA control messages do not have to through every PA in the intermediate ASes the same way as BGP announcements propagate from one BGP speaker to another, because the originating PA has the burden of determining the appropriate destination PA to contact directly. The advantages of this approach are

- eliminate intermediate PA processing
- keep convergence time of PA messages lower than that of BGP messages
- give originating PA more control
- give more privacy to originating PAs

However, some PA messages will be affected by the route changes themselves during failures or route re-convergence phases. Unreliable one-way UDP messages can be used to send policy changes upstream. Alternatively, each PA can have multiple addresses, one in each disjoint address range advertised. In rare scenarios of multiple concurrent outages, multi-hop PA communication may be required.

B. Accuracy, Correctness and Security

Our architecture's core algorithm relies on inferring the AS-level topology, AS relationships, the likely paths between two points and the path convergence points. The efficiency of our algorithm will rely partly on the accuracy of these inferences from the RMAP.

We built the RMAP because no oracle exists that can be queried about the AS structure of the Internet and the relationship between any pair of ASes. Due to the lack of an oracle, we cannot check the accuracy of our inferences. We have verified our inferences by comparing them to additional routing table views [20] and in most cases found fewer than 3% errors. We also have contacted two major ISPs and verified the list of AS relationships that involve them. We exploit access to multiple routing tables to improve both the completeness and the accuracy of our inferences. If more ASes participate in this architecture, then more views can be used.

The RMAP improves the efficiency of the protocol by quickly identifying likely route paths, route convergence points, and routing relationships between ASes at and near the convergence points. If no RMAP existed, OPCA can continue to function, but with much more signaling overhead. A PA would have to contact several distant PA's to find out where routes that it is sending are going and find convergence points for its routes. Once found, the PA would have to make several policy change requests, many of which may be denied due to conflicts with neighboring AS relationships.

PAs will not apply routing policies that conflict with their own local peering policies. In this way, the ill effects of having an inaccurate RMAP or having malicious PAs can be kept to a minimum. The OPP protocol will only allow simple requests to be made such as a request for route information, for route selection change or termination of route propagation. None of these requests would violate the BGP protocol. The BGP protocol already allows each AS to use practically any form of route selection by the application of local policies. We can incorporate a standard form of authentication and address ownership verification to the PA protocol as is proposed for BGP itself [25]. This would ensure that an AS can only influence routing for its own address space.

C. Deployment

OPCA primarily benefits multi-homed stub ASes. However, if only the stub ASes deploy policy agents, then they will receive no benefit from the system. The utility is maximized when all the different points where different routes converge have policy agents. This is because these points can control which route to use and propagate to their neighbors. Due to the hierarchy that exists at the AS-level [20], most of these aggregation points may be in the core of the Internet. The motivation for these core ASes to deploy policy agents will arise from their business relationships. If our architecture offers enough of a gain, then stub ASes will want to urge their upstream providers to deploy policy agents, and in turn urge their upstream providers to do the same. Clearly, a large benefit has to be shown in our evaluation and we have to show that the system is robust.

D. Scaling

We also need to show that the system will scale to the number of ASes that exist today and that can exist tomorrow given today's growth trends. If stub ASes choose multiple providers for fail-over, they will want to choose providers that do not also have the same upstream provider. The result may be multiple paths for reaching the stub AS that are as uncorrelated during failure as possible. In this case, most of the aggregation points where multi-homed paths converge will be in the dense

core of the Internet. This dense core consists of about 20 large ISPs [20].

If only 20 ASes receive the PA requests from all the stub ASes, then their PAs may not be able to keep up with the number of requests. We can have multiple PAs for each AS and segregate requests by originating AS. This will reduce the load on each agent. Also, we can construct a hierarchy of PAs inside each large AS that will aggregate requests so that the number of route changes required at each EBGP router will be manageable. If a small number of ASes receive most of the PA requests, they may be able to do more intelligent global optimizations across the different requests. We plan on investigating how diverse multi-homing paths are and how many route convergence points exist today.

V. Summary

We believe that the motivation behind the large increase in multi-homed stub ASes is in achieving fast fail-over and traffic load balancing. BGP today offers slow fail-over and limited control over incoming traffic. We address these issues by developing an overlay control architecture that will coexist with and interact with the existing BGP infrastructure. We have explained how the protocol allows our goals to be met and outlined some design decisions that affect deployment, scaling, choice of control path and accuracy. We plan on developing and testing this system in an emulation testbed running software BGP speakers.

Acknowledgements

We would like to thank Ion Stoica (UC Berkeley), Anthony D. Joseph (UC Berkeley) and Jennifer Rexford (AT&T Research) for their valuable feedback on early versions of this work.

References

[1] J. W. Stewart, *BGP4: Inter-Domain Routing in the Internet*. Addison-Wesley, 1998.
[2] T. Bu, L. Gao, and D. Towsley, "On routing table growth," in *draft paper*, November 2001.
[3] G. Huston, "Analyzing the Internet's BGP Routing Table," Cisco Internet Protocol Journal, March 2001.
[4] "University of Oregon RouteViews project." `http://www.routeviews.org/`.
[5] "Route Science Web Page." `http://www.routescience.com/`.
[6] R. Mortimer, "Investment and Cooperation Among Internet Backbone Firms," Ph.D. dissertation, Economics Department, UC Berkeley, 2001.
[7] C. Labovitz, R. Wattenhofer, S. Venkatachary, and A. Ahuja, "The impact of Internet policy and topology on delayed routing convergence," in *Proc. IEEE INFOCOM*, April 2001.
[8] G. Huston, "Architectural requirements for inter-domain routing in the Internet," Internet Draft 01, Internet Architecture Board, May 2001.
[9] G. Huston, "NOPEER community for route scope control," Internet Draft 00, August 2001.
[10] O. Bonaventure, S. Cnodder, J. Haas, B. Quoitin, and R. White, "Controlling the redistribution of BGP routes," Internet Draft 02, February 2002.
[11] T. Hardie and R. White, "Bounding longest match considered," Internet Draft 02, November 2001.
[12] D. Estrin, J. Postel, and Y. Rekhter, "Routing arbiter architecture," 1994.
[13] D. G. Andersen, H. Balakrishnan, M. F. Kaashoek, and R. Morris, "Resilient overlay networks," October 2001.
[14] I. Castineyra, N. Chiappa, and M. Steenstrup, "The Nimrod Routing Architecture," RFC 1992, Network Working Group, August 1996.
[15] S. K. et al, "BANANAS: A new connectionless traffic engineering framework for the internet," draft paper, 2002.
[16] R. Neilson, J. Wheeler, F. Reichmeyer, and S. Hares, "A discussion of bandwidth broker requirements for Internet2 Qbone deployment."
[17] R. Yavatkar, D. Pendarakis, and R. Guerlin, "A framework for policy-based admission control," RFC 2753, IETF, January 2000.
[18] D. Durham, J. Boyle, R. Cohen, S. Herzog, R. Rajan, and A. Sastry, "The COPS (COmmon Open Policy Service) Protocol," RFC 2748, IETF, January 2000.
[19] D. O. Awduche, A. Chiu, A. Elqalid, I. Widjaja, and X. Xiao, "A Framework for Internet Traffic Engineering," draft 2, IETF, 2000.
[20] L. Subramanian, S. Agarwal, J. Rexford, and R. H. Katz, "Characterizing the Internet hierarchy from multiple vantage points," *Proc. IEEE INFOCOM*, 2002.
[21] B. Krishnamurthy and J. Wang, "On network-aware clustering of web clients," in *Proc. ACM SIGCOMM*, August 2000.
[22] R. Mortier and I. Pratt, "Efficient network routeing," unpublished project proposal, 2000.
[23] "GNU Zebra, free routing software." $http://www.zebra.org/$.
[24] "NIST Net Emulation Web Site." `http://snad.ncsl.nist.gov/itg/nistnet/`.
[25] S. K. Charles, "Secure Border Gateway Protocol (S-BGP) — Real World Performance and Deployment Issues."