Computer Science
Technical Report

Simulating Internet Scale Topologies with Metarouting

Steve DiBenedetto, Andrew Stone, Michelle Strout, Dan Massey

Department of Computer Science
Colorado State University
Fort Collins, Colorado

March 31, 2010

Technical Report CS-10-103

Computer Science Department
Colorado State University
Fort Collins, CO 80523-1873

Phone: (970) 491-5792 Fax: (970) 491-2466
WWW: http://www.cs.colostate.edu

Simulating Internet Scale Topologies with Metarouting

Steve DiBenedetto, Andrew Stone, Michelle Strout, Dan Massey
Department of Computer Science
Colorado State University
Fort Collins, Colorado

***Abstract*—Research on Internet routing algorithms can benefit greatly by easily repeatable large-scale simulations conducted on realistic network topologies. Unfortunately, it is a challenging task to develop clear and unambiguous definitions of a routing protocol and its associated policies. Simulations often only operate on small scale topologies. Even if simulators could handle very large scale topologies, it is unclear whether we have accurate topologies that also capture policies. This paper leverage existing work on metarouting[6] to provide a large scale simulator. The simulator takes as input a topology and routing protocol/policy described using RAML[6]. We use a RAML description of BGP to evaluate the accuracy of some of the most popular AS level topologies. Our results reveal both strengths and weaknesses in these topologies. In particular, the results suggest the addition of a siblings policy actually reduces accuracy and topologies could improve by instead incorporating rules for selective prefix announcement.**

I. Introduction

Routing simulators allow us examine scaling questions, evaluate route security, the explore the potentially far reaching effects of policy changes. However, today's simulators face a multitude of challenges. Protocol specifications may be unclear or leave too much room for interpretation from the developer. Inconsistent routing protocol specifications make it difficult to understand what is actually being simulated. For example, a BGP experiment might be done in a lab testbed using vendor equipment (Cisco/Juniper). However, using all Cisco or all Juniper equipment could produce different results and it is not feasible to run such an experiment at large scale. Alternatively, a simulation might create virtual BGP routers based on an implementation of BGP found in say Zebra/Quagga code, but a particular behavior may be version specific or specific to a particular simulator.

Which routing policies to use are even more important. Policies drawn from [4][11] are widely recognized to be better than shortest paths, but are not perfect. The degree to which policy inaccuracies will impact simulation results may vary depending on how the policy is used and the question being studied. Furthermore, the research question may center on the policies themselves. In the end, it is hard to describe the relevant parameters in a *precise and repeatable manner* so other researchers can run experiments for themselves or explore different trade-offs.

All this must also scale to Internet size topologies and this often requires large amounts of computing power, on the order of a computing cluster [12].

Finally, even if researchers have a precise specification and Internet-scale simulator, what topology should be used to perform the experiments? CAIDA [1] and UCLA's Internet Research Lab [2] try to measure the autonomous system (AS) level topology of the Internet. But questions still remain as to how well these different approaches capture the true Internet topology and the policies in use at the AS level.

Our work address the above problems. We present a routing simulator built upon our own implementation of Griffin and Sobrinho's Routing Algebra Meta Language (RAML). As such, we are able to [6][9] to precisely specify routing protocols in and understandable and repeatable way. Users can compare protocols and policies, easily add new protocols, and make modifications to existing specifications. RAML is reviewed in Section II, but we also direct interested readers to [6][9] for more in-depth explanations.

Our simulator is designed to answer ”What-if” questions[5] rather than details related to convergence time or number of messages exchanged. This approach allows us to easily scale to Internet AS level topologies with minimal resource usage and answer questions about the resulting routes. This can be used to explore the impact of policies, how topological changes impact routing, how new routing protocols might fare in the Internet, and a host of other questions. Further details on our simulator are provided in Section III

We apply this simulation technique to two popular AS level topologies [1][2] and compare the resulting routes against the actual routes as seen from Internet routers who peer with RouteViews [8]. Section IV both illustrates how our simulation approach works and examines the accuracy of these AS level topology models. The results help illustrate both strengths and weaknesses in the AS level topologies.

II. MetaRouting Background

Typically, a document such as an RFC will provide the knowledge needed to correctly implement a protocol. However, there may be important details not conveyed well in the text or even such a document to begin with. Ideally, researchers could express their routing protocol ideas in a standardized, easy to understand, and expressive language. We believe Griffin and Sobrinho [6] have done exactly this in the creation of the Routing Algebra Meta Language (RAML).

RAML provides its users with a succinct way to express both simple and complex routing protocols. In essence, the algebras specified within RAML are analogous to building blocks which may be joined together to form more complex blocks. Each algebra is a mathematical function which when given a network edge *label* and a route's *signature*, or measure of preference, will produce a modified signature. Here, it is convenient to separate the link connecting two nodes into separate inbound and outbound edges to which different labels may be applied. In order to guarantee a protocol will converge, a route's signature must monotonically decrease in preference as more labels are applied [6].

To test our simulator and examine the accuracy of AS level topologies, we need a RAML description of BGP. While Griffin and Sobrinho present a more detailed RAML form of BGP in [6], we were forced to choose a smaller set of algebras to describe BGP due to what is readily available in AS-level Internet topologies [1][2]. RAML allows one to lexicographically combine multiple algebras to form a more complex one and we lexicographically combined Forced Monotonicity, Local Preference, Sequence, and integer Minimum (as a stand in for router ID) to form an approximation of BGP.

Local Preference: The local preference attribute is well known for providing administrators flexibility and its potential to prevent route convergence. While it is possible for administrators to chose a wide range of values with organization dependent interpretations, the routing policies tend to be characterized by the business relationship between peering ASes [11]. More specifically, most organizations follow a strict preference ordering of routes received from customers over peers. Both of these types of routes are in turn preferred over routes received from providers.

Forced Monotonicity However, even this simplification of local preference can violate RAML's requirements for convergence since it allows routing valleys. In order to guarantee the simulator will converge to an answer, we also apply a forced monotonicity algebra to local preference as shown in [6]. While valleys are used in the Internet, very few organizations choose to do so.

AS Path: The AS path algebra allows for path construction in a similar fashion to how actual routers would so so. Each announcing AS will prepend its number to the path. The actual label application to the route's signature also triggers a check to ensure the receiving AS is not currently present in the path. While the receiving router would normally be responsible for this loop check, this modification was necessary to allow the underlying simulator to remain independent of any one policy representation.

Router Identifier: Since the router identifier is not available at this level of abstraction, we instead leverage the AS number of the peer announcing the route. The unique number of all ASes provides us with a strictly monotonic component which allows RAML to guarantee the policy will converge to a deterministic solution.

Note the claim here is not that this is a perfect description of BGP. Instead, our claim is that is a precise, repeatable, and easily modifiable description of BGP. This RAML description is available for download just as a topology file is available. RAML provides an unambiguous easily parsed definition. Any researcher can easily repeat our simulation or run the same simulation after modifying the RAML description in anyway the researcher feels might be beneficial.

III. The Simulator

Wojciechowski [12] was able to simulate Internet topologies with BGPSIM. However, BGPSIM is restricted to simulating BGP and requires multiple machines to do so. In comparison, our policy simulator is capable of evaluating any policy that may be expressed by Metarouting on a single workstation. The work of Feamster and Rexford [5] is similar in spirit to our own. Policy configuration requires knowledge of what impact changes will bring. In order to do so, fast calculations which may omit information such as message passing or convergence times are necessary. While Feamster and Rexford solve this problem with algorithms designed to handle BGP, our usage of Metarouting and its previously proven algebra properties provide us a generalized approach.

Our simulator takes as input a RAML description of the routing protocol and policy and a topology. The topology consists of a list of edges (expressed as node X connects to node Y) and policies associated with the edge. The policies depend on the RAML specification. This approach is not specific to any routing protocol, but can be applied to any protocol whose protocol and policy can be expressed using RAML.

Given a RAML algebra, which may be the result of lexicographically combining several algebras, and a topology file, the metarouting simulator constructs a policy annotated graph. The execution of a simulation is broken in rounds where a set of nodes announce exactly one route each. This route is an abstraction which acts as a representative route for the many routes a particular node may normally originate. The route data structure used within the simulator consists of the destination (*i.e.* the originating AS in the case of BGP AS topology), a RAML signature, and the next hop.

This simulation was designed to be highly parallel. A full Internet AS level topology can be run on an off-the-shelf workstation with limited memory. Adding adding resources or additional processors speeds up the simulation. A complete description of the underlying simulation design and its performance evaluation can be found in [10]. [10] also provides more details on how to run the simulator for arbitrary RAML specifications.

A. Simulating BGP and AS Level Topologies

To provide a concrete example, this paper focuses on BGP and BGP related policies. In [6], Griffin and Sobrinho demonstrate many components of BGP can be easily expressed in RAML. However, the available AS-level topologies focus on providing information concerning the classification of peering lines in rather than MED and community values. Without an accurate algorithm to infer these values, we instead chose to express BGP as the lexicographical combination of local preference, AS path, and a router identifier. We emphasize that given a topology with such values, we could easily extend our BGP approximation to be more accurate.

Each graph node represents one AS from the topology with edges labels representing peering relationships. In the simulator's terminology, the equivalent of announcing a route to a neighboring AS is applying the label attached to the edge connecting to ASes. Under RAML's requirements for convergence, a valid algebra will result in a signature becoming equally or less preferred after each label application. If the announcing AS applies a label which results in the signature reaching a value of ϕ (no route), the would be receiving AS is not informed of the route. Depending on peering relationships, a label may not be able to propagate to one neighbor while still being available to others.

When a peer does receive a route it compares the signature of the new route against the best, if any, route for the specified destination. The simulator assumes that it is always preferable to have a route over none. If the peer determines the new route is more preferred, it replaces its previous route and the peer itself is added to a work set. Each node in the work set is dequeued one at a time and will re-announce any routes in its routing table which have changed. The round of simulation is finished when the work set is empty and a new round immediately begins with a new set of originators being added to the work set. Once all nodes have been given a chance to originate their representative route, the simulation is finished.

IV. An Analysis of AS Level Topologies

A number of research projects have analyzed the BGP AS-level topology in attempt to infer both AS level connectivity and the AS level routing policies [7], [2], [1]. Approaches such as [7] can generate graphs that are intended to capture the essential features of the Internet AS level topology and the user can generate different types of topology by specifying different parameters such as overall graph size. Other approaches such as [2], [1] use measurements and inferences to produce an estimate of the actual AS level connectivity in the current Internet. All the approaches provide a graph of AS level connectivity and policy information that can be associated with a link. For example, a topology entry may specify there is a link from ASN X to ASN Y and specify that ASN X is *customer* of ASN Y. For the UCLA topology[2], each directed link is labeled as either a *customer, provider, or peer* link. For the CAIDA topology[1], links may be labeled as *customer, provider, peer, or siblings*. This choice of policies generally follow the direction suggested in [4], [11] which also provides more details on the distinctions between the various types. The net result is that there are several estimates for the Internet's AS level topology and these estimates are used in a wide variety of Internet routing research.

Given such a topology and a RAML description of BGP, one can use our metarouting simulator to generate routes between all nodes in the topology, explore how topology changes, policy changes, or routing protocol changes might impact Internet routing. But as with any research that relies on these topologies, the accuracy of any simulation results will be at least partly dependent on the accuracy of the protocol, topology, and policy models. It is widely accepted that no topology captures all links in the AS level topology. The *customer, provider, peer* policy estimates are considered greatly superior to other metric such as AS path length, but an ASN is not required to consider links as customers or providers or peers (or siblings). Even if the *customer, provider, peer* model is correct, the inference of the policy on a particular link may be inaccurate. Finally, we showed how to approximate BGP in RAML but we did not include attributes such as BGP communities. Any representation of BGP itself may not exactly match the BGP protocol implemented in routers. A great deal of Internet routing research has made use of these topologies and various approximations of BGP, but relatively little has been shown about the accuracy of the underlying models.

To better understand the accuracy of both the RAML description and the topology, we ran metarouting simulations using both the UCLA[2] and CAIDA[1] topologies. Ideally, every ASN in the Internet has a corresponding entry in both the UCLA and CAIDA topologies. Running the metarouting simulator on such a topology produces a set of Internet routes at the AS level. For example, our simulator produced the AS path from a large provider such *AT&T* to a prefix in an edge site such as *Colorado State University*. Using data collected by BGP monitors at RouteViews[8], we can obtain the actual AS paths used by ISP routers and compare these actual paths with our simulator results. If the RAML description and topology model are correct, the simulator produced AS paths should match the actual AS path.

More precisely, we used the RAML description of BGP discussed above and the topology (including the inference of policy on each ASN-ASN link). Each ASNs in our simulation announced one prefix and we computed a full set of routes from every ASN to every announced prefix. The resulting routes were output by the simulator. To obtain the "actual Internet route", we used the December 2009 BGP routing tables at Oregon RouteViews[8]. These tables provide the actual routes from an *AT&T* router to roughly 350,000 Internet prefixes. We compared these actual routes to the routes produced by our simulator.

We repeated the comparison using routing tables from other RouteViews peers, but due to space limitations we do not show the results for every peer. In addition to view from AT&T, we also include the view from colocation provider Hurricane Electric and the Kenya Information Center (KENIC). AT&T and Hurricane Electric both report their full BGP routing tables and provide routers to 300,652, and 303,002 prefixes (respectively). The router at KENIC only reports prefixes from 287 prefixes, but it also provides a diverse view of the Internet and shows our results generally hold for diverse locations. Furthermore, KENIC did not appear in CAIDA's December 2009 data set so we were forced to shift our analysis for CAIDA, UCLA, and RouteViews to November 2009.

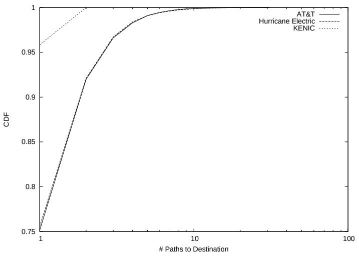


Fig. 1. CDF of the number of different routes to destination ASes

A. Limitations of Our Methodology

Representative Prefixes: In our simulation, each ASN announced a single “representative” prefix. An actual ASN can (and typically does) announce multiple prefixes. However, the UCLA and CAIDA topologies captures only ASN level information. As a result, all prefixes announced by the same ASN will encounter identical simulation conditions and produce an identical AS level paths. For example, suppose Colorado State University announced prefix 129.82.0.0/16 and 129.82.138.0/24. If topologies and RAML models were entirely accurate, AT&T’s AS path to 129.82.0.0/16 would be exactly the same as the AS path it takes to reach 129.82.138.0/24.

Figure IV-A uses the data from the actual BGP routing tables to determine how often prefixes from the same ASN follow the same path. The graph shows that for roughly 75% of ASNs, the AT&T router has exactly one AS level path to every prefix originated by that ASN. In other words, 75% of the ASNs can be simulated using exactly one representative prefix. At the other extreme, the AT&T router had 31 distinct paths to prefixes originated by AS 21433 and AS 22394. At best, our simulation can hope produce one of these 31 paths. The results from the router at Hurricane Electric are similar. The KENIC peer sees far fewer path differences, but KENIC only reports routes to 287 prefixes compared with over 300,000 prefixes reported by the other peers.

If there are multiple AS level paths to the same ASN, our simulation will hope to capture at least one of these multiple paths. To produce additional paths, the topologies need to be enhanced with prefix level information and an interesting open question is whether such a model could be produced.

Missing Autonomous Systems: Some ASNs exist in the real routers from AT&T, Hurricane Electric, and KENIC, but were not present in the UCLA and CAIDA topologies. Since these ASNs did not appear as input to our simulation, the simulation clearly cannot produce correct routes to prefixes announced by the ASNs. Table I shows the breakdown of these missing ASes among different combinations of ISPs and simulation topologies. Both UCLA and CAIDA miss a small number of ASNs, but the AS level topology is a constantly changing system with new ASNs added, some ASNs dropping out, and any number of transient changes. In total topology of roughly 30K ASNs, the two topologies are missing less than 0.006% of the ASNs. For the rest of the analysis, all prefixes originating from these “missing ASNs” were removed from the Oregon RouteViews tables.

	AT&T	Hurricane Electric	KENIC
CAIDA	173	177	7
UCLA	174	174	0

TABLE I
MISSING ASES FOR EACH TOPOLOGY AND ISP COMBINATION.

	AT&T	Hurricane Electric	Mohawk	KENIC
CAIDA	534	453	603	33,061
UCLA	2,396	2,311	2,472	34,887

TABLE II
EXTRA ASES FOR EACH TOPOLOGY AND ISP COMBINATION.

Phantom ASNs: The simulation topologies also include a number of ASNs that *do not appear* in the actual routing tables router. In other words, the topologies report a number of ASNs exist but the actual routers either cannot reach these ASNs. Due to policies, some ASNs may be unreachable some locations and thus will not appear in the actual routing tables. For example, the routing policies on some links may prevent AT&T from reaching a remote ASN. In this case, we would expect the simulation to also produce “no route” to the ASN’s representative prefix.

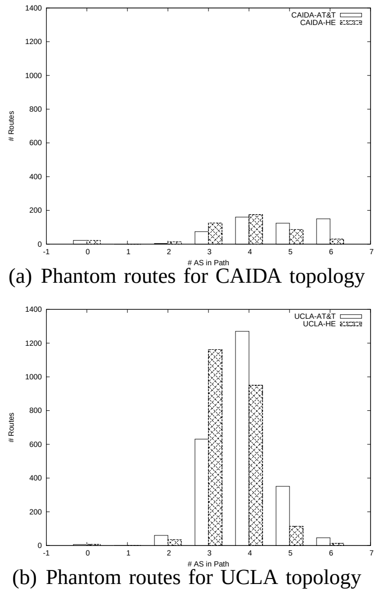
However, there are large number of ASNs that our simulations says should be reachable. Table II shows the number of “Phantom ASNs” that our reachable in our simulation but are not present in the actual routing tables. Since KENIC only reports 287 prefixes, its is not surprising that a large number of ASNs will not appear in the KENIC routing table. However, the table shows UCLA has over-estimated the number of ASNs (or has incorrectly inferred the policies leading to those ASNs). For each router other KENIC, UCLA reports the existence of over 2,000 ASNs that did not appear in actual routing tables.

To better understand these “Phantom ASNs”, Figures 2(a) and 2(b) show path lengths used to reach these ASNs. In the UCLA topology simulation, AT&T could reach 1,270 Phantom ASNs using an AS path of length 4. But the actual AT&T router peering RouteViews had no route to any prefix originating in these ASNs. The fact that a large ISP such AT&T cannot reach these prefixes, combined with the fact that UCLA sees an order of magnitude more “Phantom ASNs” than the CAIDA topology suggests the UCLA needs to improve its pruning of old ASNs.

These results also show that out simply having more ASNs is not necessarily a feature of an AS level topology. The UCLA topology has more ASNs than the CAIDA topology, but both UCLA and CAIDA miss roughly the same number of ASNs that actually appear in our sample routing tables (see Table I). The vast majority of ASNs that appear only in the UCLA topology do not have reachable prefixes when viewed from the RouteViews monitors used in this study.

B. Perfect Matches

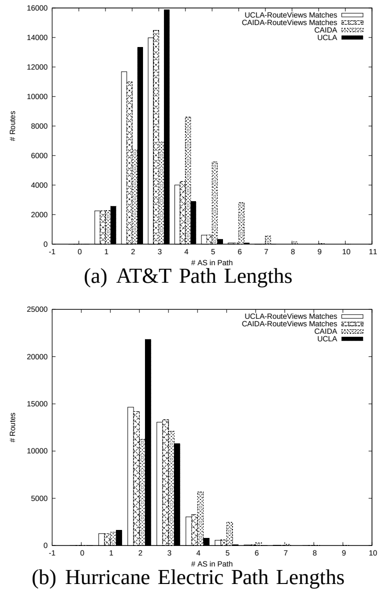
After removing the less than 200 “extra ASNs” that do not appear in the simulation and the “Phantom ASNs” that only



(a) Phantom routes for CAIDA topology

(b) Phantom routes for UCLA topology

Fig. 2. Path lengths of simulator phantom routes.



(a) AT&T Path Lengths

(b) Hurricane Electric Path Lengths

Fig. 3. Comparison of path lengths

appear in the simulation, we are left with simulation produced AS paths to over 30,000 ASNs. Viewed from a particular point such as AT&T, each simulation produced path should match the actual AT&T path leading to that same ASN. As noted above, in some case AT&T may have multiple distinct paths to a particular ASN. In the case, it is hoped the simulation will produce one of these actual paths. Table III shows the number of simulated routes that exactly match the route observed in the real topology.

The two topologies are remarkably close in the number of simulation paths that match precisely. Interestingly, there have been concerns over the how well the *customer, provider, peer* policies actually capture real ISP policies. Partly in response to these concerns, CAIDA used a more complex policy that also allows links to be designated as *siblings*.

C. Near Matches

The simulation perfectly predicts a large number of AS paths, but there are also a number of paths where the simulation does not match any path found in the actual ISP routing tables. To better understand the degree to which the simulation missed, we compared the simulated paths with the actual paths seen in the real topology. In some case, the real routing table may provide multiple paths to same ASN. In this case, we look for the most similar RouteViews path where "most similar" is defined as one sharing the same originating AS and having the minimum Levenshtein string distance[3]. Essentially, this algorithm counts the number of AS substitutions needed to make the generated path exactly match a RouteViews path with the same originator.

Figure 3(a) compares the simulated path lengths with their equivalent RouteViews route as seen from AT&T. For reference, routes AT&T originates have a length of one. There is only one AT&T routing table, but two distinct bars are shown in the graph. If an actual AT&T path does not exactly match the simulation path, we find the AT&T path that is "most similar" (as defined) above to the path produced by the simulation. This "most similar" path can vary depending on whether the simulation used the UCLA or CAIDA topology. As a result, we plot the path lengths for both the AT&T path that most closely matched UCLA (first bar) and the AT&T paths that most closely match CAIDA (second bar).

Similarly, Figure 3(b) compares the simulated path lengths with their equivalent RouteViews route as seen from Hurricane Electric.

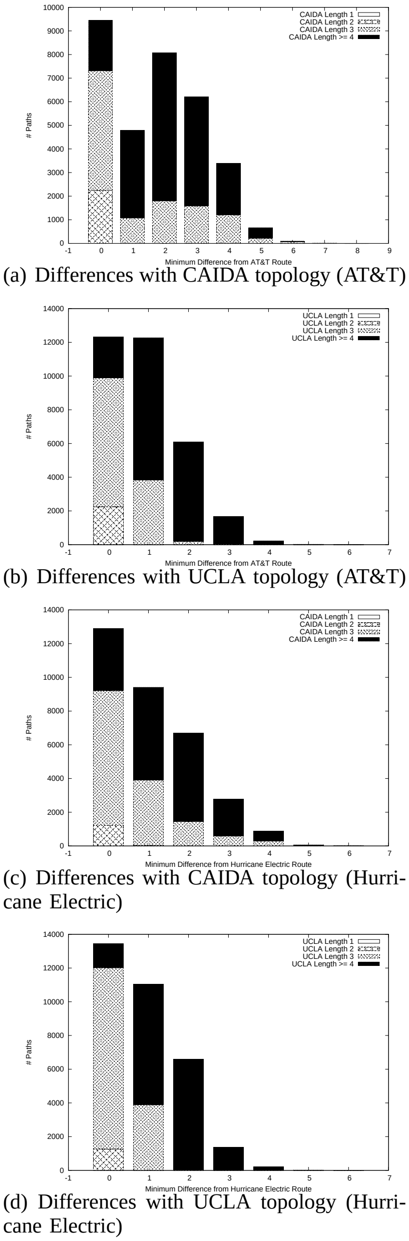
The simulated paths produced using the UCLA topology are shorter than those created with the CAIDA topology. Note that CAIDA topology includes the additional *sibling* relationship. Siblings are two ASNs that belong to the same organization despite having different AS numbers. The RAML policy algebra only allows for local preference labels to be applied if the receiving AS is not a sibling. Contrary to our initial intuition, the addition of this seemingly more accurate policy statement did not improve path matches and may have contributed the longer path lengths.

We further examined the simulated paths based on the Levenshtein distance of the paths from their RouteViews counterpart. Figures 4(a) and 4(b) compare the CAIDA and UCLA topologies to the AT&T paths. As noted above, both topologies tend to create a nearly equal number of exact matches. However, the UCLA topology produced a larger number of paths that were off by only one ASN. The paths produced using the CAIDA topology tend to be longer which in turn lead to less accurate paths.

Similarly, Figures 4(c) and 4(d) compares the CAIDA and UCLA topologies to the Hurricane Electric paths. For this AS, the two topologies produced roughly the same number of exactly matching routes. However, these correct routes seem

	AT&T Total ASNs	Exact Matches	Hurricane Electric Total ASNs	Exact Matches	KENIC Total ASNs	Exact matches
CAIDA	32,258	12,159 (37.7%)	32,420	13,485 (41.6%)	41	14 (31.1%)
UCLA	30,395	12,321 (40.5%)	30,565	13,449 (44.0%)	48	25 (52.1%)

TABLE III
SIMULATED ASN PATHS MATCHING ACTUAL PATHS



(a) Differences with CAIDA topology (AT&T)

(b) Differences with UCLA topology (AT&T)

(c) Differences with CAIDA topology (Hurricane Electric)

(d) Differences with UCLA topology (Hurricane Electric)

Fig. 4. Comparison of path differences for AT&T and Hurricane Electric.

to differ slightly in terms of *which* routes were correct. More specifically, CAIDA produces roughly 2,000 more correct routes of length 4 or longer. For UCLA, these correct paths have a length of 3 instead.

We also simulated CAIDA's topology treating siblings ASNs as normal *peer* links. This policy change *tended to be more accurate and have less spread in path length*. This suggests either the sibling relationship is not a good representation or that the policies applied to siblings are not yet well understood.

V. CONCLUSIONS AND FUTURE WORK

We have built a routing protocol simulator based on RAML that allows for easily repeatable experiments for Internet scale topologies. This simulator was evaluated using two different AS level topologies and a RAML formulation of BGP to attempt to predict what routes a real router would pick under the same circumstances. For a tier-1 ISP, we were able to exactly match around 40% of its routes. However, we believe the lack of information on how multiple prefixes behave did account for a large number of differences from the real BGP. In the future, we hope researchers will develop ways to model these prefixes in both RAML and topology files so that we can attempt to better simulate route selection for BGP and other routing protocols for the future Internet.

REFERENCES

[1] The caida as relationships dataset. http://www.caida.org/data/active/as-relationships/, November - December 2009.

[2] The ucla internet research lab internet topology collection. http://irl.cs.ucla.edu/topology/, November - December 2009.

[3] E. Bendersky. http://www.merriampark.com/ldperl.htm.

[4] M. Caesar and J. Rexford. Bgp routing policies in isp networks. In *Network, IEEE , vol.19, no.6, pp. 5- 11, Nov.-Dec. 2005*.

[5] N. Feamster and J. Rexford. Network-wide prediction of bgp routes. *IEEE/ACM Trans. Netw.*, 15(2):253–266, 2007.

[6] T. G. Griffin and J. L. Sobrinho. Metarouting. In *SIGCOMM '05: Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 1–12, New York, NY, USA, 2005. ACM.

[7] Y. He, M. Faloutsos, S. V. Krishnamurthy, and M. Chrobak. Policy-aware topologies for efficient inter-domain routing evaluations. 2008.

[8] Routeviews.org. Route views archive. http://archive.routeviews.org.

[9] J. L. Sobrinho. Network routing with path vector protocols: theory and applications. In *SIGCOMM '03: Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 49–60, New York, NY, USA, 2003. ACM.

[10] A. Stone, S. DiBenedetto, D. Massey, and M. Strout. Scalable simulation of complex network routing policies. In *ACM International Conference on Computing Frontiers*. ACM, 2010.

[11] F. Wang and L. Gao. Inferring and characterizing internet routing policies. 2003.

[12] M. Wojciechowski. Border gateway protocol modeling and simulation. Master's thesis, University of Warsaw and VU University Amsterdam, 2008.