On the Scalability of Path Exploration Using Opportunistic Path-Vector Routing

Hasan T. Karaoğlu, Murat Yüksel, and Mehmet H. Güneş
University of Nevada - Reno, Reno, NV 89557.
karaoglu@cse.unr.edu,yuksem@cse.unr.edu,mgunes@cse.unr.edu

***Abstract*—It can be argued that, BGP, de-facto inter-domain routing protocol, provides fairly stable routes. Path stability is a desired product of that limited level of deterministic performance of BGP. To attain this performance level, BGP relies on keeping up-to-date (aggregated) global information by incurring the cost of control traffic and delayed convergence. In this work, we developed an Opportunistic Path-Vector (OPVR) protocol which provides nice trade-offs between path stability, routing scalability and path quality to enable flexible inter-domain level routing services. Our approach is to redefine routing problem as a set of smaller scale problems which can be solved locally without requiring a global coordination but local communication. We also provide guidelines on how to solve these localized routing problems efficiently. Our analysis show that our method provide a good compromise between scalability and opportunity through smartly randomized (non-deterministic) choices. Our experiments with OPVRs on Internet AS-level topology show us that OPVRs can provide non-deterministic, scalable path exploration mechanisms with reasonable control traffic cost.**

I. Introduction

The de facto inter-domain routing protocol, i.e., BGP, is generally accepted to provide reasonably stable end-to-end reachability on the Internet. Assuming there are no policy fluctuations or routing pathologies, BGP can achieve some-level of deterministic performance, favoring path stability. Path stability can be defined as the increased chance of choosing (or observing) a particular set of end-to-end (e2e) paths instead of a random distribution. Previous observations on routes to popular Internet destinations confirmed persistence and prevalence of paths with lifetimes spanning over weeks and longer [1]. To be able to provide such stable routes, BGP relies on keeping up-to-date aggregate global information at the cost of reasonable control traffic and delayed convergence.

In this work, we take an alternative non-deterministic approach on routing where routers discover paths using a probabilistic decision process. We utilize trade-offs between path stability, path quality and routing scalability. We do not follow either of a purely randomized or a deterministic method. While preserving path stability as much as possible, we reduce determinism in favor of improved path diversity, path exploration, and routing scalability. By managing such trade-offs, our approach enables flexible routing services.

To examine efficiency of our approach, we analyze path exploration performance of an Opportunistic Path-Vector Routing (OPVR) protocol. OPVR seeks on-the-fly composition of high quality end-to-end paths in a scalable way using path inquiries. One of the key challenges is to establish quality paths in a distributed manner without relying on the global information. This challenge has to be met since Internet conditions may change quite rapidly. Global information for path quality may quickly become stale and update traffic for such cases would be prohibitive at the Internet scale.

Since we seek not only end-to-end reachability but also certain properties, distributed path composition problem becomes more complicated. If intermediate Internet Service Providers (ISPs) do not prefer participating in the composition of such end-to-end paths, then they simply do not forward or filter out path inquiry traffic. Hence, routing protocol has to be more resilient and graceful in cases of non-cooperating parties. Due to these challenges, OPVR acts in an opportunistic way to explore as many end-to-end paths possible while staying within a time and control traffic budget.

Inspired by methods from various research domains (e.g., wireless sensor networks, ad-hoc networks, and fluid mechanics), we propose a hybrid solution based on a combination of locality and smart-randomization. OPVR takes advantage of the Internet topology structure (e.g., tiers, path diversity, and short diameter). Routing problem is divided into individually solvable local subproblems (e.g., forwarding) while targeting a reasonable cost to integrate local solutions. Whenever local information is insufficient, OPVR triggers controlled smart-randomized forwarding mechanism to determine a path towards the destination.

In this paper, we evaluate the key trade-off between the cost of keeping local information and the cost of smartly-randomized forwarding decisions. Randomness reduces local information and related protocol state costs which in turn significantly improves routing scalability. In general, randomness in path exploration process also enhances the path diversity beyond what could be achieved using a deterministic approach. We experiment with genuine Internet topologies to quantify our path exploration performance assuming different ISP cooperation levels. We analyze OPVR performance in terms of path-stretch, cost of control traffic, path diversity, and end-to-end reachability metrics. We quantitatively show that OPVR is capable of discovering a rich-set of paths satisfying certain properties using a strictly controlled traffic budget. Finally, we demonstrate that OPVR offers graceful performance degradation at modest ISP cooperation scenarios where ISPs filter out (i.e., drop) routing update packets.

In the rest of the paper, we give a summary of previous research on routing problems in various research domains

which face similar challenges in Section II. In Section III, we explain specifics of opportunistic path exploration problem. In Section IV, we present Opportunistic Path-Vector Routing protocol. Finally, we evaluate our approach in various scenarios in Section V, and briefly summarized our work in Section VI.

II. Related Work

Forwarding proposals based on controlled flooding and random-walk have been proposed previously for specific problem domains. Such non-deterministic solutions have been often proposed as solutions to search and routing problems in dynamic and highly distributed environments like peer-to-peer (p2p) and ad-hoc wireless networks [2]. A key challenge for these environments is to find content or a route in a distributed manner without requiring the existence of a central authority or global information. However, for these solutions, to achieve a certain level of path exploration performance, significant amount of control traffic should be expected [2], [3].

Similarly, inspired by the research on Percolation Theory of Physics, there have been efforts to explain non-deterministic routing behavior with the model of spread patterns of fluids on maze like environments where there exist open and closed gates with given probability. These research efforts have focused on coverage of such a spread varying with given conditions and characteristic properties of topology [3], [4]. Inspired by these theoretical foundations, improved controlled flooding approaches have been proposed. Another similar approach, where pre-existing conditions of the networked medium (e.g., structure of topology) are utilized, Kleinberg proposed an upper-bound performance for routing problem in structured networks such as lattice or Power-law networks [5].

Many solution proposals (e.g., Compact Routing [6]) depend on exploitation of existing structure of the topology through rather strict tier-based routing. Also, many other researchers proposed solutions (NIRA, HLP [7], [8]) that depend on the tiers with less-strict routing schemes to take advantage of the diversity of the routes within tiered layers. From another perspective, many others developed protocols that purely take advantage of diversity of routes in topology to improve routing performance as a service (Multi-path Routing [9]). Also, there are several proposals which loose the strict requirements (e.g. convergence and accuracy of route conditions) on protocol performance while they still expect certain level of global coordination or information such as iRex and NIRA [7], [10] to be able to offer flexible and demanding services (e.g., QoS, Economic-incentive aware routing).

For preserving generality of the routing solution, we cannot take an approach that purely depends on tier-based routing. Partly, this is because we have to consider Economics of Internet Peering for the wired backbone and to prevent the loss of opportunity in exploring diverse set of paths offered by Internet. However, we also cannot depend on total non-deterministic structures like flooding or random-walks due to scalability, timeliness and control traffic budget. So, here in this work, we question whether or not we can find a solution in between.

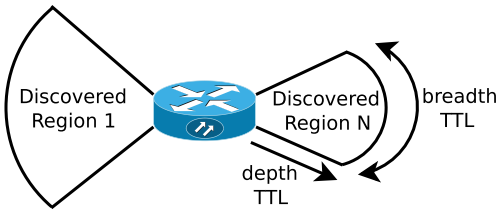

Fig. 1. Coverage areas through controlled-flooding

III. Opportunistic Path-Vector Protocols

OPVR path exploration process starts with a path query initiated by a source ISP, which describes the properties (e.g.,QoS, price and duration) of an end-to-end path whose composition is desired between a source and a destination ISP. Through forwarding these path queries, ISPs along the way either choose to participate in composing such an end-to-end path or just simply to retreat and drop (filter) the inquiry packet. The problem for intermediate ISPs is to decide whether it is technically and economically feasible to involve in such a path composition. BGP uses this type of path-vector composition process to compute end-to-end paths. But, it only considers the number of hops (i.e., ASes) a path traverses and computes the shortest-path based on a single parameter.

Each participant ISP adds itself into the path-vector list in a received inquiry packet and forwards it to its selected neighbors. Path-vector lists kept in these path inquiry packets provide loop prevention mechanism for OPVRs similarly to BGP. If path inquiry packet ever reaches destination ISP, source ISP will be informed about the e2e path with an acknowledgment packet. Finally, after reservation process, such e2e paths will be setup by stiching these forwarding tunnels (edge-to-edge ISP paths).

In our OPVR design, each ISP only has to keep connectivity information for a bounded region. This connectivity information solely consists of peering relationships of ISPs within this region and does not include any path quality information (e.g., QoS). The boundaries of concerned region can be fine-tuned by individual ISPs, too. OPVR utilizes locality-awareness to reduce non-determinism in making next hop selections in path inquiry packet forwarding. If an ISP cannot locate destination ISP in his locality, then path inquiry packet forwarding will be handled by smartly-randomized mechanisms which will be described in following section.

IV. Implementation

In this section, we will give technical details on how we challenge the problems described previously in Section III.

A. Mechanism

1) Path Exploration: To explore end-to-end paths in a distributed manner, we choose to introduce packet flooding mechanisms. Without hitting scalability problems and performance budget issues (time and control traffic), we propose a path exploration process controlled within a region bounded by breadth and depth limits as depicted in Figure 1.

We achieve this through breadth and depth time-to-live (TTL) fields placed in path inquiry packets . Through this controlled flooding process, we target to increase our chance of composing e2e paths and also exploit diversity in topology if it

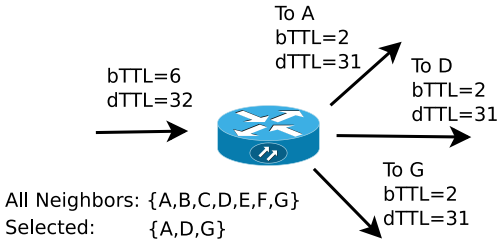

Fig. 2. Random Packet Forwarding

exists. Moreover, by providing these limits, we have theoretical bounds on the amount of routing control traffic.

2) Inquiry Packet Forwarding: Another challenge in this design is how to choose next hop(s) to forward received path inquiries among all the neighbors. As described in Figure 2, an intermediate node upon receiving the path inquiry, selects a subset of its neighbors randomly (A, D, G), updates TTL fields of original message and forwards the inquiry to this subset only (in a sense by forking and deflecting the path inquiry announcement). We will later explain how neighbor (next hop) selections can be made smarter as an improvement to OPVR.

So, breadth TTL basically limits the number of copies of original path inquiry in the system. Once bTTL field hits its minimum value of 1, then intermediate nodes can only forward the path inquiry to a single neighbor (not being able to fork it any more). In specific case of Figure 2, they may be at most 6 copies, since every time a copy is made, bTTL value is decremented. Also, as another cap to limit number of next hops that our packet can be forwarded to, there is Maximum Forward Destination (MAXFWD) parameter. So, for a specific MAXFWD value of 8, a node can only make 8 copies of the path inquiry to be sent to 8 of its neighbors even if bTTL value is greater than 8.

Since depth TTL field is decremented every time a packet forwarding made, original packet inquiry (and its copies) can go at most dTTL hops away from its initial source. Otherwise it is dropped when dTTL hits the minimum value of 0. So, coverage area of such a process is bounded by both dTTL and bTTL fields as depicted in Figure 1.

Random forwarding process prevents significant overlapping of coverage areas at deflection points. A deflection point is where a path inquiry is forked and its copies forwarded to smart-randomly selected next hops. So, there may exist N coverage areas (see Figure 1).

3) Local Information: Here, we will discuss if we make random forwarding decisions in a smarter way. First of all, if a node is the destination node described in path inquiry or a direct neighbor of this destination node, then it does not fork or forward copies of path inquiry to others, but to its destination as terminating the exploration process. So, without any effort, we have an 1-hop local information about our topology. Now, extending this local information to N-hop locality-awareness, where we know neighbors and neighbors' neighbors and so on, we can perform better smart-forwarding decisions. Unless we know that the final destination is in N-hop local coverage, we activate smart-random forwarding.

We will explain how routing problem itself will be solved within this N-hop local area in the next section. First, we want to note that as local information size kept by an ISP increases, randomness of the forwarding decision process decreases. However, this also reduces the number and diversity of the opportunities which comes with randomness of the process. However, as the randomness decreases, control traffic load decreases too. But, if we increase randomness, then the cost of storing local informantion as states will reduce. So, there is a nice compromise where protocol parameters themselves act as scalability factors balancing each out.

4) Local Routing Subproblems: If we have N-hop range locality-awareness of our neighbors, then we have a local routing sub-problem inside this locality. We propose using M-level Bloom Filters (BF) to check if a destination is in N-hop range locality. Bloom Filters are mechanisms that can verify set membership relationships for large sets with small inaccuracies and greatly reduced memory space requirements.

For a typical Tier-1 ISP with 5400 neighbors, local cache size is 4 kilobyte which is required by a BF with 4 hash functions and 5 percent false positive probability .

By using M-level BFs, it is not necessary to store all N-hop range local neighbor information. Alternatively, an ISP can verify if a destination address is in its locality. If it is indeed in ISP's locality, then it can apply (M-1) level BF to further verify which coverage area it is in specifically. By determining the value of M, an ISP can again manage the accuracy of locality-based forwarding process.

B. Improvements

Such non-deterministic process can be further improved by balancing caching and aggregation mechanisms such as classification of neighbors. Also, a node can store previously explored e2e paths in protocol cache to make more informed decisions on smart forwarding. Such methods proved themselves efficient in case of DNS address resolution caching at all levels of application (browsers), router and ISPs.

V. Evaluation

For our analysis, we used a recent Internet ISP topology map, which contains 33,508 ISPs, provided on January, 2010 by CAIDA [11]. To experiment with path exploration performance, we randomly selected 10,000 ISP pairs as source and destination and run OPVR 101 times for each.

A. Scenarios:

We analyzed multiple scenarios to evaluate scalability, stability and cost trade-offs of OPVR. We examine OPVR performance with varying *bTTL*, *dTTL* and *MAXFWD* parameters (which are described in Section IV). We also inspect routing characteristics of OPVR against increasingly risk-averse ISP policies. In our case, ISP policy reflects the risk-taking tolerance of an ISP which is shaped by the availability of resources (e.g., bandwidth) and economic utility perception of an ISP (e.g., more or less risk averse expectation for cost recovery). During our simulations, ISP policy directly determines the probability of an ISP dropping (i.e., filtering) the path query packet upon receiving it.

We have five simulation scenarios. For all scenarios including No Locality-aware **(NL)** scenario, path inquiries are forwarded in a smartly-randomized way. However, in Locality-aware **(L)** scenario, smartly-randomized forwarding mechanism is only activated when destination is not found in local cache. Locality-aware Policy cases are similar to Locality-aware **(L)** scenario, however in these scenarios path inquiries are filtered by ISPs due to their policies with increasing risk aversion levels. According to these policies, 30%, 50% and 70% of path inquiries will be dropped by **(LP30)**, **(LP50)** and **(LP70)** respectively.

B. Results:

We first analyze the path exploration success rate, i.e., whether OPVR finds any end-to-end path(s) for a given ISP source-destination pair, in Figure 3. Note that, each plot displays different set of results for varying MAXFWD values of 2, 4, 8, and 16.

In our evaluations, path exploration success rate slightly improves with increasing bTTL values from 512 to 4096. As expected, OPVR can capture more paths since more query packets are generated. Moreover, locality-aware OPVR (i.e., L) finds path(s) in at least 90% of the times for all bTTL and MAXFWD values (worst is 92.46% for bTTL=512 and MAXFWD=2; and best is 98.80% for bTTL=4096 and MAXFWD=16). An interesting result is that the increase in MAXFWD parameter value only contributes small increments to the overall path exploration success rate.

When we compare NL case with the others, we observe that locality information, i.e., cache, considerably increases the path exploration success rate. NL can only yield 50% to 60% successes whereas L always performs better than 90% success. Even with the advantage of no packet filtering, NL cannot compete with LP50 which drops 50% of inquiry packets. Locality-awareness provides improved performance even at high packet filtering rates.

When we compare NL and LP30, LP30 overperforms NL considering all bTTL and MAXFWD combinations except the only case of MAXFWD=2 and bTTL=512. After that threshold, OPVR finds end-to-end paths over 80% of the times. Hence, OPVR is resilient to non-cooperating or risk averse ISPs even at 30% levels by preserving acceptable path-exploration performance.

When we look at LP50 and LP70 cases, even with 70% of path inquiries filtered, OPVR is able to find paths with 42% or more success rate where bTTL is greater than 2048 and MAXFWD is greater than 2.

We now look at the quality and diversity of the paths found, i.e., the number of unique paths. In Figure 4, we plot the number of explored paths against varying bTTL and MAXFWD values. In contrast to path exploration performance, MAXFWD is considerably more effective than bTTL in exploring paths. Although increase in both parameters improves path diversity, MAXFWD has almost a linear positive effect proportional to increased value (e.g., path diversity doubles with transition from 2 to 4, and from 4 to 8 MAXFWD

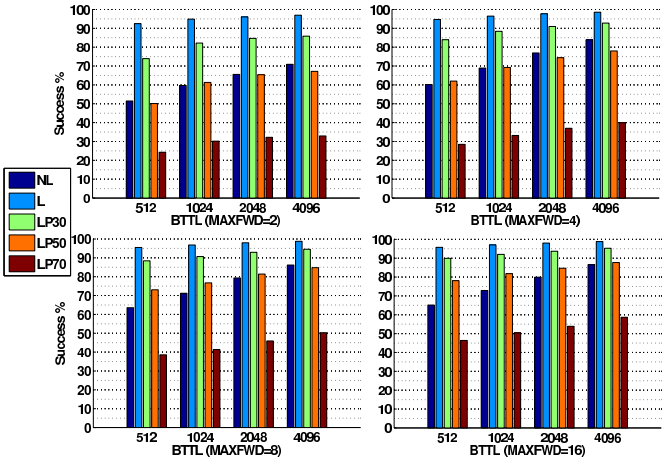

Fig. 3. Path Exploration Success Ratio (99th percentile confidence interval)

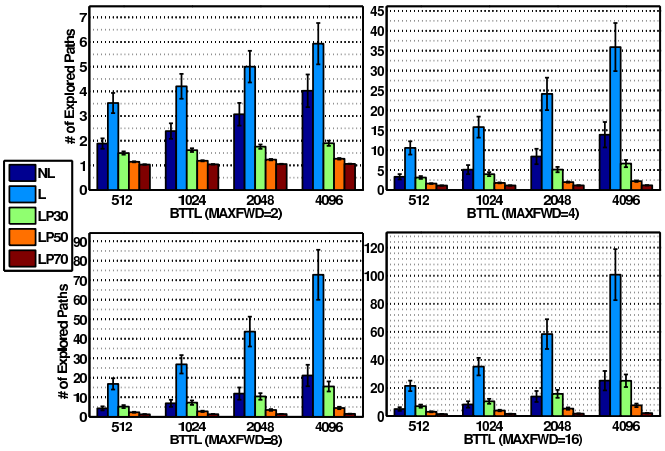

Fig. 4. Number of Explored Path (99th percentile confidence interval)

values). bTTL parameter also has a similar effect on path diversity but less significantly.

Moreover, we observed that packet filtering significantly reduces path diversity. High variation in the number of explored paths reflects the non-deterministic characteristics of OPVR and Power-law characteristic of the Internet topology [6]. We infer that the considerable effect of MAXFWD on path diversity is due to the increased randomization (or non-determinism) resulting from the relaxation on how many neighbors a node can forward the received path inquiries.

Another path quality indicator is the path stretch, i.e., the ratio of the shortest path among all explored paths to the theoretical shortest path (calculated by Dijkstra's shortest path algorithm). Note that, currently BGP performs policy-routing which is not necessarily the shortest-path routing. Hence, our comparison with the theoretical shortest path reflects a more conservative view of the path stretch.

In Figure 5, we observe that for all bTTL and MAXFWD combinations locality-aware OPVR is able to find reasonable short paths which are only 0.7 times longer than the theoretically optimal one. While smart-randomized NL can only come up with 2.0 to 2.3 times longer paths than the optimal, locality information yields better path stretch values. Moreover, path query filtering does not significantly effect the path-stretch values.

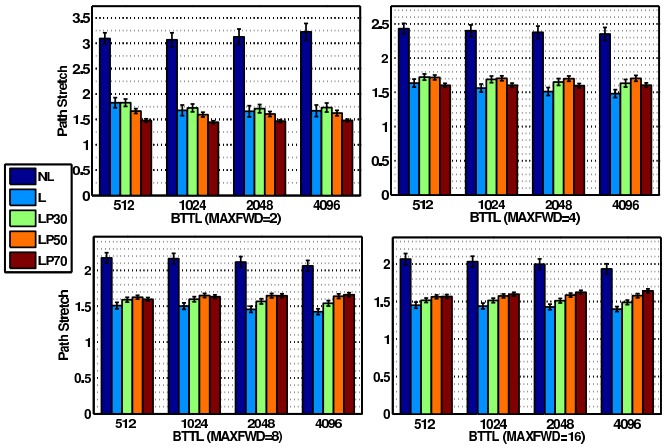

Fig. 5. Path Stretch (99th percentile confidence interval)

Finally, to quantify the control traffic load of OPVR, we analyze two metrics. First metric is the *maximum number of total message forwarding* per path inquiry. The second metric is the *maximum number of path inquiry packet copies* that can exist in the system. With bTTL, we theoretically limit the number of copies. With dTTL and bTTL parameters combined, we also theoretically limit the maximum number of packet forwarding. In practice, these values and control traffic cost are significantly less than these theoretical bounds. For example, even with the worst case of NL, OPVR generates no more than 1500 packet forwarding at bTTL=4096 as depicted in Figure 6. As parallel with our initial expectation, with locality-awareness and decreased non-determinism (through decreased MAXFWD), number of packet forwarding is greatly reduced. Moreover, with increasing packet filtering, the number of packet forwarding significantly drops (e.g. compared to NL case, it differ by two orders of magnitudes). As packet filtering increases, the total message forwarding cost decreases. Hence, we can execute more trials for the cases where no-path is found. This indicates OPVR resilience. In Figure 7, we plot the maximum number of copies of a path inquiry exist in the system. Here, these values are significantly less than what they are theoretically allowed to be. This is due to several reasons namely: (i) Internet topology characteristics and (ii) more importantly Path-Vector loop prevention mechanisms that will drop packets whenever a loop is detected.

VI. Summary and Further Issues

In this work, we analyze path exploration performance of Opportunistic Path-Vector protocols on the Internet. By using local topology information and smartly-randomized routing decisions, we developed an opportunistic protocol which can perform successfully even at the scale of Internet. Our experiments show that well-balanced tradeoffs between path stability, path diversity and scalability enable OPVRs to attain over 90% path exploration success with both theoretical and practical hard bounds on control traffic. OPVRs also reflect graceful performance degradation properties, even resilient to conditions where 30-50% of control traffic is being filtered or lost.

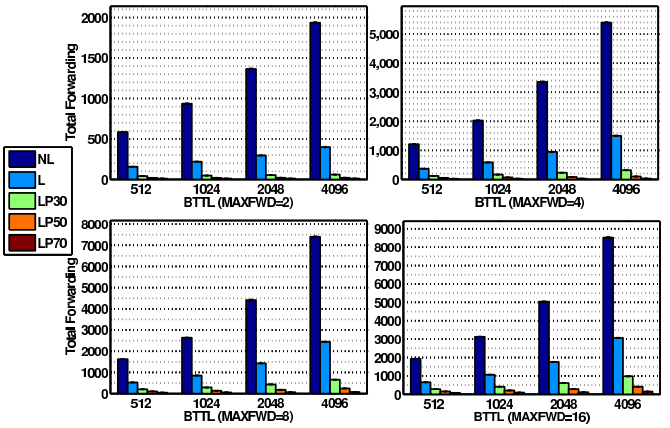

Fig. 6. Maximum Number of Messaging in the system (99th percentile confidence interval)

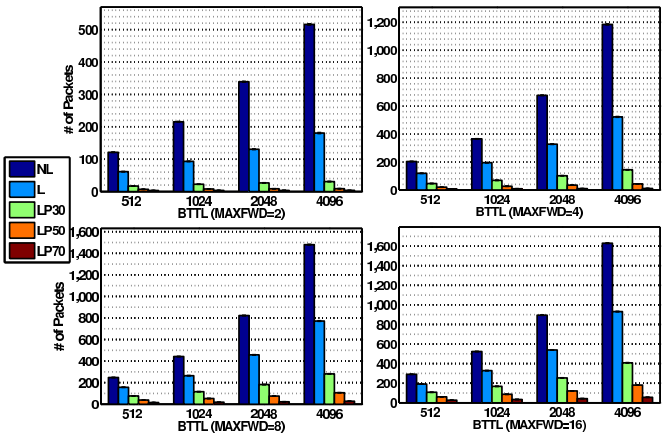

Fig. 7. Maximum Number of Packets in the system (99th percentile confidence interval)

VII. Acknowledgment

This project is supported in part by National Science Foundation awards 0721600, 0721609, and 0831957.

References

[1] J. Rexford, J. Wang, Z. Xiao, and Y. Zhang, "Bgp routing stability of popular destinations," in *Proc. of ACM IMW*, 2002, pp. 197–202.

[2] A. V. Kini, V. Veeraraghavan, N. Singhal, and S. Weber, "Smartgossip: an improved randomized broadcast protocol for sensor networks," in *IPSN*, 2006, pp. 210–217.

[3] V. Raman and I. Gupta, "Performance tradeoffs among percolation-based broadcast protocols in wireless sensor networks," in *Proc. of ICDCS*, 2009, pp. 158–165.

[4] A. Jiang and J. Bruck, "Monotone percolation and the topology control of wireless networks," in *INFOCOM*, 2005, pp. 327–338.

[5] J. Kleinberg, "Navigation in a small world," *Nature*, p. 845, 2000.

[6] D. Krioukov and K. Fall, "Compact routing on internet-like graphs," in *In Proc. IEEE INFOCOM*, 2004, pp. 209–219.

[7] X. Yang, D. Clark, and A. Berger, "NIRA: A new inter-domain routing architecture," *IEEE Trans. on Network.*, vol. 15, pp. 775–788, 2007.

[8] L. Subramanian, M. Caesar, C. T. Ee, M. Handley, M. Mao, S. Shenker, and I. Stoica, "HLP: a next generation inter-domain routing protocol," in *Proc. of SIGCOMM*, 2005, pp. 13–24.

[9] W. Xu and J. Rexford, "MIRO: multi-path interdomain routing," *ACM CCR*, vol. 36, no. 4, pp. 171–182, 2006.

[10] A. D. Yahaya and T. Suda, "iREX MPO: A multi-path option for the irex inter-domain qos policy architecture," in *ICC*, 2008, pp. 5815–5822.

[11] "The CAIDA AS Relationships Dataset," January 2010, "http://www.caida.org/data/active/as-relationships/".