April 2006

Light-Weight Overlay Path Selection in a Peer-to-Peer Environment

Teng Fei
University of Massachusetts

Shu Tao
IBM T.J. Watson Research Center

Lixin Gao
University of Massachusetts

Roch A. Guérin
University of Pennsylvania, guerin@acm.org

Zhi-Li Zhang
University of Minnesota

Light-Weight Overlay Path Selection in a Peer-to-Peer Environment

Abstract
Large-scale peer-to-peer systems span a wide range of Internet locations. Such diversity can be leveraged to build overlay "detours" to circumvent periods of poor performance on the default path. However, identifying which peers are "good" relay choices in support of such detours is challenging, if one is to avoid incurring an overhead that grows with the size of the peer-to-peer system. This paper proposes and investigates the Earliest Branching Rule (EBR) to perform such a selection. EBR builds on the Earliest Diverging Rule (EDR) that selects relay nodes whose AS path diverges from the default path at the earliest possible point, but calls for monitoring a much smaller number of paths. As a result, it has a much lower overhead. The paper explores the performance and overhead of EBR, and compares them to that of EDR. The results demonstrate that EBR succeeds in selecting good relay nodes with minimum control overhead. Hence, providing a practical solution for dynamically building good overlays in large peer-to-peer systems.

Keywords
peer-to-peer, networks, overlay

Light-Weight Overlay Path Selection in a Peer-to-Peer Environment

Teng Fei[†] Shu Tao[‡] Lixin Gao[†] Roch Guérin[*] Zhi-li Zhang[§]
[†] Dept. Elec. & Comput. Eng., U. Massachusetts
{tfei, lgao}@ecs.umass.edu
[‡] IBM T. J. Watson Research Center
shutao@us.ibm.com
[*] Dept. Elec. & Sys. Eng., U. Pennsylvania
guerin@seas.upenn.edu
[§]Dept. Comput. Sciences & Eng., U. Minnesota
zhzhang@cs.umn.edu

***Abstract*— Large-scale peer-to-peer systems span a wide range of Internet locations. Such diversity can be leveraged to build overlay "detours" to circumvent periods of poor performance on the default path. However, identifying which peers are "good" relay choices in support of such detours is challenging, if one is to avoid incurring an overhead that grows with the size of the peer-to-peer system. This paper proposes and investigates the Earliest Branching Rule (EBR) to perform such a selection. EBR builds on the Earliest Diverging Rule (EDR) that selects relay nodes whose AS path diverges from the default path at the earliest possible point, but calls for monitoring a much smaller number of paths. As a result, it has a much lower overhead. The paper explores the performance and overhead of EBR, and compares them to that of EDR. The results demonstrate that EBR succeeds in selecting good relay nodes with minimum control overhead. Hence, providing a practical solution for dynamically building good overlays in large peer-to-peer systems.**

I. Introduction

Access to multiple, diverse (non-overlapping) paths can help applications and end-systems improve end-to-end performance by allowing them to temporarily bypass network segments that are experiencing poor performance. The highly inter-connected nature of the Internet topology offers a wealth of connectivity options that could support such an approach. Tapping into the opportunities this presents is, however, currently difficult. Specifically, most traffic typically follows a single ("default") path, because of constraints imposed by IP routing. There is, therefore, an incentive to develop solutions capable of overcoming this limitation. This has led to a number of recent proposals [1], [2], [3], [4], [5], [6] that allow end-systems to directly exploit the available Internet path diversity through multi-homing and overlay networks. These have been shown [3], [4], [5] to translate into meaningful performance improvements, especially for QoS-sensitive real-time applications, such as VoIP and video streaming.

The emergence of large-scale peer-to-peer (P2P) systems (e.g., Skype) offers even more opportunities to exploit path diversity. These systems typically involve a significant number of peer nodes (end hosts) distributed over broad geographic areas. Collaboration among peers may overcome certain limitations or restrictions imposed by underlying networks. One node (source) may select another peer out of all available peers as relay to forward traffic to the destination instead of sending the traffic directly using the default Internet path. Such overlay paths may provide "uncorrelated" performance with that of the default path so that they are unlikely to experience degradation at the same time. Realizing the potential benefits that large P2P systems offer is, however, challenging in that the *existence* of good alternate paths does not mean that their *identification* is easy. Designing a solution to this problem calls for addressing the following challenges: i) How to select a *candidate* set of *good paths* from a potentially very large[1] set of overlay paths; ii) Can it be done in a *scalable* manner, i.e., without incurring too much overhead? Clearly, the trade-off between performance and overhead in selecting good candidate paths must be carefully investigated.

In [7] we focused on the first question, namely, how to select a candidate set of "good" overlay paths. Through extensive measurements and route analysis, we validated that the performances – in terms of loss and delay variations seen by end hosts – of the default path and relay paths that diverge early, at the AS-level, from it, were indeed relatively uncorrelated. This insight motivated us to propose the *earliest divergence rule* (EDR) for selecting candidate overlay paths. However, as we discuss further in Section III-A, the overhead of EDR remains substantial because it requires constant maintenance of path state from the source to all relay nodes.

Overcoming this limitation is the main motivation for this paper, which builds on the results of [7] and the intuition gained from EDR. Specifically, the paper introduces and evaluates a new approach – the *earliest branching rule* (EBR) that offers performance comparable to that of EDR, but at a fraction of the cost in terms of overhead. Rather than comparing for each destination d the default AS path from s to d with the AS paths from s to all possible relay nodes in order to identify the proper subset $\mathcal{O}$ to choose from, EBR identifies *a priori* a small (as small as just two) set of relay nodes that it uses to select, for any destination, an overlay path that exhibits a similar disjointness with the default path as that produced

[1]If *one-hop* overlay paths are used, the size of this set is $O(N)$, where N is the number of peers; and it is $O(N^2)$ if two-hop overlay paths are used!

by EDR. Because EBR only needs to monitor a much smaller set of AS paths, it incurs a much smaller overhead than EDR. There is, however, a cost paid for reducing the set of choices available to EBR, since in general access to the full AS path tree to all possible relay nodes offers a richer set of choices that can be refined in a destination specific manner (as in EDR) to deliver better performance. This trade-off between performance and (monitoring) overheads is formalized in Section IV, where we establish using various Internet BGP datasets that the performance advantage of EDR over EBR is relatively small, while the reduction in overhead that EBR affords over EDR can be very substantial. As a result, EBR embodies a more practical trade-off between performance and feasibility than EDR, at least in large P2P systems.

II. Related work

The feasibility of using overlay path to obtain path diversity has been demonstrated by several recent studies, e.g. [2] [8] [3]. In order to find overlay paths with uncorrelated performance, several schemes have also been proposed. For instance, [9] introduced a routing underlay dedicated to inferring AS topology and constructing AS disjoint overlay paths. In [10], Gummadi *et al*. studied the use of randomly selected overlay paths to bypass the performance degradations on the default path.

Although these approaches have proved effective in various settings, they are not specifically tailored for overlay path selection in large P2P systems, where finding a solution that achieves a reasonable tradeoff between performance and scalability is challenging. This is the focus of this paper. In particular, path selection mechanisms used in such systems should not incur excessive control overhead, so that the system can maintain its scalability. In [7], we proposed the earliest-divergence rule (EDR) for selecting alternate overlay paths. Compared with a naïve disjoint path selection scheme, EDR effectively reduces the path information maintained by each node. However, it requires monitoring the AS paths from source to all possible relay nodes, which still causes non-negligible control overhead. The method introduced in this paper, the earliest branching point rule (EBR), requires only a fraction of the overhead, while achieving performance close to that of EDR, which makes it more suitable for deployment in large P2P systems.

III. Path Selection Approaches

Our goal is to enable, for any source and destination, the selection from possibly thousands of choices (peers), of *one* "good" overlay path to be used as a (standby) alternate during periods of poor performance on the default path. A brute-force approach based on full information about all potential overlay paths is clearly not feasible, while a naïve random selection only delivers "average" path diversity, which as shown in [7] and illustrated later in Figure 2(b), often results in poor performance, even if it is arguably simple and scalable. In this section, we first present EDR that was introduced in [7] as a possible solution, and review its motivations, advantages and disadvantages. We then propose EBR that is designed based on the insight gained from EDR in an attempt to retain its benefits while lowering complexity.

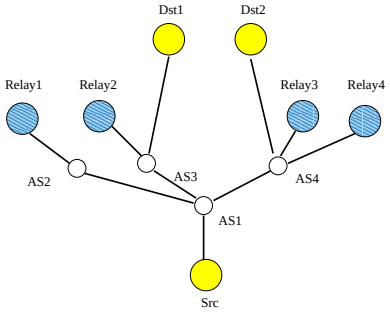


Fig. 1. Illustration of Earliest Divergence Rule (EDR) and Earliest Branching Rule (EBR)

A. Earliest Divergence Rule (EDR)

The relay selection process of EDR involves two steps. i) For a pair of source and destination nodes, EDR selects, among all possible relay nodes, those whose AS path from the source diverges from the default path to the destination at the earliest possible AS. ii) From the preselected subset of overlay paths, EDR then chooses *one* either randomly or based on additional criteria (e.g., the distance from the source node, or the round trip delay). Fig. 1 illustrates the configuration involving four possible choices of relay nodes. When the destination is Dst1, EDR selects Relay1, Relay3 and Relay4 as candidate relay nodes, while it selects Relay1 and Relay2 when the destination is Dst2, to maximize path disjointness.

In EDR, the source knows the paths towards all relay nodes and makes relay selections based on this information. The effectiveness of EDR is predicated on two assumptions. 1), by choosing overlay paths that share fewer AS, and therefore hopefully fewer physical links, with the default path, correlation of performance between the paths should be as minimal as can be. 2), paths that diverge early from the default path will also tend to converge back on it late, minimizing their total overlap. In [7] we verified the validity of these assumptions via extensive topology and measurement-based analysis.

Although EDR does not require inspecting all possible full (from source to relay, and then from relay to destination) overlay paths, it stills requires knowing the AS paths from the source to *all* relay nodes. When the number of possible relay nodes to choose from becomes large, maintaining and processing all the required path information can be expensive. The goal of EBR, which we describe next, is to retain the benefits of EDR while incurring a lower overhead.

B. Earliest Branching Rule (EBR)

Like EDR, EBR also relies on AS-level path information to select relays. The difference is that, while EDR chooses a specific subset of relay nodes for each destination, EBR identifies *a priori* k relay nodes (k can be as small as two), from among which it limits its choice across all possible destinations. Specifically, EBR first constructs the AS path tree

rooted at the source and extending towards all candidate relay nodes, and then uses it to identify the *earliest branching point* (EBP) in the tree. EBR then selects k relay nodes, possibly spanning k different branches (if they exist) in the AS path tree at the EBP. If there are fewer than k branches at EBP, EBR selects only enough relay nodes to cover all branches. The smallest value of k is obviously 2.

Note that the EBP identifies the earliest divergence point between the default path to *any* destination, and an overlay path through a selected relay node. Furthermore, once a destination is given, applying EDR to the preselected set of k nodes, we will always be able to identify a relay node (from these k nodes) that yields an overlay path that diverges at the EBP from the default path to this destination. This is because, by selecting k nodes covering different branches at the EBP, EBR guarantees that no matter where the destination is located, there are at least $(k-1)$ relay candidates that satisfy the earliest divergence criterion. For example, in Fig. 1, if $k = 2$, any two nodes among Relay1, Relay2, Relay3 and Relay4, can be pre-selected by EBR, except for the combination of Relay3 and Relay4. If, say, Relay 1 and Relay 2 are selected and the destination is Dst1, then Relay1 yields an overlay path that diverges from the default path to Dst1 at the EBP (AS1). If the destination is Dst2, then both Relay1 and Relay2 are valid selections. Relay3 and Relay4 cannot be pre-selected together by EBR, as the earliest divergence point in their respective paths is not at the EBP.

The control overhead of EBR in identifying and maintaining the AS path information involves techniques similar to those used by EDR, which may use either control plane or data plane information. For example, it is possible to obtain AS path information by monitoring BGP routing updates for those prefixes that cover the relay nodes over time. The problem of this approach is that many networks in which the source nodes locate do not have complete routing information for the global Internet, thus may limit the source node to monitor AS paths to some of the relay nodes. Alternatively, as described in [7], the source node can rely on `traceroute` to obtain the IP-level path information for relay and destination nodes alike. The resulting IP paths can then be mapped into AS paths. In reality, performing `traceroute` probing at the data plane is a more practical approach.

Therefore, as discussed next in Section IV, the main overhead for both EBR and EDR is that the `traceroute` based probing of paths that needs to performed regularly to detect changes. The smaller number of paths (nodes) on which EBR relies is the primary reason for its lower overhead.

IV. Performance and Overhead Evaluation

In this section, we investigate the trade-off between performance and overhead achieved by EBR and EDR, respectively. We first define the metrics used to measure performance and overhead, and introduce the approach and experimental setting we rely on to perform our comparison. The results of this comparison are discussed in Sections IV-D to IV-F, which demonstrate the clear advantages that EBR affords.

A. Performance Metrics

Both EBR and EDR rely on AS level path information to select relay nodes. As shown in [7], the performance of different selection schemes can be measured by the overlap, or number of common ASes, between the default path and the selected overlay path. Specifically, let $A(t)$ and $B(t)$ be two AS paths with the same source and destination ASes (at time t), such that $A(t) = a_1, a_2, ..., a_m$ and $B(t) = b_1, b_2, ..., b_n$, where a_i and b_j are the ASes along the two paths with $a_1 = b_1$ and $a_m = b_n$. Define $a_i \otimes b_j$ =1 if $a_i = b_j$ or 0 otherwise. Their overlap at time t is then defined as

$$V_{[A,B]}(t) = A(t) \otimes B(t) - 2 = \sum_i \sum_j a_i \otimes b_j - 2. \quad (1)$$

Note that $V_{[A,B]}(t) = 0$ if the two paths have no overlap (except for the source and destination ASes). Since AS paths (thus their overlap) change over time, their overlap can be computed as an average over time.

When evaluating the performance of a relay selection rule such as EBR or EDR, it is possible for the rule to identify more than one choice[2]. In such cases, performance is computed as the path overlap averaged over all choices (and time). Specifically, given a default path D and a set of candidate overlay paths $\mathcal{O}(t) = R_1, R_2, ..., R_L$ produced by a selection rule at time t, we define its performance at time t as $\mathcal{P}(t, D) = \frac{1}{L}\sum_{k=1}^{L} V_{[D,R_k]}(t)$. A time average is then computed over the time interval $[t_1, t_2]$ during which performance is being monitored

$$\mathcal{P}_{[t_1,t_2]}(D) = \frac{1}{t_2 - t_1}\int_{t=t_1}^{t_2} \mathcal{P}(t, D). \quad (2)$$

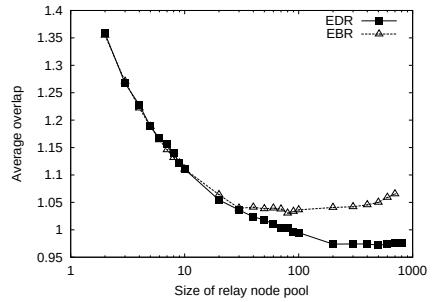
B. Overhead Metrics

As discussed earlier, the overhead of both EDR and EBR comes mainly from the periodic `traceroute` probing that the source node needs to perform to maintain accurate path information for the candidate relay nodes. Overhead is, therefore, measured as the total number of probes sent per hour by the source.

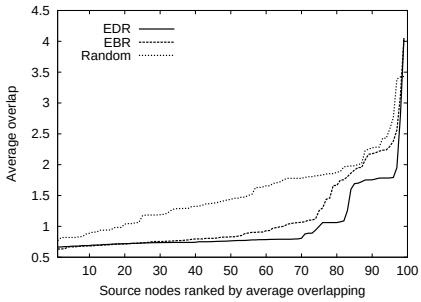
For EDR, given a source node s and a set of relay nodes R of size N, the overhead of EDR is given by the product of the `traceroute` probing rate, r, and the number of monitored paths, N, namely, $O_{EDR} = rN$. The rate r is chosen based on the frequency of path changes to the N relay nodes.

The monitoring required by EBR depends on the two distinct types of events that affect the information on which EBR bases its decisions, namely, the EBP itself changes, or a change occurs on one or more of the k selected paths so that they no longer diverge at the EBP. Catching the first events calls for monitoring *all* the AS paths, while the second only require monitoring the AS paths to the k pre-selected relay nodes. Besides differences in the number of paths that need to be monitored, the rates at which these two measurements need to be performed can also be quite different. We denote as r_1 and r_2 the probing rates for detecting the first and second types of events, respectively. The overhead of EDR is then $O_{EBR} =$

[2]EDR typically returns multiple possible choices, and so may EBR even in the base case of $k = 2$.



(a) Varying relay node pool size



(b) Varying source nodes

Fig. 2. Performance comparison between EBR and EDR

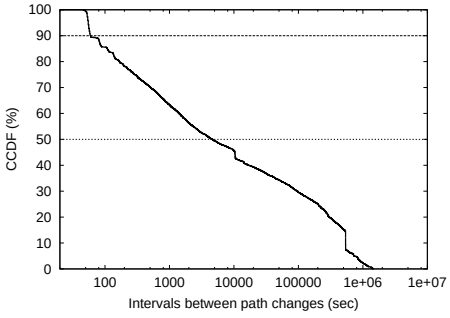


Fig. 3. The distribution of AS path change intervals

$r_1(N-k)+r_2k$. Intuitively, r_1 is likely to be significantly lower than r in EDR, as most individual path changes do not affect the location of the EBP, i.e., they occur further up in the tree and/or the EBP involves multiple branches that are unlikely to all change simultaneously. Conversely, because r_2 depends mainly on the frequency of path changes to the k pre-selected relay nodes, it should be of the same order as, and often lower than (not all path changes affect the location of the branching point between two paths), the rate r of EDR. This intuition is confirmed by our evaluation based on actual BGP AS path datasets, as described below.

C. Datasets

In the following sections, we emulate an actual P2P system to carry out a realistic evaluation and comparison of EDR and EBR. Here we base our analysis on AS path information extracted from global BGP routing tables instead of using direct `traceroute` measurements because we are trying to evaluate the effectiveness of EDR and EBR in more general Internet settings. Even though traceroute probings are usually sufficient to monitor the state of the AS paths from the source node's perspective, it would restrict our analysis to a few sources. Instead, using BGP data we are able to rely on much more comprehensive AS level routing information that is more representative of the global Internet. For example, by peering with different BGP routers, the RouteView [11] and RIPE [12] servers provide information on active prefixes in the Internet and AS path dynamics observed from dozens of vantage points. In some cases, the BGP data collected from various vantage points are not sufficient to provide complete AS path information. In those instances, techniques of [13] and [14] can help infer the AS path from a source to a destination. In our analysis, we rely on the heuristic of [14] to infer AS paths when they are not directly observable from BGP data.

In selecting the location of nodes (source, relays, or destinations), their IP addresses are produced from actively routed IP addresses[3] assuming a uniform distribution. When comparing the performance of EDR and EBR, we use a collection of BGP tables archived by Oregon/RIPE on June 10, 2005. Conversely, the comparison of their relative overhead is carried out using a collection of BGP tables archived by Oregon on October 15, 2005, and for which we record all BGP updates until October 31, 2005. This allows us to track AS path changes (and the corresponding ASes) observed by Oregon's BGP neighbors to all destinations in the Internet over these 15 days. Note that our occasional reliance on AS path inference algorithms will not impact our comparison of EDR and EBR, as both are equally affected.

[3]Actively routed IP addresses are addresses covered by the prefixes observed from the global BGP routing tables.

D. Performance Comparison

Using the first BGP data set, we randomly generate 100, 100, and 800 IP addresses as sources, destinations, and relay nodes respectively. The algorithm of [14] is then used to produce complete AS paths for 9702 source-destination pairs. For these, we compare the performance $\mathcal{P}$ of EBR and EDR as a function of the size of the relay node pool.

Figure 2(a) shows the average performance of EDR and EBR ($k=2$) across all sources-destination pairs, as the relay node pool size varies from 2 to 800. We repeat the experiments 30 times for each value (except for size 800), by randomly selecting relay nodes from among the 800 possible choices. We find that when the relay node pool is small, the performance of EBR is similar to that of EDR. As the relay pool size increases, EDR gradually outperforms EBR. This is because although the two relay nodes of EBR ensure at least one earliest diverging overlay path, this rather limited choice is not equally good across source-destination pairs and results in relatively variable performance. In contrast, when the number of relay nodes increases, the performance of EDR improves as the greater number of choices helps limit the impact of occasional poor choices. Figure 2(b), compares EBR and EDR for all 100 sources each averaged across all destinations using all 800 relay nodes. For reference purposes, we also show the performance of a simple *random* selection rule. It can be seen that while EDR outperforms EBR (the number of relay nodes is large), the two are reasonably close across all sources. In addition, both schemes do significantly better than the random selection rule.

E. Overhead Comparison

With both EDR and EBR, a source needs to perform periodic `traceroute` to maintain accurate AS path information.

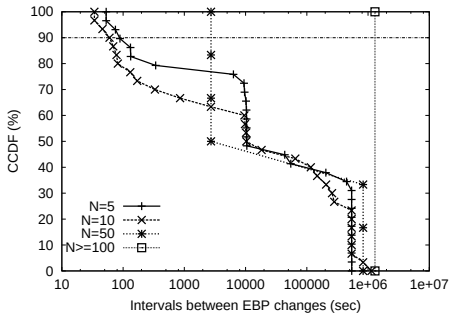


Fig. 4. Distribution of EBP change intervals

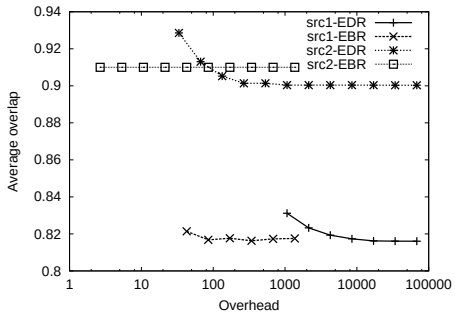


Fig. 5. Performance comparison between EBR and EDR

The overhead of both EDR and EBR is affected by their probing rates, while performance depends on the frequency of path changes and the rule's ability to detect them given its probing rate. A low probing rate can prevent EDR and EBR from adapting quickly to path changes, while a high probing rate incurs a large overhead. In this subsection, we compare the performance of EBR and EDR as a function of their probing rate (overhead).

Because of its importance on both performance and overhead, we first investigate the frequency of AS path changes. We choose five neighbors of the Oregon Route View server as sources. The five nodes are selected such that each node is in a different AS, and the ASes exhibit a reasonable degree variety. For each source, we randomly select 1765 destinations and monitor AS path changes from the sources for 15 days using BGP updates collected from the Oregon server. In order to eliminate the impact of transient route changes, we investigate the inter-arrival time distribution of BGP updates. A heuristic threshold was chosen to aggregate BGP updates so that transient path changes are filtered. We observed that for updates toward all destination prefixes, about 10% of them had inter-arrival times of 30 seconds or less, while the remaining 90% had noticeably larger intervals. We use a heuristic threshold of 45 seconds to aggregate BGP updates so that updates received within 45 seconds of each other are considered transient routing behavior and the resulting path changes are not considered in our analysis. A threshold of 90 seconds shows very similar result. Based on this observation, Figure 3 shows the distribution of the time between successive path changes after this aggregation. We observe that about 50% of the intervals larger than 4,600 seconds ($\approx$ 1.3 hour). This indicates that in order to capture 50% of AS path changes, the probing rate needs to be one probe per hour or higher. Improving this accuracy to, say, capture 90% of path changes, calls for a much higher probing rate, i.e., one probe every 60 secs or less based on Figure 3.

Based on the above observations on the AS path changes, we conclude that EDR's overhead, measured as rN, can be quite high when N is large, even if r is in the order of only 1 per minute. As far as EBR is concerned, its overhead has two components: r_2 used to monitor changes to the k selected paths, and r_1 for monitoring the remaining $N - k$ paths for EBP changes. The rate r_2 is a function of the overall frequency of AS path changes, and can, therefore, be chosen approximately equal to r of EDR, i.e., $r_2 \approx r$. Hence the corresponding overhead is 2 probes per minutes, when $k = 2$ and $r = 1$ probe/min. The rate r_1 depends on the frequency of EBP changes, which we investigate next using the same five neighbors of the Oregon Route View server as sources, and for each randomly select 10 sets of prefix groups of various sizes 5, 10, 50, 100, 300, 500 and 800, to emulate different relay pool sizes N. We compute the EBP change intervals for each source-relay combination and aggregate the corresponding $5 \times 10 = 50$ data sets for each relay pool size. The resulting distributions are shown in Figure 4, which indicate that EBPs are much more stable than AS paths. In addition, the frequency of EBP changes decreases as the relay node pool size N increases, which further helps scalability. This observation provides some guidelines on how to choose r_1 as a function of the relay node pool size. When the relay pool sizes are relatively small, for example for $N = 5$ or 10, a probing rate of about once every 60 seconds is required to capture 90% of EBP changes according to our data. When the relay pool size grows large, the frequency of EBP changes is much smaller. For example, when $N = 50$, a probing rate of once every 2,000 seconds is sufficient to capture 90% of the EBP changes. When N exceeds 100, we do not observe any EBP changes during the entire 15-day for all five sources, which suggests that even a very low r_1 would be acceptable. This is because EBP is determined by the AS paths to all N relay nodes. When N is small, individual path changes are more likely to affect EBP, but the frequency and the absolute number of EBP changes will not exceed those of AS path changes. As N grows big, the EBP becomes much less sensitive to path changes. We believe that this conclusion holds true in general, and that the stability of EBP increases with the relay node pool size, even if the actual distribution of times between changes varies across scenarios. The benefit of this behavior is that it improves the scalability of this second component of EBR's control overhead, i.e., as N grows, the required probing rate r_1 decreases, so that the probing overhead Nr_1 remains small an even decrease, e.g., from 10 probes/min when $N = 10$ to about 1.5 probes/min when $N = 50$.

TABLE I

RANGE OF PATH PROBING RATES FOR TWO SOURCE NODES

Source	EDR r (1/hour)	EBR r_1 (1/hour)	EBR r_2 (1/hour)
src1	10.67 - 682.67	10.67	10.67 - 682.67
src2	0.67 - 682.67	0.67	0.67 - 682.67

F. Trade-off between Overhead and Performance

Finally, we investigate the trade-off between overhead and performance for both EDR and EBR. In order to account for variability between different sources, we choose two of the Oregon neighbors as source nodes. We randomly select 100 IP addresses as relay nodes, and 100 IP addresses as destination nodes. The performance is averaged across all destination nodes. We again choose $k = 2$ for EBR. The path changes are monitored over periods of 6 hours using different "probing rates", as listed in Table I.

For EBR, the measurements were repeated 20 times for different choices of the $k = 2$ relays. The results are shown in Figure 5. We observe that the performance of EDR improves as its overhead increases, until it reaches about 4,267 for $src2$ and 133 for $src1$, which translates into probing rates r of about 42.67 and 1.33 probes per hour respectively. The difference is due to the fact that paths from $src1$ experience much more frequent changes than those from $src2$. For EBR, performance improvements as the overhead increases are much less significant, even if, as expected, a similar trend can be observed for paths where changes are more frequent. More importantly, the comparison confirms that EBR affords a much better trade-off between overhead and performance, even if EDR is in general capable of outperforming EBR. However, this only happens at high probing rates that correspond to an unacceptable overhead. Note that selecting an appropriate probing rate at a given source does require some knowledge of the frequency of AS path changes as observed from this source. This knowledge can be acquired by observing path changes over time. The details of such an approach are left for future study.

V. CONCLUSIONS AND FUTURE WORK

This paper introduces and evaluates a new rule, the earliest branching rule (EBR), to construct one-hop overlay paths in large P2P systems. EBR's ability to identify alternate standby paths that are as disjoint as possible with the default path, and in realizing an effective trade-off between performance and overhead were demonstrated using extensive measurements that spanned a broad range of sources, destinations, and relay nodes. As part of our future work, we plan to also evaluate the performance of EBR with respect to the QoS requirements of applications, as well as explicitly take attributes of P2P systems into consideration, e.g., the heterogeneity of peer nodes in terms of their connectivity, resource, and their willingness to share, as well as the dynamics of node join/leave. A refined EBR that accounts for the specific characteristics of P2P systems will clearly be a more practical real-world solution.

ACKNOWLEDGMENTS

This work is supported by the National Science Foundation under the grants CNS-0435444, CNS-0085848, ITR-0085824 and ITR-0085930.

REFERENCES

[1] S. Savage, T. Anderson, A. Aggarwal, D. Becker, N. Cardwell, A. Collins, E. Hoffman, J. Snell, A. Vahdat, G. Voelker, and J. Zahorjan, "Detour: a case for informed Internet routing and transport," *IEEE Micro*, vol. 19, no. 1, pp. 50–59, January 1999.

[2] D. G. Andersen, H. Balakrishnan, M. F. Kaashoek, and R. Morris, "Resilient overlay networks," in *SOSP*, Banff, Canada, October 2001.

[3] S. Tao, K. Xu, Y. Xu, T. Fei, L. Gao, R. Guérin, J. Kurose, D. Towsley, and Z.-L. Zhang, "Exploring the performance benefits of end-to-end path switching," in *IEEE ICNP*, Berlin, Germany, October 2004.

[4] S. Tao, K. Xu, A. Estepa, T. Fei, L. Gao, R. Guérin, J. Kurose, D. Towsley, and Z.-L. Zhang, "Improving VoIP quality through path switching," in *IEEE INFOCOM*, Miami, FL, March 2005.

[5] S. Tao and R. Guérin, "Application-specific path switching: A case study for streaming video," in *ACM Multimedia*, New York, NY, October 2004.

[6] J. G. Apostolopoulos, T. Wang, W.-T. Tan, and S. Wee., "On multiple description streaming with content delivery networks," in *IEEE INFOCOM*, New York, NY, June 2002.

[7] T. Fei, S. Tao, L. Gao, and R. Guérin, "How to select a good alternate path in large peer-to-peer systems?" in *IEEE INFOCOM*, Barcelona, Spain, April 2006.

[8] A. Akella, J. Pang, S. Seshan, and A. Shaikh, "A comparison of overlay routing and multihoming route control," in *ACM SIGCOMM*, Portland, OR, August 2004.

[9] A. Nakao, L. Peterson, and A. Bavier, "A routing underlay for overlay networks," in *ACM SIGCOMM*, Karlsruhe, Germany, August 2003.

[10] K. P. Gummadi, H. V. Madhyastha, S. D. Gribble, H. M. Levy, and D. Wetherall, "Improving the reliability of Internet paths with one-hop source routing," in *6th Usenix/ACM Symposium on Operating Systems Design and Implementation (OSDI)*, San Francisco, CA, December 2004.

[11] (2005, November) RouteViews project. [Online]. Available: http://www.routeviews.org/

[12] (2005, November) RIPE network information center. [Online]. Available: http://www.ripe.org/

[13] Z. M. Mao, J. Rexford, J. Wang, , and R. Katz, "Towards an accurate AS-level traceroute tool," in *ACM SIGCOMM*, September 2003.

[14] J. Qiu and L. Gao, "AS path inference by exploiting known AS paths," Department of Electrical and Computer Engineering, University of Massachusetts, Amherst, MA, Tech. Rep. TR-05-CSE-04, 2005.