# SYNTHETIC COORDINATES FOR DISJOINT MULTIPATH ROUTING OVER THE INTERNET

Andrei Agapi, Thilo Kielmann, Henri E. Bal
*Dept. of Computer Science, Vrije Universiteit*
*Amsterdam, The Netherlands*
aagapi@few.vu.nl, kielmann@cs.vu.nl, bal@cs.vu.nl

**Abstract** We address the problem of routing packets on multiple, router-disjoint, paths in the Internet using large-scale overlay networks. Multipath routing can improve Internet QoS, by routing around congestions. This can benefit interactive and other real-time applications.

One of the main problems with practically achieving router-disjoint multipath routing is the scalability limitation on the number of participating nodes in such an overlay network, caused by the large number of (expensive) topology probes required to discover relay nodes that provide high router-level path disjointness. To address this problem, we propose a novel, synthetic coordinates-based approach.

We evaluate our method against alternative strategies for finding router-level disjoint alternative paths. Additionally, we empirically evaluate the distribution of path diversity in the Internet.

**Keywords:** path disjointness, path similarity, Quality of Service, overlay networks

## 1. Introduction

Many Internet-based applications suffer from the lack of proper quality-of-service (QoS) provisioning. Examples are multimedia communication (telephony, video streaming), interactive systems (tele conferencing, games), as well as distributed scientific experiments, like the LOFAR distributed radio telescope.

It has been proposed [1, 8, 10]to improve the achievable QoS by using overlay networks that provide alternative paths, designed to circumvent performance bottlenecks within the Internet. Such overlay networks use (application-level) gateways to relay data around bottlenecks, using paths that are *disjoint* from the default path given by Internet routing. The idea is that alternative paths through relay gateways should

*avoid* using as many routers from the default Internet path as possible, as to minimize correlation between congestion events on default and alternative paths. One of the main problems in practically applying this solution, generally not addressed by these systems, is scale: for large numbers of hosts within an overlay network and in lack of complete a priori knowledge of Internet topology underlying the overlay, identifying hosts that provide highly disjoint alternative paths becomes non-trivial.

In this paper, we propose to select such relays by path similarity, based on previously discovered relays. The idea is that if a relay is suitable for a given path, it is also likely to be good for other, similar paths. Paths can be similar when senders and receivers are respectivelly geographically close to each other and/or serviced by the same ISPs. Even if this is not the case, similar BGP-level connections between ISPs traversed for paths might yield the same relays to provide highly disjoint alternative paths. With our path similarity-based approach, exhaustive topology probing to search for good relays can be avoided.

We study measures of path similarity and propose an algorithm to select relays based on previously used, similar paths. We evaluated our approach using 200 PlanetLab nodes. Our evaluation shows that we can indeed quickly identify relay nodes that lead to paths that are highly disjoint from the default Internet routes. Specifically, the cost of our approach in number of topology probes is a small constant (e.g. about 20 traceroutes), independent of the total number of nodes in the overlay, $N$. Alternatively, as explained in our evaluation section, an *exhaustive* search for the best relay for a given path is $O(N)$. The performance of our constant-cost method (i.e. the disjointness provided by relays found), while worse than that of exhaustive search, significantly improves over random search with equal, constant cost.

## 2. Identifying Relays for Alternative Paths

We propose to use a synthetic coordinate system modeling Internet path diversity to overcome the above-mentioned scalability problems. Below, we denote by the default Internet path, or simply default path between a source $s$ and a destination $d$, the set of routers that an IP package has to pass through when being routed between $s$ and $d$ on the Internet. We denote by alternative path between $s$ and $d$, through a relay host $r$, the union of default paths $(s, r)$ and $(r, d)$. We define the disjointness of an alternative path as the number of routers in the default Internet path that are *not* part of the alternative path.

## 2.1 Synthetic Coordinates for Path Disjointness

Our approach is based on the idea of *path similarity*. The intuition behind is that many Internet paths exhibit similarity w.r.t. which relays provide them with good disjointness. For instance, a relay that is good for a path between New York and Amsterdam, is likely to also be good for a path between e.g. New Jersey and Brussels. In the following, we will try to evaluate this hypothesis and quantify the afore-mentioned probability.

When building a synthetic coordinate system for latency prediction ( [7], [3]), distance is easy to measure, e.g., as the round-trip time between hosts. When modeling path disjointness provided by other peer nodes, "distance" is less obvious. For this purpose, we define path similarity between paths P1 and P2 as the probability that a relay that is good for P1 is also good for P2. This probability can be quantified in multiple ways, depending on the working definition of relay goodness. In this paper, we propose and evaluate 3 such similarity functions: SimKendall (based on Kendall rank correlation [12]), SimPearson (based on classic Pearson correlation) and SimEuclidian (based on Euclidian distance).

In all cases, a path's coordinates are derived as an $N$-tuple. Each tuple element represents the disjointness provided to that path by a relay. A consistent, randomly chosen relay set, $RS$, is used and maintained for determining coordinates for all paths. A path's coordinates can be derived by direct topology probing (e.g. using traceroute). For each path positioning, $2 * |RS| + 1$ topology probes (where $|RS|$ is the size of $RS$, typically about 10-20) are needed: 1 probe from source to each relay, 1 from each relay to destination, and 1 for the default path itself.

Once a path is positioned in the coordinate space this way, relays that were previously found to be good for paths close to ours in the path similarity space are also likely (with a probability given by the similarity function) to be good for our path. Path coordinates are calculated in the same manner for all versions of our algorithm; the difference is only in the way the path similarity between two paths is calculated (thus the distance function of the coordinate space). We detail the three similarity functions evaluated in the following.

## 2.2 Path Similarity Evaluation Functions

Kendall's rank correlation is a non-parametric measure of correlation between two sets of values, which gives the probability that any two corresponding pairs of values in the two sets are concordant (identically ordered). In our case, if we consider the two sets to be the disjointness

values provided by the same, consistently ordered, set of relays for paths $P1$ and $P2$, the Kendall correlation gives the probability that if a relay $R1$ provides a better disjointness than $R2$ (note $R1 > R2$) for $P1$, we also have $R1 > R2$ for $P2$. Consequently, if a relay $R$ provides high path disjointness for $P1$ relative to the entire set of relays used, it will also have a high rank, thus will be a good relay, for $P2$. We denote by *SimKendall(P1,P2,RS)*, the similarity between $P1$ and $P2$, through relay set $RS$, as given by the Kendall rank correlation test.

The second path similarity estimator we used, *SimPearson*, is perhaps the most commonly used correlation measure in statistics, based on linear regression. We considered Kendall in addition to the more classic, parametric, Pearson correlation, because it is distribution-free (not assuming the distribution of disjointness, as defined above, to be uniform), less sensitive to outliers and more accurate for small samples [2](which helps minimizing the number of probes needed for initial positioning).

*SimKendall* and *SimPearson* quantify the relative ordering of relays for the two paths, w.r.t. disjointness provided. Basically, we identify a path in the path disjointness space by the relative order of a consistent set of relays. While Pearson takes into consideration the actual disjointness values, Kendall only considers rank order. However, none takes into account the difference between the average values of the two disjointness sets (e.g. {5, 6, 7, 7} and {1, 2, 3, 3} are identic as far as they are concerned). This is on purpose, based on the following insight: while the actual number of routers from the direct path avoided by using a same relay may vary from path to path, we hypothesize that the "relative goodness" of relays is enough to characterize a path's positioning in the Internet w.r.t. path disjointness. This is true because the feature we are interested in is exactly reusing the best relays from similar paths. However, for comparison purposes, the last function takes into account the afore-mentioned difference. It is calculated based on the Euclidian distance in the space formed by the disjointnesses provided to a path by a consistent set of relays.

## 2.3 Relay Identification Algorithm

We store the needed information in a database (called PathCache) containing Internet paths, together with relays that provide high disjointness and respective disjointness values. Our current PathCache implementation is centralized but work is ongoing on a distributed version.

Figure 1 outlines a very simple, distributed algorithm that uses path similarity coordinates and PathCache to find good relays. Initially, $RS$, a set of random relays, is picked from all nodes of the overlay and pub-

lished through a shared database. This set will act as a consistent random sample for path similarity calculations. Statistical studies [2]show that, for accurate calculation of correlations (Kendall), the minimum sample size should be between 10 and 20. This would thus also be a reference minimal size for $RS$. Making sure that the nodes in $RS$ are alive is handled by a separate polling algorithm: if nodes in $RS$ become unavailable, new random nodes are picked and published instead. In Fig. 1, when a node needs relays for a new path $P$, first the set $D(P, RS)$ containing disjointness values provided by relays in $RS$ for path $P$ is calculated ($2 * |RS| + 1$ topology probes required). This set represents the Path Diversity (PD) coordinates of path $P$. PathCache is then queried for the most similar paths for which PD coordinates have previously been published (a k-Nearest Neighbors query with distance function based on one of the similarity functions). The best relays that have been discovered so far for these paths are then used for our path.

As we can see, the algorithm relies on previously discovered relays. To ensure that the database is populated with good relays, several approaches are possible. For instance, each node can periodically randomly probe for good relays for random PathCache paths and publish them in PathCache. However, it is important to note that the search for better relays to populate PathCache can be done in parallel with queries, and need not be in real time. Conversely, relays found in this search are likely to benefit multiple queries in the future.

```
getDisjointRelaysPD(P)
  RS ← PathCache.getSampleRelaySet()
  Coord_p ← getDisjointnessSet(P, RS)    // 2 · |RS| traceroutes needed
  SimilarPaths ← PathCache.kNNQuery(Coord_p,SimFunction)
  DisjointRelays ← PathCache.pickTopRelays(SimilarPaths)
```

*Figure 1.* Sample use of PD coordinates to find good relay nodes.

## 3. Evaluation

To evaluate our system we used Internet topology traces obtained on a platform of about 200 geographically distributed PlanetLab nodes. We have fed these topology measurements into a trace-driven simulator of an overlay network, based on PlanetSim . Nodes ran the algorithm described above to derive their coordinates and find relays.

### 3.1 Quantifying Internet Path Diversity

In this section, we present an evaluation of the amount of path diversity inherent to the Internet. Specifically, we characterize the distribu-

tion of disjointness values provided to Internet paths by a single, third relay. Fig. 2.a) presents a cumulative distribution function of the disjointness ratio over about 400 random Internet paths using 150 random relay hosts. Here, we define the disjointness ratio as the ratio between the disjointness provided by a relay and the total length of the path (i.e. number of routers in it). We thus normalize for different path lengths. In the figure, the X axis represents disjointness ratio and Y the probability that the disjointness ratio provided by a random relay is $< X$.

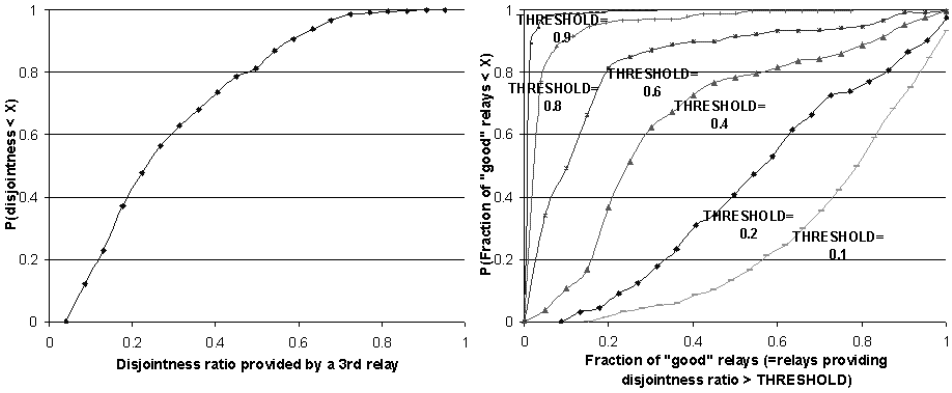


*Figure 2.* a) CDF of disjointness provided by a third relay to default Internet paths. b) CDFs for distributions of paths w.r.t. fraction of relays providing high disjointness.

We can see that the probability $p$ that an alternative path through a random relay will *avoid less than* $r$=49% of the routers in the default path is of 81%. For $r$=72%, $p$=98%, whereas for $r$=90%, $p$=99.89%. These results suggest that indeed, for large system sizes, randomly looking for relays will perform poorly. Fig. 2.a) basically characterizes the probability distribution of disjointness provided by random relays, averaged over all paths. Depending on the Internet path, this distribution may vary. For instance, depending on the Autonomous Systems the ends belong to, good relays might be easier to find for some paths than for others. In this respect, Fig. 2.b) characterizes the distribution of Internet *paths* w.r.t. the number of good relays that exist for them. "Good" relays are defined as those that provide disjointness ratio larger than a certain threshold; CDFs for various values of this threshold are plotted. The X axis shows the fraction of "good" relays, given the respective THRESHOLD. We see that the CDF becomes very steep as THRESHOLD increases; for instance, if we consider good relays to be those providing disjointness ratio $> 0.8$, we note that for 77% of Internet paths, there exist less than 3% good relays.

## 3.2 Evaluation of Synthetic Coordinates

We now evaluate the performance of searching for good relays using path similarity-based synthetic coordinates. Fig. 3.a) shows CDFs of disjointness provided to paths by relays found with our heuristic. Performance is compared with the "optimal" algorithm (if knowledge of the complete relevant topology would be obtained and the absolute best relays would be picked) and the "random" algorithm, the latter using R random picks and choosing the best relay found, where R is the actual number of relays that are probed in our approach (basically the size of the $RS$ set mentioned in section 2). This way, the two approaches are comparable in terms of cost in topology probes. Obviously, the optimal approach is much more expensive: $(2 * (N - 2) + 1$ probes needed to aquire topology information on all possible relays for a given path, $N$ beeing the size of the network), but is plotted as a reference. From Fig. 3.a), we can see that our approach significantly improves on the quality of the relays found by random search. While our performance is sub-optimal, let us recall that we only require a low constant cost in topology probes as compared to the O(N) cost of the optimal approach. We evaluated our heuristic with the three similarity functions presented. As expected, especially the Kendall and Pearson metrics seem to perform well. This confirms our hypothesis that the order of the reference set of relays w.r.t. disjointness provided is a good heuristic in positioning Internet paths among each other in what path disjointness is concerned.

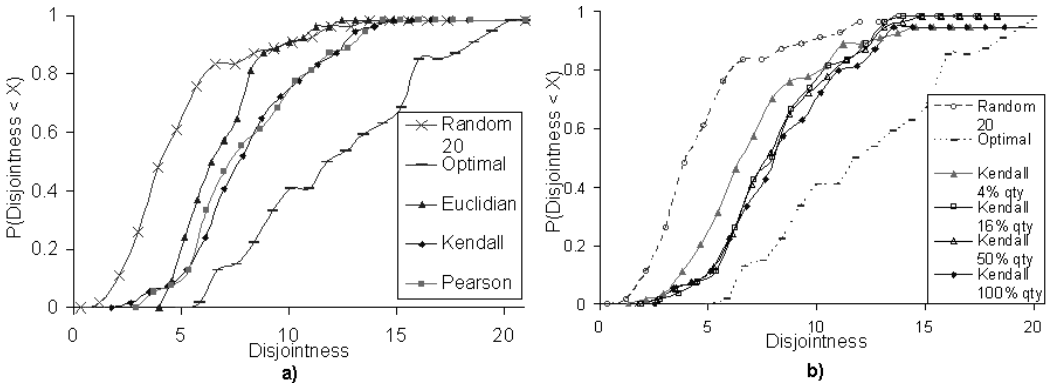


*Figure 3.* a) CDFs for disjointness distributions of relays found by our heuristic, with methods: Euclidian, Pearson, Kendall vs. random and optimal. b) CDFs for Kendall PathCache at relay quality levels: 4%, 16%, 50%, 100% vs. random, optimal.

The main parameters that can influence the performance of our approach are the similarity function used, $|RS|$, the PathCache "fill ratio" (i.e. number of paths that exist in PathCache) and what we call the

PathCache "relay quality". This last parameter basically reflects the average quality of relays published for PathCache paths. As mentioned in subsection 2.4, a separate algorithm can optionally be employed in the background to improve the quality of relays published in PathCache, based on periodic random probes. We estimate a published path's "relay quality" as the number of random probes that were executed so far to derive its current relay set. PathCache's overall "relay quality" is an average of the relay qualities of the paths it contains. We found that search performance does not significantly depend on $|RS|$, as long as it is at least 10. Therefore, an accurate positioning can be done with about 20 traceroutes. Also, against statistics community guidelines, we found Kendall only marginally better than Pearson at small sample sizes.

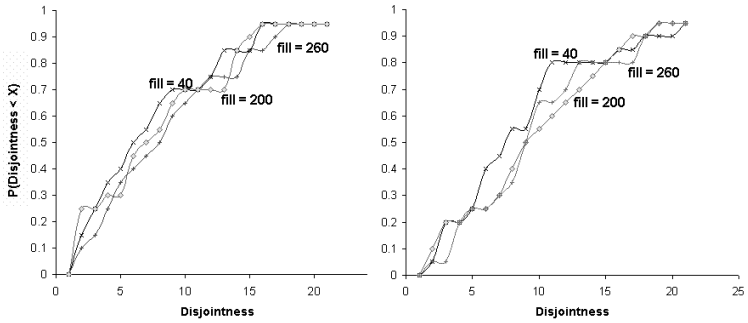


*Figure 4.* CDFs of average disjointness for relays returned by PathCache at various cache fills: 40, 200 and 260 paths. Similarity metrics: Kendall (left), Pearson (right).

Fig. 4 shows how the distribution of disjointness provided by returned relays varies at various PathCache fill ratios for both Kendall and Pearson similarity metrics. We can see that the quality of relays does improve as the cache contains more paths and that Kendall seems to be slightly more appropriate at large cache sizes than Pearson. As reference, CDFs in Fig. 3.a) were plotted at a fill ratio of 265 paths.

In Fig. 3.b), we evaluate PathCache at various average relay quality levels. We only present the Kendall evaluation, as the results for Pearson are very similar. Qualities are shown in percentages of total search space covered (100%=exhaustive search). Again, this is the background search done for a limited number of paths, not the search needed at query time for all paths. We see that relays returned by PathCache get better as average relay quality increases. However, even for very low relay qualities (lowest quality tried is 10 random samples per path), the improvement is consistent when compared to random search. The improvement increases slowly between 16% and 50% quality levels, becoming slightly more consistent as the quality approaches exhaustive search. A cause

might be that, as seen in subsection 3.1, extremely good relays are relatively rare, thus hard to find in a large set of nodes without exhaustive search. Where our approach helps a lot is in quickly finding relays that were relatively good for similar paths, thus significantly improving over random search. This is useful, because we can imagine that continuous, long-lived random (or even exhaustive) searches can be conducted in the background for a limited number of paths that populate PathCache, benefiting all future path queries.

## 4. Related Work

Recent work is focusing either on Internet topology discovery [9]or on using overlay networks for improving QoS via multipath routing [8, 1]. Resilient Overlay Networks (RON) [1]uses overlays for routing around congestions and network outages. RON and [8]rely on small overlay network sizes for which topology can be discovered by exhaustive probing. Therefore, we consider these approaches applications that could benefit from our scalable discovery of relays. Other work on QoS-improving overlays [10]is focusing on packet loss reduction via Forward Error Correction and alternative path routing rather than *discovery*. A *routing underlay* is suggested in [6], exposing topology information to overlays on top, avoiding redundant probing. The approach differs from ours as AS-level path inference, rather than path similarity, is employed.

[4]suggests a heuristic for finding good alternative paths in large systems. It considers paths at BGP- rather than router level. The approach relies on the *earliest divergence rule*, stating that BGP paths that diverge from the default path the earliest have a high chance of converging later. Compared to ours, this approach requires unbounded probing of candidate relay nodes and BGP information. RSIM [5]is a *node* similarity metric, used to *predict* path similarity. It is based on the number of common routers shared by paths from two sources to multiple destinations. In comparison, we directly estimate and employ path similarity. Using synthetic coordinates to faster predict Internet properties was extensively pursued ( [7], [3]), however focusing on latency. To the best of our knowledge, we make the first attempt to derive and use synthetic coordinates for path disjointness prediction.

## 5. Conclusions

Using alternative paths helps improving QoS of communication across the Internet. Such paths can be formed by using explicit relays among the nodes of an overlay network that lead communication around network bottlenecks. The crux of the approach is to identify a suitable

relay that leads to a new path which is highly disjoint from the default path given by Internet routing. In small-scale systems, such relays can be identified by exhaustive search of the overlay network. Our work focuses on large-scale, possibly dynamic, systems (e.g. volunteer p2p networks), in which such exhaustive search can not be applied. We propose a technique that identifies good relays from the ones that have been suitable for similar paths in the recent past. Our evaluation on 200 PlanetLab nodes shows that we successfully identify suitable relays much faster than exhaustive search, with low, constant cost. We are currently investigating distributed storage mechanisms for path similarity data to enable a distributed PathCache implementation.

## Acknowledgments

This work is partially supported by the *CoreGRID* Network of Excellence, funded by the European Commission's FP6 programme (contract IST-2002-004265).

## References

[1] D. G. Andersen, H. Balakrishnan, M. F. Kaashoek, and R. Morris. Resilient overlay networks. In *Proceedings of ACM SOSP*, 2001.

[2] D.G. Bonett. Sample Size Requirements for Estimating Pearson, Kendall and Spearman Correlations In*Psychometrika*, 2000.

[3] R. Cox, F. Dabek, F. Kaashoek, J. Li, and R. Morris. Practical, distributed network coordinates. In *ACM SIGCOMM Computer Communication Review*, 2004.

[4] T. Fei, S. Tao, L. Gao, and R. Guerin. How to Select a Good Alternate Path in Large Peer-to-Peer Systems. In *Proceedings of IEEE INFOCOM*, 2006.

[5] N. Hu and P. Steenkiste. Quantifying Internet End-to-End Route Similarity. In*Passive and Active Measurement Conference (PAM)*, 2006.

[6] A. Nakao, L. Peterson, and A. Bavier. A routing underlay for overlay networks. In *Proceedings of ACM SIGCOMM*, 2003.

[7] T. S. E. Ng and H. Zhang. Predicting Internet network distance with coordinates-based approaches. In*Proceedings of IEEE INFOCOM*, 2002.

[8] T. Nguyen and A. Zakhor. Path diversity with forward error correction (PDF) system for packet switched networks. In *Proceedings of IEEE INFOCOM*, 2003.

[9] N. Spring, R. Mahajan, D. Wetherall, and T. Anderson. Measuring ISP topologies with Rocketfuel. In*IEEE/ACM Transactions on Networking*, 2004.

[10] L. Subramanian, I. Stoica, H. Balakrishnan, and R. H. Katz. OverQoS: An Overlay Based Architecture for Enhancing Internet QoS. In *Proc. NSDI*, 2004.

[11] B. Zhang, T. S. E. Ng, A. Nandi, R. Riedi, P. Druschel, and G. Wang. Measurement based analysis, modeling, and synthesis of the Internet delay space. In *Proceedings of ACM SIGCOMM on Internet measurement*, 2006.

[12] M.G Kendall. A New Measure of Rank Correlation. In*Biometrika, Vol. 30, No. 1/2, 81-93. Jun.*, 1938