# Discovery of In-Band Streaming Services in Peer-to-Peer Overlays


John Buford
*Avaya Labs Research*
*Lincroft, NJ 07748, USA*
*buford@avaya.com*

Angela Wang, Xiaojun Hei, Yong Liu, Keith Ross
*Polytechnic University*
*Brooklyn, NY 11201, USA*
*{ywang07,heixj,yongliu,ross}@poly.edu*



***Abstract*--Peer-to-peer overlays can be used for service discovery over a global network fabric. We describe and evaluate a new service indexing mechanism for in-band streaming services such as application relays, mixers, and media transcoders. For this type of service, the location of the service in the network and service admission status are key attributes. We describe and analyze a service indexing mechanism which uses network position-based advertisement. We show that this mechanism gives good service selection, provides a close to uniform advertisement distribution in the overlay, reduces message overhead, and exhibits acceptable stability and setup delay.**



***Keywords-service discovery; application relay; DHT; overlay.***


## I. INTRODUCTION

Real-time streaming applications are growing in importance as an overlay application [1]. The overlay can provide a variety of services to support streaming overlay applications. Such services include various types of application layer multicasting, content caching, and content-based search. In this paper we examine the advertisement and discovery of in-band streaming services such as application relays, mixers, and transcoders. In this category of services, network placement of the service-offering peer is a primary attribute used in the discovery process.

The conventional approach for service discovery in structured overlays is to parse service description documents into keyword fragments called strands and index these strands separately into the Distributed Hash Table (DHT). Services which have a high frequency of occurrence in the overlay are not effectively indexed as strands of service description documents. The indices will be heavily clustered at a few points in the DHT, and there are many strands per service description.

Further as we discuss in [9], such strand indexing may not be suited to service discovery patterns such as location-based discovery or meta-based discovery. Also, as we discuss later, they do not address dynamic aspects that are important for certain types of service discovery such as current work load and availability. Consequently, different indexing mechanisms depending on the type of service being advertised can co-exist in the service overlay.

In this paper we make the following contributions:

- We describe a new service discovery mechanism for wide-area peer-to-peer overlays for important classes of services in which service attributes such as network position and peer work load are important discovery parameters. This method uses network coordinate-based discovery.
- We evaluate this by analysis of application relay discovery using an internet coordinate system.
- We compare this approach with the conventional technique such as candidate caching and direct probing, and show that DHT based discovery has generally less overhead for the same quality of service selection.

The remainder of this paper is organized as follows. We review related work in the next section. Section 3 characterizes several in-band streaming services. Section 4 presents the position-based discovery method, and Section 5 describes position-based advertisements in this method. Section 6 provides a cost comparison of several in-band service selection schemes. Section 7 concludes the paper.

## II. RELATED WORK

Application relays can reduce end-to-end delay and increase throughput for TCP connections. Liu et al. [2] analyze application relays and show optimal relay selection algorithms for multi-relay paths in a connected overlay graph. Ren et al. [3] describe the ASAP cluster-based relay selection mechanism which selects relays to insure that the direct path RTT is less than 300 msec., meeting an important constraint for voice calls.

The development of accurate mappings of internet address space to a geometric coordinate space has been studied for its potential use in a number of applications. Examples include GNS [4], Landmarks [5], and Vivaldi [6]. The limitations of the accuracy of existing ICSs are well known, and a recent analysis is found in [7], which also suggests using subspace embeddings to further improve ICS accuracy. In this paper, using an existing super-space ICS, we combine direct probing with ICS-based selection to validate the ICS selection. If a future ICS corrects the problems due to triangle inequality violations in internet routing, then direct probing can be eliminated, further improving the method described here.

In the area of stream-based overlays (SBONs) there is a related problem of placement of stream operators in the network. In evaluation with two different multi-hop overlays (Bamboo and a Pastry-like DHT called Pan), [8] shows that using paths provided by the routing layer of the overlay produces poor results for optimizing the cost problem (e.g., network usage) of operator placement in the SBON.

## III. STREAMING SERVICES IN THE OVERLAY

To improve the performance of streaming in the overlay, addition capability can be added such as application relays, media transcoding, and mixers. Peers which provide these functions can be dynamically integrated in to streaming sessions. This provides more flexibility in meeting QoS, fair load distribution, reliability, and performance goals. Dynamic selection of peers for these infrastructure roles is a service discovery problem.

Unlike many services, the invocation of such streaming services (TABLE 1) is typically in band during session initiation. These services operate continuously on the media streams during the session. Session characteristics might change, requiring re-negotiation or re-selection.

TABLE 1
IN-BAND STREAMING SERVICES

| | Relay | Mixer | Transcoder |
|---|---|---|---|
| **Summary** | Protocol endpoint to reduce end-to-end delay | Adds 2 or more streams of the same format in a multi-point session | Translates a stream from one format to another |
| **Media Handling** | No | Yes | Yes |
| **Admit criteria** | Available bandwidth | Available bandwidth<br>Media formats<br>CPU utilization | Available bandwidth<br>Media formats<br>CPU utilization |
| **Advert. Attributes** | Current bandwidth capacity<br>Availability<br>Network Position & Address<br>Max/Avg Delay & Jitter | Media format(s)<br>Current bandwidth capacity<br>Max # of streams per mixer<br>Availability<br>Network Position & Address<br>Max/Avg Delay & Jitter | Media formats<br>Current bandwidth capacity<br>Max # of streams per mixer<br>Availability<br>Network Position & Address<br>Max/Avg Delay & Jitter |

A service instance is described by a combination of fixed and dynamic attributes. Fixed attributes include the media formats supported, network address, function, and guarantees on jitter and delay. Dynamic attributes include current capacity and availability.

## IV. RELAY DISCOVERY AND SELECTION

### *A. Overview*

Application relay discovery and selection is an important problem for streaming media applications, and is representative of other in-band streaming services. TCP throughput B over a link is $B \sim 1/d{\cdot}p^{1/2}$ [2], where d is the link delay and p is the packet loss probability. Assuming homogenous packet loss probability on all links, then the effectiveness of the relay for increasing throughput and reducing end-to-end delay depends on its proximity to the mid-point position in the end-to-end delay between the end points. Although multiple relays can be used along a path, here we analyze the case of a single relay. The discovery method we describe is applicable to multi-relay and multi-point bridge selection.

Using round-trip time measurement as an estimator of packet delay over a path, a direct way to find a relay which is located closest to the mid-point position is to measure the round-trip time between each candidate relay and each of the end points to obtain $RTT_1$ and $RTT_2$. Since the end-to-end throughput is constrained by the segment with the largest value of d, choose the relay which produces the minimum of $RTT_1+RTT_2+ABS(RTT_1-RTT_2)$. This requires at least two probes per relay candidate, one for each segment.

In a large overlay there are many thousands, perhaps millions, of possible candidates. We expect the population of relay candidates to be a significant portion of the overlay population in order for there to be fair load distribution and sufficient capacity for many concurrent streaming sessions. Because of the large number of peers, it is impractical to probe all the candidates. It might be possible for a peer to cache RTT measurements for a large number of other peers. However, the other endpoint must also cache RTT measurements, and the set of cached measurements must be for common peers, otherwise additional measurement is needed, reducing the utility of the cache. Let the overlay size be N, the probability that a peer is a relay be f, and the size of any peer's relay candidate cache be m. The probability that any relay candidate is in both caches is $(m/f{\cdot}N)^2$ and the expected size of the intersection of any two caches is $E(|m_1 \cap m_2|) \leq m\,(m/f{\cdot}N)^2$. TABLE 2 shows some representative values for N, f, and m. Thus direct probing doesn't scale for relay selection for large overlays, and caching measurements doesn't eliminate the need for large amounts of probing in large overlays.

TABLE 2
FOR OVERLAY OF SIZE N, PROBABILITY OF A PEER BEING A RELAY F, AND RELAY CANDIDATE CACHE SIZE M, EXAMPLE VALUES OF THE PROBABLY OF ANY RELAY CANDIDATE BEING IN BOTH CACHES AND THE EXPECTED SIZE OF THE INTERSECTION OF ANY TWO CACHES

| N | f | m | $(m/fN)^2$ | $E(\|m_1 \cap m_2\|)$ |
|---|---|---|---|---|
| 1,000,000 | .3 | 1,000 | 0.000011 | 0.011 |
| 1,000,000 | .1 | 1,000 | 0.0001 | 0.1 |
| 10,000,000 | .5 | 10,000 | 0.000004 | .04 |

We describe a new method for relay selection using service discovery in the overlay in which relay service advertisements are indexed according to the estimate of the relay position in some coordinate system. This approach is applicable to other communication services such as media transcoders and mixers, since for communication services that are in band it is important that end-to-end delay be reduced and bandwidth be maximized.

In addition, relay candidate suitability varies by dynamic attributes such as peer load and availability which are not typically considered in existing service discovery mechanisms. Finding candidate relays is a service discovery problem. The method we describe incorporates both coordinate estimation and use of dynamic attributes.

### B. *Definition of the Relay Discovery Problem*

A peer $p_1$ initiates a relayed session with a remote peer $p_2$. Peers $p_1$ and $p_2$ must agree on an intermediate peer to act as a relay. We call those peers which are prospective relays *candidate relays*. Identifying those peers which are candidate relays is *candidate discovery*. Choosing a relay from a set of candidates is *relay selection*. Assuming that peers engage in many sessions to different peers in the overlay, having a large set of candidate relays is a prerequisite condition for selecting a high quality relay for a session. This is due to the arbitrary distribution of session endpoints in the overlay and the Internet-scale overlay topology.

We assume that peers and relays are part of an overlay network. To be a relay candidate, a peer must have a public internet address and have sufficient capacity.

The are several ways for a peer to obtain relay candidates, including configuration, using entries in the peer's overlay routing table, sharing relay selection history between proximal peers, and explicit advertisement and lookup. If the peers in the overlay register with a bootstrap server, then when a peer joins the overlay it can receive a configured list of relay candidates. Peers can remember relay selection history for future sessions, and proximal peers can share such history. Peers which participate in the overlay have a routing layer which maintains address information about other peers in the overlay, and this information is updated as peers join and leave the overlay. Routing table entries could be used as relay candidates. For multi-hop overlays routing table size is O(log N). However for one-hop overlays a routing table of .5 N or more is maintained at each peer.

TABLE 3
TOPOLOGY DATA SETS USED TO EVALUATE APPLICATION RELAY SELECTION USING AN ICS

| | MIT [6] | Cornell [10] | Ohio State (OSU) [3] |
|---|---|---|---|
| # endpoints | 1700 | 2500 | 1103 |
| Source | Gnutella | DMOZ and Yahoo directories | Gnutella |
| RTT Measurement | Median of King measurements, filter out 10% of outlying nodes | Median of 10 King measurements | Median of 6 King measurements, remove nodes with incomplete measurements |
| Average RTT | 180 ms | 74 msec | 145 msec |
| Measurement date | 2004 | May 2004 | Jan 2007 |
| Measurement Period | 1 week | 9 days | 1 day |

Given such a list of nodes, then the position of the node in relation to the end points is important for relay selection. There are two ways to obtain this position information, by direct probing and by estimation using a coordinate system.

We assume that peers are organized in a DHT, and that peers can identify themselves as relay candidates for discovery by other peers by placing an advertisement in the DHT. An advantage of this compared to locally stored historical or configured lists is the advertisements are refreshed by the relay peers when changes in the relay state occur.

Application relays do not have rich APIs such as would exist for web services applications. Instead, important differentiating criteria are the relay position in the Internet with respect to the end points, capacity for additional media sessions, and availability. For example, to use hash-based indexing in a DHT, we could construct a key such as: "media-relay:coordinates" which returns a relay service description which includes current capacity and availability.

We form this advertisement using some type of global coordinate system, which could be either geographic or an Internet Coordinate System (ICS). Previously we described a geographic grid for indexing location-based services in a DHT [9]. Next we adapt this grid approach for using an ICS. The use of a grid eliminates mismatches due to fractional differences in coordinates, and generally increases the probability of a match.

### C. *ICS Position-based Discovery*

An ICS maps network position to a Euclidean space with a distance function. If the coordinates of the session end points and the relay candidate are known, then relay selection becomes finding the relay candidate that is closest to the midpoint position in the ICS space. Generally the use of an ICS trades off measurement probes versus computation. It is important that the ICS provide a globally consistent coordinate scheme, so systems such as Meridian [10] which calculate relative positioning can not be used for DHT indexing.

To demonstrate the suitability of using ICS coordinates for relay indexing, we use two-phase relay selection. First, the relay's ICS coordinates are used for candidate selection by DHT lookup to find k relay candidates. Then each k′ relay candidate is directly probed for an RTT measurement. Finally the relay r from k′ which satisfies the admission test and minimize the end-to-end delay is selected. We evaluate relay discovery using the Vivaldi [4] ICS using 3 different data sets which measure RTT for a set of hosts selected from the Internet. The data sets are summarized in Table 3.

Figure 1 shows the resulting relay selection for end points in the data sets with RTT > 200 msec. In each case selection of the best five candidates using the ICS coordinates followed by probing to choose the best one produces comparable results to randomly probing 10 nodes followed by choosing the best one. For comparison, the "brute force" case is the optimal relay in the data set which would be found by probing all relay candidates, an impractical approach in general. Thus, use of the ICS can substantially reduce the need for probing during relay selection.

## V. ADVERTISEMENT USING NETWORK POSITION AND DYNAMIC ATTRIBUTES

### A. *DHT Indexing of ICS Coordinates*

Two end points are represented by coordinates $C_1$ and $C_2$ in the ICS. Relay candidates are sought in the vicinity of C which is midway between $C_1$ and $C_2$. We can align all coordinates on a grid and hash each relay's nearest grid position. Then discovery involves using the nearest grid position to C as the key. Each peer that acts as a relay determines its position in the ICS, aligns the position to the nearest grid point, and creates the key "media-relay:grid-coordinates" as its advertisement and inserts its relay service description at the position in the DHT indicated by the key. For example, Vivaldi can produce six dimensional coordinate values. So a key might be: "media-relay:$g_1$:$g_2$:$g_3$:$g_4$:$g_5$:$g_6$" where each $g_i$ is an integer in one of the coordinate dimensions.

We evaluate the uniformity of the distribution of such keys in DHT using SHA hashing function used by many DHTs. The distribution of N relay keys in a DHT of 2N peers using the OSU, Cornell, and MIT data sets is shown in Figure 3 and shows a distribution comparable with randomly generated keys. The average number of keys per node is about .5 and the maximum keys per node is 3.

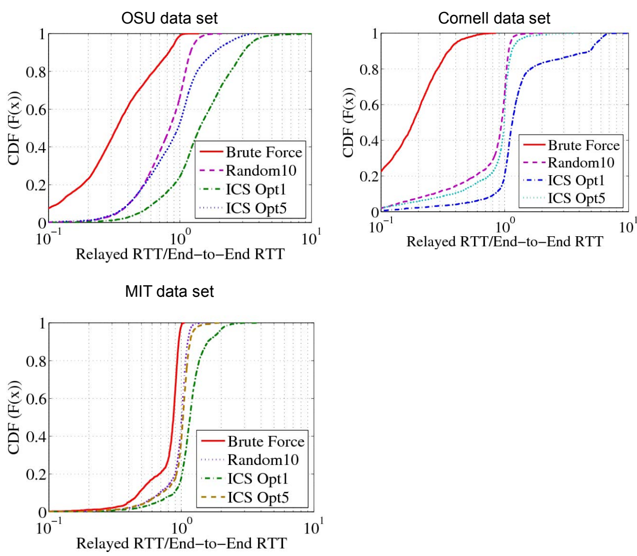


Figure 1 For end points whose RTT is > 200 msec, the CDF for RTT of selected relay comparing brute force, 10 randomly selected peers in the data set, best five from the ICS distance function, and best one from the ICS distance function. The x axis is the ratio of the RTT via the relay versus direct.

## B. *Churn and Coordinate Recalculation*

We evaluate the stability of the Vivaldi coordinates since frequent coordinate changes would make the service index less reliable and add overhead for index maintenance.

A Vivaldi node periodically re-computes its coordinates due to arrival and departure of nodes in the overlay. At each calculation interval called an iteration, it updates its coordinates using the RTT values between it and a set of other nodes to be used in the calculation. The size of the set used in our experiments was 32. From one interval to the next, the sets can overlap or be entirely different. The interval depends on the churn rate of the overlay.

For each of the three data sets we randomly selected 90% of the nodes and computed the Vivaldi coordinates for each. The coordinates stabilize after 15 or more intervals. After the coordinates stabilize we then included the remaining 10% of the nodes and recomputed all peer coordinates in one interval. We calculated the ratio of the distance between the coordinates for each point versus the size of the coordinate space. For all 3 data sets the average % change was < 1%. Figure 2 plots the % change sorted in decreasing size for each data set. The % of the data set whose coordinate change exceeded the grid spacing was 2% for Cornell data set, 10% for MIT data set and 16% for OSU data set. The rate of advertisement change also depends on the churn rate which for relays should be substantially less than for non-relay nodes.

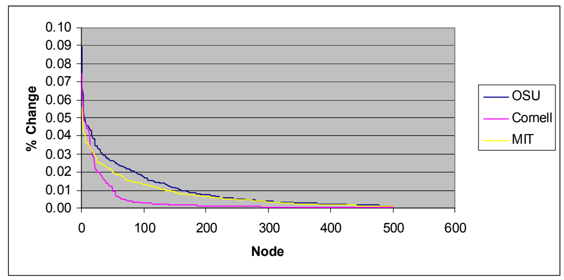


Figure 2 Percent change in peer coordinate values, starting from stable values for 90% of the peers in each data set and then merging in the remaining 10%

## C. *Load Distribution*

The distribution of DHT index operations due to relay advertisement and discovery should be evenly distributed. As shown in Figure 3 the keys show a somewhat uniform distribution in the DHT. The distribution of lookup requests depends also on the distribution of end points in the set of sessions as well as the topology of the Internet. For example [11], in analyzing the distribution of RTT measurements versus geographic position found clusters associated with the US east coast, US west coast, and Europe/Asia. This suggests that relay discovery load is likely to be concentrated at points within each of these clusters as well as between these clusters.

## D. *Capacity and Availability*

The relay peer may have an admission control step which considers its current load and capacity to accept new sessions. After the end point peers select the relay, then admission control occurs when the initiating end point contacts the selected relay to initiate the session. If the relay has insufficient capacity, then the relay rejects the request and the end point must choose another relay candidate. To avoid delay in session set up due to failed admission, relays might include their current available capacity in the relay advertisement. Similarly, availability information about the relay itself might be an important criteria for relay selection, since it can be used as a predictor of the potential of the relay to complete the session.

This type of information is dynamically produced at the relay peer. It can be stored either at DHT index point as part of the relay service description or at the relay peer itself, depending on the rate of change ($r_u$) of this information versus the rate $r_r$ at which the advertisements are requested. If $r_u > r_r$, then the relay peer can store the relay service description locally and index a URI to its service description. Otherwise it can store the relay service description at the DHT index.

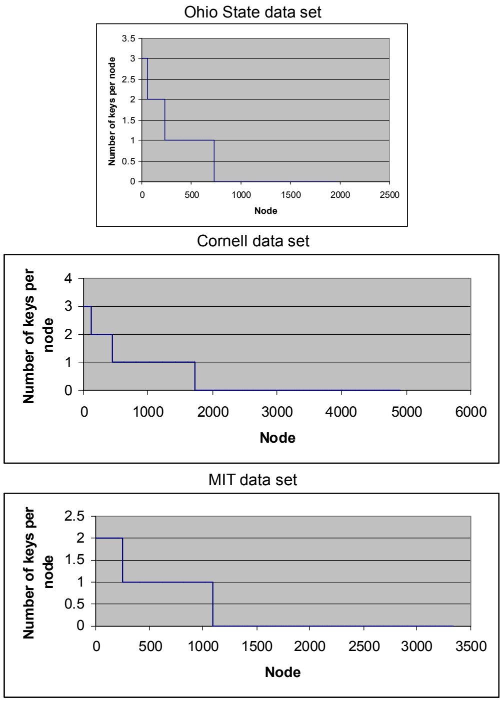

Figure 3 Key distribution of ICS coordinates for (top) 1100 peers from the Ohio State data set, (center) 2500 nodes from the Cornell data set, and (bottom) 1740 nodes from the MIT data set.

TABLE 4
PARAMETERS FOR MESSAGE COST ANALYSIS

| Symbol | Description | Example Value |
|---|---|---|
| N | Number of peers in overlay | 1,000,000 |
| RC | Number of relays in overlay | 500,000 |
| RCp | Number of relay candidates known to each peer | 1000 |
| h | Number of overlay hops for put or get | 1 - 20 |
| S | Number of relayed sessions initiated by each peer in time period | 10 |
| k | Number of relay candidates per session for DHT-based discovery | k = 5 |
| j | Number of relay candidates per session for direct probing, j = 2k | j = 10 |

## VI. SELECTION COST

### A. DHT-based Discovery versus Direct Probing

We compare the message cost for the 3 different relay discovery methods (direct probing, direct probing with caching, and DHT-based discovery) described earlier using the parameters shown in Table 4.

The relay discovery methods we compare are listed in Table 5. We make the following assumptions: 1) we exclude DHT maintenance traffic, since the DHT is used for many other purposes, 2) a DHT lookup for a position-based key retrieves multiple candidates in one lookup, since they all share the same grid position, 3) DHTs are either multi-hop O($\log_b N$) (e.g., Chord has b=2 and Pastry has b=16) or one-hop O(1) (e.g., EpiChord or OneHop), 4) for RTT caching, cache exchange is done in 2 messages per session, and (best case) cache intersection always produces half the candidates for selection with remaining candidates requiring additional probing, 5) each relay is also a peer that may participate as an end-point in other streaming sessions, 6) 32 RTT probes are needed to compute Vivaldi coordinates which are stable for the time to complete S sessions, 7) from our experimental work (section 4), j = 2k, 8) in the non-DHT schemes, nodes acquire relay candidates using overlay routing table and lookup traffic without cost. We divide the cost into two components: candidate collection and per session candidate selection. The sum of these two cost columns in Table 5 represent the total message cost for all nodes during a period in which all nodes participate in S sessions. We also show (Table 5, right column) the relay-specific overhead for each method so that load on the relays can be compared across the different methods.

TABLE 5
COST FORMULA FOR MESSAGE LOAD COMPARISON OF DIFFERENT RELAY DISCOVERY METHODS

| | Candidate Collection | Per Session Candidate Selection | Total Relay Work |
|---|---|---|---|
| RTT probe with no cache | 2*N | 4*j*S*N | 4*j*S*N |
| RTT probe with cache | 2*N+ 2*N*$RC_p$ | (2+2*j)*S*N | 2*N*(1+RCp)+ 2*(j+1)*S*N |
| Position-based lookup in DHT | 32*2*N + RC*h + (h+1)*S*N | 4*k*S*N | 32*2*RC + 4*k*N*S + RC |

In Table 6 we apply values from Table 4 to the formula in Table 5 to calculate representative values for total message cost. The total message cost is the sum of the candidate collection and candidate selection message counts. This analysis shows that using a DHT is better when multihop base 16 (such as Pastry) or a one-hop overlay is used. Use of the DHT-based discovery method has an additional advantage of reducing message load on the relays. ICS coordinate calculation has other uses in the DHT such as proximal node selection, so the ICS costs can be amortized over other operations.

TABLE 6
MESSAGE COST FOR DIFFERENT DISCOVERY AND SELECTION SCHEMES FOR OVERLAYS OF 1M NODES

| | N | f | RC | RCp | S | h | k | Candidate Collection | Candidate Selection | Relay Work |
|---|---|---|---|---|---|---|---|---|---|---|
| RTT probe with no cache | 10^6 | 0.5 | 0.5^6 | x | 10 | x | 10 | 2M | 400M | 400M |
| RTT probe with cache | 10^6 | 0.5 | 0.5^6 | 1000 | 10 | x | 10 | 2,002M | 220M | 2,222M |
| Position-based DHT Discovery | | | | | | | | | | |
| Multi-hop base 2 | 10^6 | 0.5 | 0.5^6 | x | 10 | 20 | 5 | 283M | 200M | 232M |
| Multi-hop base 16 | 10^6 | 0.5 | 0.5^6 | x | 10 | 5 | 5 | 126M | 200M | 232M |
| One-hop | 10^6 | 0.5 | 0.5^6 | x | 10 | 1 | 5 | 84M | 200M | 232M |

The relative performance stays the same if the size of the overlay is increased to 10x. If the number of sessions S is increased 10x, the performance of cache versus non-cache is comparable, and as S increases further, caching has lower message overhead (however, the cost model optimistically assumes that each pair of caches have half intersection, while our analysis in Table 2 showed this to be unlikely). If the percentage of relays in the overlay drops to 10% from 50%, the relative performance is similar. Increasing the number of candidates k, j makes DHT-based discovery improve further.

Improvements to the accuracy of the ICS would reduce k/j, further benefiting the DHT-based discovery method. For example, [7] shows that subspace embedding gives better ICS accuracy than super-space embedding.

### B. *Admission and Dynamic Attributes*

Assume that some percentage q of relay candidates are at capacity and can not admit additional sessions. In the direct probing methods this leads to up to q percent more relay selection messaging. In the DHT-based method, each relay sends 2*S number of status updates to the DHT index containing its position. As q increases, the message overhead on direct probing increases at a faster rate than for DHT-based discovery.

### C. *Relay Selection Delay*

The delay to complete relay selection is about 2 RTTs for direct probing, assuming each relay candidate can be probed in parallel. For small numbers of relay candidates this is a reasonable assumption, but for large numbers of relay candidates the probes would need to be serialized to prevent interference.

The delay to complete relay selection for the DHT-based discovery is 2 RTTs for the probing plus the DHT lookup which is in the best case about one RTT if a one-hop overlay is used.

## VII. Conclusion

Selection of peers to perform in-band stream services is important for providing good session quality for streaming overlay applications. Important in-band services include application relays, mixers, and media transcoders. Systems today use either configuration or gossip algorithms to discover relay candidates, and use RTT probing to select the relay. To improve upon these approaches we examined this as a service discovery problem, using the overlay indexing mechanism.

We described and evaluated a new service indexing mechanism which uses network position-based advertisements in structured overlays. By using an ICS we reduced probe messaging overhead. We showed that the advertisements are close to uniformly distributed in the DHT index, and that dynamic service attributes such as capacity and availability can be included in the advertisement step. Further, for representative operational scenarios, the DHT-based discovery when using multi-hop $O(\log_{16})$ or one-hop overlays has significantly less message overhead while the session setup delay is slightly higher.

## Acknowledgment

We gratefully acknowledge the use of the ASAP data set kindly provided by Xiaodong Zhang at Ohio State University.

## References

[1] S. Narayanan, D. Braun, J. Buford, et al. Peer-to-Peer Streaming for Networked Consumer Electronics. *IEEE Communications Magazine*. June 2007.

[2] Y. Liu, Y. Gu, H. Zhang, W. Gong and D. Towsley. Application Level Relay for High-bandwidth Data Transport', *First Workshop on Networks for Grid Applications (GridNets)* , San Jose, Oct. 2004.

[3] S. Ren, L. Guo, and X. Zhang, ASAP: an AS-Aware Peer-Relay Protocol for High Quality VoIP with Low Overhead, in *Proceedings of 26th International Conference on Distributed Computing Systems (ICDCS'06)*, Lisbon, Portugal, July 4 - 7, 2006

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

[5] L. Tang and M. Crovella. Virtual Landmarks for the Internet. *Internet Measurement Conference*, Miami, Florida, Oct. 2003.

[6] F. Dabek, Cox, R., Kaashoek, F., and Morris, R. 2004. Vivaldi: a decentralized network coordinate system. In *Proceedings of the 2004 Conference on Applications, Technologies, Architectures, and Protocols For Computer Communications* (Portland, Oregon, USA, August 30 - September 03, 2004). SIGCOMM '04

[7] E. K. Lua, T. Griffin, M. Pias, H. Zheng, and J. Crowcroft. On the Accuracy of Embeddings for Internet Coordinate Systems. *Internet Measurement Conf. (IMC) 2005*, Oct.19-21, 2005.

[8] P. Pietzuch, J. Shneidman, J. Ledlie, M. Welsh, M. Seltzer, and M. Roussopoulos. Evaluating DHT-Based Service Placement for Stream-Based Overlays. *Proc. of the 4th Intl Workshop on Peer-to-Peer Systems* (IPTPS'05), February 2005.

[9] J. Buford, A. Brown, M. Kolberg. Meta Service Discovery. *IEEE 3rd Intl Workshop on Mobile Peer-to-Peer Computing (MP2P) 2006*.

[10] B. Wong, A. Slivkins, E. G. Sirer. Meridian: A Lightweight Network Location Service without Virtual Coordinates. *SIGCOMM 2005*, Philadelphia, Pennsylvania, August, 2005.

[11] B. Huffaker, M. Fomenkov, D. Plummer, D. Moore, and k. claffy, Distance Metrics in the Internet, *IEEE Intl Telecommunications Symp. (ITS),* Brazil, Sept 2002.