IP-based Clustering for Peer-to-Peer Overlays

Piotr Karwaczyński
Institute of Applied Informatics, Wrocław Univ. of Technology, 50-370 Wrocław, Poland
Email: piotr.karwaczynski@pwr.wroc.pl

Jaka Močnik
XLab d.o.o., Teslova 30, SI-1000 Ljubljana, Slovenia
Email: jaka.mocnik@xlab.si

Abstract— **The efficiency of overlay networks built on top of the IP network is often threatened by the mismatch between the topologies of the overlay and the underlying IP network, resulting in unnecessary traffic and increased latencies. Substantial improvement can be achieved by optimizing the logical links between overlay nodes to better match the IP network topology.**

In this paper, we propose a new method for self-optimization of a DHT-based peer-to-peer overlay. Our method has no need for active measurement of inter-node latencies, thus minimizing network traffic costs of node insertion and topology maintenance. We verify our method by means of analysis of large data sets of latency measurements between arbitrary nodes on the Internet, proving correlation among common IP prefix length of communicating nodes and latency.

Index Terms— **peer-to-peer, overlay network, topology mismatch problem, proximity neighbour selection**

I. Introduction

Using a distributed hash table (DHT) data structure, built on a peer-to-peer overlay network has proven an efficient method for construction of large systems with publish/discover functionality [1]–[4]. As an example, such a network can be used to build a dependable service discovery infrastructure for very large, dynamic service-oriented systems upon [5]. Service providers advertise the offered services, using the service interface as the key in the DHT. Service consumers use a well known interface in order to lookup appropriate service providers. Use of the DHT provides inherent, natural load balancing of both the published data and the traffic induced by publication and discovery over all the nodes in the system, avoiding traffic hot spots and single points of failure.

The implementations of DHT use an overlay network, an application-level, logical network built on top of a physical network that enables key-based routing [6]. Nodes in the overlay are connected by logical links, each comprised of a number of physical links. A node is a neighbour of another one if they are connected by a logical link. In structured overlay networks, the structure of their topology is a result of a constrained neighbour selection process: peers can select only such neighbours that satisfy the constraints related to the peer identifiers, and thus create directed logical links. These constraints are unavoidable since they impose a suitable structure on the overlay topology, which is necessary for deterministic routing of messages in the overlay.

A routing path in an overlay network does not usually map to the optimal path in the physical network. A message routed on the overlay can traverse the same physical link multiple times before it reaches its destination, resulting in unnecessary additional latencies and network load. In the context of peer-to-peer systems, this problem was identified as a topology mismatch problem [7] and has been an active research area. In this paper, we present a novel approach to its solution.

Current methods for neighbour selection that attempt to match overlay topology to IP network topology are discussed in section 2. Section 3 outlines the algorithms we propose as a replacement for costly neighbour selection in the Tapestry system [3], [6]. Section 4 provides a statistical verification of our approach based on large data sets of measurements of latencies between nodes on the Internet that has been performed prior to implementation of our approach for testing purposes. Finally, we conclude in section 5.

Our contribution is on one hand a simple method for selection of neighbours based on static, readily available information (namely the IP addresses of the nodes), that does not involve costly periodic probing of many nodes. On the other hand, the information on close-by nodes is stored in the overlay network itself, being available to new nodes that join the network immediately, whereas the current methods require new nodes to probe many potential neighbours for their latencies, which results in network traffic overloads during the process of a node joining the network. Finally, as a result of analysis of three large data sets containing latency measurements between arbitrary nodes, the relationship between latency and longest common IP prefix length (LCPL) of communicating nodes is presented. The analysis proved that these two attributes are correlated and heuristics based on IP addresses can be used to generate a topology-aware overlay with small latencies.

This paper is based on "Self-optimization of a DHT-based Discovery Service," by P. Karwaczyński, and J. Močnik, which appeared in the Proceedings of the 2nd International Multi-Conference on Computing in the Global Information Technology (ICCGI), 2007, Guadeloupe, French Caribbean. © 2007 IEEE.

This work has been partially funded by the European Community under the FP6 IST project DeDiSys (Dependable Distributed Systems, contract number 4152), http://www.dedisys.org/.

II. State of the Art

Nodes belonging to an overlay network may select neighbours with no regard for the properties of the underlying physical network. On the other hand, overlays that map to the underlying network well enable more efficient routing and lower maintenance costs. However, it is difficult to collect information about the physical network topology at run-time. Therefore, nodes should select such neighbours that are close in terms of some network metrics – this approach is known as *Proximity Neighbour Selection (PNS)*. The most popular metric is network latency, but latency measurements in large-scale systems are not straightforward either.

Precise measurements based on multiple RTT (round-trip time) samples are rarely used due to the communication load they impose. Hence other measures are implemented, estimating proximity in the system. In [8], the authors thoroughly evaluate four of them: IP path length, AS path length, geographic distance, and measures related to RTT. They conclude that the most precise are estimations based on RTT, even if they are simplified.

The distance between two nodes can be estimated well by the King method [9], measuring the network latency between DNS servers of these nodes using recursive queries. It is particularly useful when the nodes cannot actively co-operate.

Another option is to analyse IP addresses or AS numbers [10], [11] of nodes to estimate if they are close to each other. The results may be imprecise due to international ISPs and the CIDR [12] technique, but can be obtained inexpensively and may be sufficient for applications that do not need precise measurements.

The estimation of distance between two nodes is a challenge in terms of precision and communication overheads. Unluckily, this is not the only challenge that must be tackled in order to create an overlay that maps to the physical network well. Since a node usually cannot afford to estimate the distance to all its prospective neighbours, another problem is how to narrow the set of such candidates to a reasonable number.

According to [13], probing only a small random subset of all possible neighbours gives very good results: an improvement of an order of magnitude, when compared to a system without any PNS method implemented. In [14], the dependency between the number of random samples and average lookup latency is studied: simulations suggest that probing 16 nodes from a permissible node ID range gives optimal results.

Another frequently used heuristic is to perform an expanding ring search. Potential neighbours are selected if they are neighbours of known nodes. The question arises when to stop: in the simplest case, after a fixed number of expansions. More practical approaches take a subset of best candidates in each step, treat them as an input for the next iteration, and continue until the best subset is found (hill climbing) [6], [15]. Although this procedure gives satisfactory results, it is costly and threatened by local optima.

A number of methods make use of clustering: landmark clustering [16]–[18] and clustering based on the network infrastructure [19]–[21]. The former assumes that nodes close to distinguished ones (landmarks) are close to each other, but suffers from landmarks' vulnerability to failures, overloading, and attacks. If the landmarks are elected at run-time to improve fault tolerance and resilience, the communication costs grow due to necessary election algorithms. The usability of the latter is limited by the availability of information on the network infrastructure: typically a node is unaware of its AS (autonomous system) number. Nevertheless, such information as the node IP address or DNS server can be obtained inexpensively and used effectively in overlays to improve their awareness of the underlying network.

The early work mentioning the IP-based clustering was related to the clustering of Web servers [21]. The authors were interested in grouping nodes that are not only topologically close but also under common administrative control. They claimed that IP-based clustering fails in about 50% of cases with regards to such requirements. However, these results should not discourage from using the structure of IP prefixes to solve the Proximity Neighbour Selection problem since its requirements are much looser.

According to the author's knowledge, the first approach to building a p2p overlay in accordance with the structure of IP prefixes was TOPLUS (Topology-Centric Look-Up Service) [22]. In TOPLUS, nodes that are topologically close (i.e. have common IP prefixes) are organized into groups. Furthermore, these groups form a multi-level hierarchy founded on common IP prefixes as well. Finally, the method results in the overlay structure that is very close to the structure of underlying Internet. In other words, overlay messages are routed to their destinations along a path that is usually very close to the shortest Internet path.

The feature that makes TOPLUS mostly useless in practical applications is a non-uniform population of ID space. Namely, the number of keys assigned to a group of nodes is approximately proportional to the number of IP addresses covered by this group, not to the number of alive nodes in the group. Consequently, some nodes may become overloaded whereas others – mostly idle. The authors are aware of this problem and propose a solution. However, since their solution requires global knowledge about the system state, it is not useful in practice.

The application of knowledge on the structure of IPv6 addresses to Chord overlay is simulated in [23]. The results are promising, but they are completely founded on a two-level hierarchical network topology generated by BRITE [24] simulator and nothing is said about how IP addresses were assigned to the nodes. Lack of such information does not allow a reader to evaluate reliability of the results achieved.

Another recent work on evaluating proximity between nodes by measuring lengths of their common IP prefixes is presented in [25]. More specifically, the authors calculate

how many IP octets two nodes have in common in order to estimate the relative physical distance between these nodes. They apply this approach to Gnutella, an unstructured p2p system. A node gains knowledge about live overlay nodes that have similar IP addresses by analysing headers of various overlay messages that it receives and requesting such information from bootstrap servers. However, there are no guarantees that a node will eventually find other nodes sharing a common IP prefix.

A p2p overlay that makes use of IP prefixes to estimate the distance between nodes and cluster them accordingly is presented in [26]. In the network-aware clusters, distinguished superpeers enable other peers to discover resources in an unstructured fashion. The weakness of this approach lies in the fact that the bootstrap peer maintains the cluster routing table and thus becomes a single point of failure. In addition, it is assumed that this node is able to download BGP routing tables from nearby routers, whereas typically BGP routers cannot be accessed by unprivileged users.

III. IP-BASED CLUSTERING

IP-based clustering (IPBC) is a proximity neighbour selection technique that makes use of the longest common IP prefix length as a measure of proximity among neighbours. Such kind of proximity information, i.e. IP addresses of nodes participating in the overlay implementing the DHT abstraction, can be conveniently stored in a decentralized manner in the overlay itself, being advertised and discovered in the same way as any other resource.

Each node, characterized by a unique identifier *ID* and an IP address *IP*, keeps track of its neighbours as well as object advertisements it is responsible for. To advertise itself in an overlay, a node first generates a key by hashing a fixed-length prefix of its *IP* (or of its gateway IP address if a node has an address from a private pool [27]) and stores both *ID* and *IP* in DHT using this key. This process is outlined in Figure 1 where the node 6 stores advertisements of the nodes 2 and 3, whereas the node 0 – of 6 and 0.

Consequently, all nodes that share the same IP prefix can be easily found by querying the DHT for the appropriate key. In Figure 2, the node 5 requires a neighbour for the top-left entry of its neighbour table. Instead of simply selecting one of the nodes with suitable identifiers (0,2,3), it first queries DHT for a node that should be close – sharing the same IP prefix. The answer it gets from DHT implies that the node 0 is the best candidate neighbour.

Nodes may query DHT for close neighbours not only when they join an overlay, but at any time during their lifespan. Usually, this would be a part of the reaction to a failure of some neighbour.

If there are more than one near nodes that fit in a given position of a node's neighbours table, then the selection of one of them is made at random. Such procedure prevents

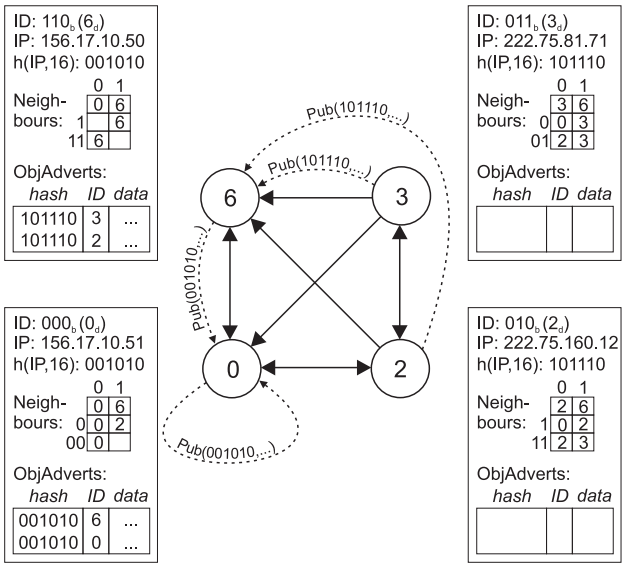

Figure 1. Advertising nodes in DHT

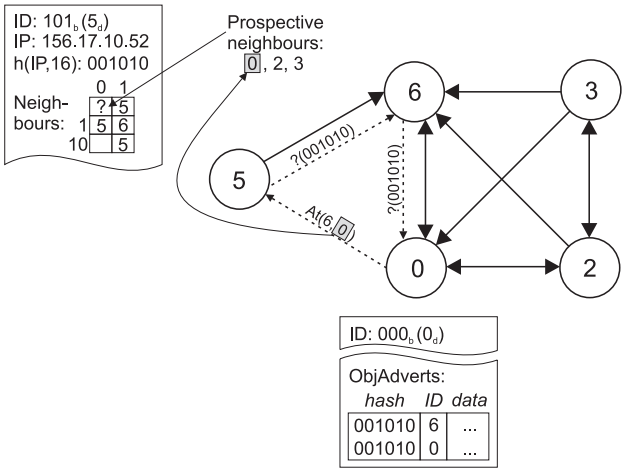

Figure 2. Discovery of nodes with required IP prefixes

a single node from becoming overloaded by being chosen as a neighbour of too many nodes.

The freshness of IP data stored in DHT is taken care of by the advertising nodes, periodically renewing their advertisements. When they voluntarily leave the overlay, they explicitly remove their data. If they fail, their advertisements eventually expire and are removed.

Regarding the choice of the prefix length, the negative impact of too long or too short prefixes is obvious. In the first case, the probability of finding any near neighbours is low, and in the second case, nodes responsible for given prefixes are overloaded. We are opting for the prefix length of 16 bits for Internet-wide systems and further elaborate our choice in the next section.

IV. VERIFICATION OF THE APPROACH

The relation between an IP address of a node and its location in the network topology stems from the organization of the Internet: nodes in the same sub-network share the same IP prefix. Consequently, the relation between

IP addresses of two nodes and network latency between them is quite intuitive: it is more probable that two nodes that have the same IP prefix are close to each other and thus can communicate with low latency. It is obvious that it holds for nodes located in one building, within a small sub-network, all sharing the same IP prefix longer than 24 bits. [28] reports that about 97% of prefixes longer than 24 bits belong to IP addresses at a single geographic location. Observing how the first octet values are distributed across the world [29], it could be expected that the relation between common prefixes of length 1-7 and latency between respective nodes is mostly random. However, among the first octets of IP addresses assigned by Internet Assigned Numbers Authority (IANA) to continent-wide Regional Internet Registries (RIRs) – there are some blocks of consecutive /8 prefixes assigned to the same RIRs and thus it is probable that some dependencies may exist.

In the following sections, we attempt to establish a relationship between prefixes and the communication latency between nodes sharing such common prefixes. First, the data sets used for verification are introduced, their representativeness is evaluated, and their analysis with regards to LCPL-latency relationship is performed.

A. Data Sets

Three large data sets containing measurements and estimations of latency between arbitrary Internet hosts were analysed. We called them `p2psim`, `DIMES`, and `S3` after the names of projects they come from. All of them were preprocessed in order to remove unreliable values, measurements that involved nodes with IP addresses from private pools, as well as to calculate LCPLs. Furthermore, multiple latency measurements for the same pairs of nodes were replaced by their medians.

The first data set we used was `p2psim`, consisting of about 10^8 latency samples measured for pairs of 1740 Gnutella nodes using the King method [9]. This method estimates RTT between arbitrary two Internet hosts by estimating RTT between their DNS servers using recursive queries. The methodology of measurements is detailed in [30]. The data set was collected in 2004 .

The `DIMES` data set was provided by the DIMES project [31] aiming to study the structure and topology of the Internet. The DIMES agents, distributed in a community of volunteers from all over the world, co-operate similarly to the pattern introduced by SETI@Home [32], performing Internet measurements such as traceroute and ping at a low rate. The data set was collected in 2006 and contains 6.5×10^6 measurements.

The `S3` data set contains over 2×10^5 of latency estimations between pairs of PlanetLab nodes. The estimations were generated by Netvigator [33], a landmark-based network latency estimation tool. For over 2000 pairs, latencies were both estimated and measured. Comparison of the results shows that Netvigator is quite precise: the correlation coefficient between measured and estimated latencies equals 0.97. The data set was collected in 2006.

B. Representativeness

1) Selection of Nodes: All approaches to collecting latency measurements intended to utilize Internet nodes selected at random. But did they succeed? To verify if the nodes were selected at random or not, we use the report [34] and its statistics, denoted below as `Bei06`, generated on 19 July, 2006 about the distribution of IPv4 addresses among RIRs.

Moreover, using the report [29] we had prepared a classification schema that allowed us to determine the respective RIR of a given IP address using the first octet as an indicator. Unfortunately, this schema attributes about 17% of first IP octets to the general class called *Various Registries*, where pools of IPs from diverse RIRs can be found. In our RIR-based classification, we are ignoring addresses beginning with these octets.

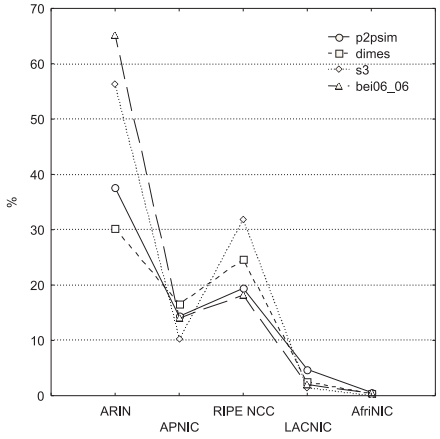

Figure 3. Distribution of IP addresses among RIRs

In Figure 3, we compare the results of classification with `Bei06`. Regarding `DIMES` and `p2psim`, the analysis shows that for all RIRs but ARIN, the distribution of IPs from the data sets closely resembles the distribution of `Bei06`. In case of ARIN, we speculate that pools of IP addresses governed by this RIR are in the overwhelming majority of *Various Registries* classes and this is the possible reason for their underestimation. Summing up, we claim that the nodes in both data sets were selected at random with high probability: The distributions of IPs that were generated using different random processes are very similar.

In the `S3` data set, as opposed to the others, all IP addresses could be assigned to particular RIRs. Consequently, the distribution should closely resemble the one of `Bei06`. And indeed it does, but with an interesting exception: It reports that the number of addresses allocated to ARIN is about 10% smaller whereas this number is accordingly larger in case of RIPE NCC. This finding shows that the distribution of IP addresses of PlanetLab members among RIRs do not precisely correspond to the analogous distribution in the Internet. Nevertheless, the trend is coincident with `Bei06`.

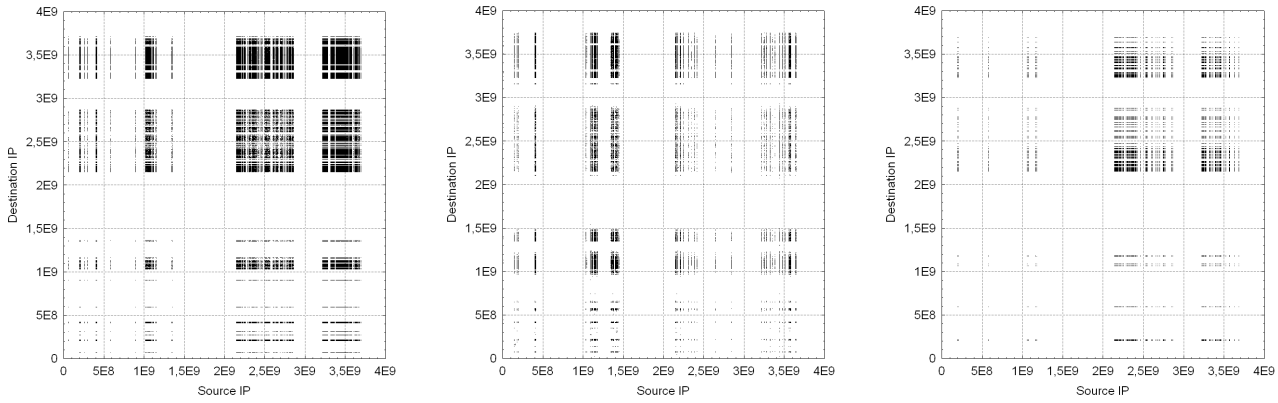

Figure 4. Distribution of measurements between pairs of nodes: (a) p2psim (b) DIMES (c) S3.

2) Selection of Node Pairs: Latencies were measured or estimated for pairs of nodes. We would like to verify if the pairs were selected evenly. If they were not, the relationship between LCPL and latencies could be obscured. Namely, if there are nodes that prefer measuring latencies only to nodes that have IP prefixes similar to (or different from) their own ones, then this would skew the distribution of latencies for such LCPLs. The distributions of measurements between pairs of nodes are shown in figure 4 where the IP addresses are represented as 32-bit numbers, e.g.: $128.0.0.0 = 2147483648(d) \approx 2,15E9$

The latencies from `p2psim` and `S3` were collected by estimating latencies between nodes A and B as well as B and A. Hence, the measurements in figure 4 (a,c) are distributed diagonally symmetric. The `DIMES` latencies were measured by source nodes with no guarantee that destination nodes will perform the inverse measurement. Thus the diagonal symmetry cannot be observed in figure 4 (b), but this difference does not affect our reasoning.

Looking at the graphs, it can be observed that in all cases, for each source IP there are sets of almost the same destination IPs selected. In other words, there are no nodes that would prefer measuring latencies only to nodes that have IP prefixes similar to (or different from) their own. The gaps in the figures are caused by blocks of reserved and special use IP addresses.

3) Distribution of Latencies: The characteristics of latencies in the Internet have been measured and estimated a number of times, using different methods [13], [35]–[39]. The distributions of latencies for the data sets are drawn in figure 5.

Distributions of latencies in `p2psim` and `DIMES` are two-modal and long-tailed. However, in case of `DIMES`, both modes are observed clearly and equally often whereas in case of `p2psim`, the leftmost mode is about 15% more frequent than the rightmost, hardly formed one. We attribute this difference to a slightly diverse distributions of IP addresses from the data sets among RIRs (see figure 3). Summarizing, the two distributions are strongly correlated and follow those found in the literature, e.g. [13], [36].

The distribution of latencies in `S3` varies from the

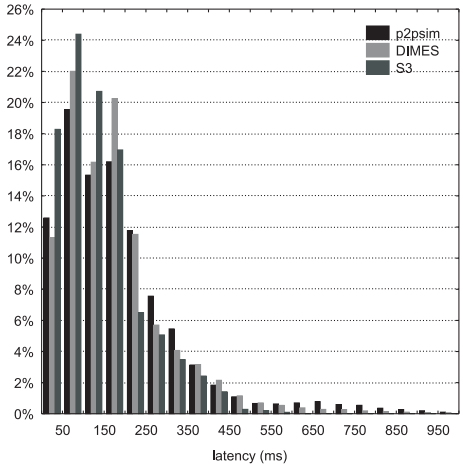

Figure 5. Distribution of latencies

others: it is one-modal and rather short-tailed. Furthermore, its median latency is smaller than in other data sets. These discrepancies originate from the properties of the PlanetLab network, built by universities and international companies owning high-speed network connections, incomparable to those connecting the household computers.

4) Discussion: Recapitulating, the `DIMES` data set appears to be the most reliable: it is up-to-date, founded on the latency measurements, and characterized by latency distribution that is close to that of the Internet. The reliability of `p2psim` is slightly worse: it is older and based on latency estimations. Nevertheless, the respective distribution is quite similar to that of `DIMES`. The `S3` data set is the least representative: even though very recent, it relates to a subset of Internet nodes that are much better connected and more powerful than a majority of computers communicating via the Internet. However, we do not exclude this data set from further analysis since it may be of interest to other researchers to see if and to what extent the relation between IP prefixes and inter-node latencies also holds in this popular testbed for distributed applications.

C. Analysis

The number of samples per LCPL in the pre-processed data sets is shown in figure 6.

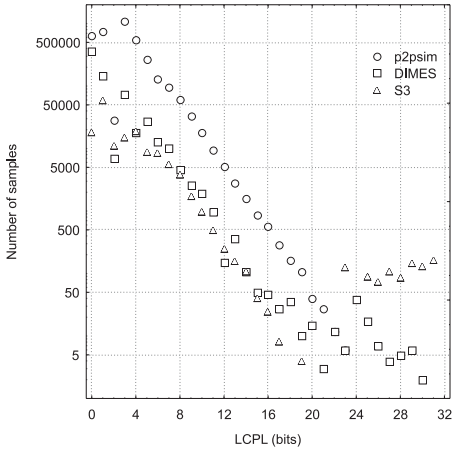

Figure 6. Number of samples in data sets

For all data sets, an unusual decrease in the number of samples for $LCPL = 2$ may be observed. We attribute this anomaly to the pools of reserved and multicast octets [29]. Namely, two IP addresses have common prefix of length 2 if their first octets belong to pools (a) 128-159 and 160-195 or (b) 196-227 and 228-255 respectively. The option (b) is completely excluded by the existence of a block of special use octets: 223-255 and another option is limited by the block of reserved octets 173-187.

It may be expected that the number of samples decreases exponentially as LCPL grows. It proves true for `p2psim` and `DIMES`; in case of `S3`, such a trend is preserved for $LCPL \leq 19$. For larger values of LCPL, the number of samples is approximately constant and definitly exceeds those from the other data sets. The explanation of this phenomenon lies, once again, in the organization of PlanetLab: participating institutions are obliged to donate not a single machine, but a set of computers for the PlanetLab community.

1) p2psim: The relation between the longest common IP prefix length of a pair of nodes and median latency for `p2psim` is presented in figure 7(a), revealing that median latencies are strongly correlated with length of the longest common prefix within the 8-24 bits range: for this range, the correlation coefficient equals -0.95 (for the whole range: -0.88). Furthermore, three sub-ranges can be easily identified such that have characteristic median latencies, 75, 45, and 12 milliseconds respectively.

The variability of latencies is drawn in the same figure as the 25th and 75th percentile. It decreases as LCPL grows in the range 0-18 bits; for $LCPL > 18$, it stabilizes.

2) DIMES: The LCPL-latency relationship for the `DIMES` data set is shown in figure 7(b). The correlation coefficient for the whole range of LCPL is high and equals -0.79. The grouping effect of median latencies, as seen in figure 7(a), can be also identified in such subranges of LCPL as 6-14, 15-18, and 25-29 bits. In the range 19-24 bits, the distribution of quartiles appears random and we point out two possible reasons for this:

- There were very few samples for $LCPL > 18$ and it was possible that the distribution for such LCPLs could be dominated by measurements performed within a single (or few) /19-/23 network. For example, in case of $LCPL = 23$, 33% of samples were collected from nodes belonging to the same network with dial-up Internet access.
- The observed randomness could be the effect of CIDR IP addresses allocation.

Some of the quartile ranges for long common IP prefixes stand out as unusually wide (e.g. $LCPL = 23$) and it suggests the influence of the first of the reasons listed above. In case of LCPLs between 19 and 22, the influence of CIDR on the latency distribution seems to be more probable.

3) S3: Figure 7(c) presents the relation between LCPL and median latency. It reveals strong correlation between them: the correlation coefficient equals -0.85. Moreover, three sub-ranges of LCPL can be easily identified such that have characteristic median latencies, 52, 35, and 0.8 milliseconds respectively. The variability of latencies decreases as LCPL grows. For $LCPL > 26$, it practically disappears.

4) Discussion: The probable reason of grouped latencies in figures 7(a-c) is geographic distribution of IP (sub)networks. Intuitively, we presume that sub-networks sharing 6-7 bits are specific to continents, 8-15 bits to countries, 16-18 bits to cities, and 19 bits and more to particular organizations (LANs). These are approximate ranges and they differ slightly between the analysed data sets. Nevertheless, such reasoning can also be justified based on [40], where the relation between geographic distance and latency is investigated.

The analysis of the `DIMES` data set suggests that the influence of CIDR within the /19-/22 range of LCPL is substantial and disturbs the relationship presented in the paragraph above. Nonetheless, the latencies for LCPL values belonging to this particular range are in-between those specific to countries and cities.

In figure 7(a), the correlation between LCPL and latency is the highest from among all data sets and the quartile ranges are very regular, without outliers. We attribute these almost ideal results to the method of estimations that produced the `p2psim` data set: the King method estimates inter-node latencies by measuring latencies between the nodes' DNS servers. As opposed to `DIMES` and `S3`, it completly obscured the influence of nodes with poor Internet connections or those belonging to small, geographically distributed CIDR blocks on the LCPL-latency relationship. On the other hand, it shows that in case of DNS servers, this relationship is exceptionally strong.

Coming back to our discussion regarding the best length of IP prefix that could be used as a key in the IPBC method deployed world-wide, we claim that 16 bits would be a good choice (table I). It ensures low inter-node

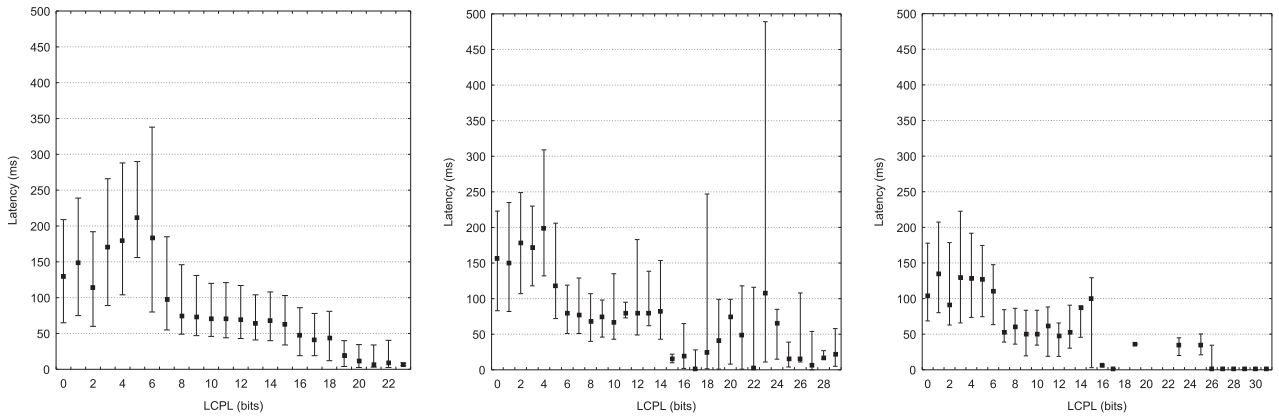

Figure 7. Quartile ranges: (a) p2psim (b) DIMES (c) S3.

latency and does not embrace too many nodes. However, if a storage space is not an issue and, on the other hand, it is preferable to have larger groups, the selection of a shorther prefix might be a better choice.

	Latency		% of samples
	median (ms)	75th perc. (ms)	
p2psim	47	86	0.03%
DIMES	19	65	0.03%
S3	6	7	0.62%

TABLE I.
INTER-NODE LATENCIES AND NUMBER OF NODE PAIRS FOR LCPL=16 BITS

Summing up, the analysis supports our claim that the IP-based measure of proximity used in IPBC can be successfully utilized for the neighbour selection algorithms in DHTs. Regarding the selection of longest common IP prefix length, 16 bit prefix ensures low latencies and discovery of physically very close nodes. However, the trade-off between median inter-node latency in a group, group size, and a storage space required for storing advertisements of prefixes in DHT should be considered before applying IPBC to a particular system.

V. CONCLUSION

We have devised a novel method of self-optimization for peer-to-peer overlays deployed in large and dynamic environments that uses only static information about existing nodes in the overlay network, i.e. their IP addresses.

Our approach is an improvement over current methods since it eliminates the need for costly probing of latencies between possible neighbours. Also, this information - being stored in the overlay network itself - is ready for use by new nodes, lowering the cost of the joining procedure, because the new nodes need not probe the latencies of the potential neighbours.

We established a correlation between the longest common prefix of IP addresses of communicating nodes and network latency by analysis of two large, representative sets of latency measurements on the Internet. The correlation suggests that our approach can be used for neighbour selection in order to better match the overlay and IP network topologies, thus reducing latencies.

ACKNOWLEDGMENT

The authors are grateful to the DIMES team, the researchers of p2psim, and HP Labs NAPA research group for the collection of latency measurements between pairs of Internet nodes.

REFERENCES

[1] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, "Chord: A scalable peer-to-peer lookup service for internet applications," in *Proceedings of SIGCOMM '01*. ACM Press, 2001, pp. 149–160.

[2] A. Rowstron and P. Druschel, "Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems," in *Proceedings of IFIP/ACM Middleware 2001*, November 2001.

[3] B. Zhao, L. Huang, J. Stribling, S. Rhea, A. Joseph, and J. Kubiatowicz, "Tapestry: A resilient global-scale overlay for service deployment," *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 1, pp. 41–53, 2004.

[4] P. Maymounkov and D. Mazieres, "Kademlia: A peer-to-peer information system based on the xor metric," in *Proceeding of 1st International Workshop on Peer-to-peer Systems (IPTPS'02)*, 2002.

[5] J. Močnik, M. Novak, G. Pipan, and P. Karwaczynski, "A discovery service for very large, dynamic grids," in *Second IEEE International Conference on e-Science and Grid Computing (e-Science'06)*. Washington, DC: IEEE Computer Society, December 2006.

[6] K. Hildrum, J. D. Kubiatowicz, S. Rao, and B. Y. Zhao, "Distributed object location in a dynamic network," in *SPAA'02: Proceedings of the 14th ACM Symposium on Parallel Algorithms and Architectures*. ACM Press, 2002, pp. 41–52.

[7] M. Ripeanu, I. Foster, and A. Iamnitchi, "Mapping the gnutella network - properties of large-scale peer-to-peer systems and implications for system design," *IEEE Internet Computing Journal*, vol. 6, no. 1, 2002.

[8] B. Huffaker, M. Fomenkov, D. Plummer, D. Moore, and K. Claffy, "Distance metrics in the internet," in *Proceedings of the IEEE International Telecommunications Symposium*, 2002.

[9] K. P. Gummadi, S. Saroiu, and S. D. Gribble, "King: estimating latency between arbitrary internet end hosts," *SIGCOMM Comput. Commun. Rev.*, vol. 32, no. 3, pp. 11–11, 2002.

[10] J. Hawkinson and T. Bates, "RFC 1930: Guidelines for creation, selection, and registration of an autonomous system (AS)," IETF, 1996. [Online]. Available: http://www.ietf.org/rfc/rfc1930.txt
[11] IANA, "Autonomous system numbers," IANA, April 2006. [Online]. Available: http://www.iana.org/assignments/as-numbers
[12] V. Fuller, T. Li, J. Yu, and K. Varadhan, "RFC 1519: Classless Inter-Domain Routing (CIDR): an address assignment and aggregation strategy," IETF, 1993. [Online]. Available: http://www.ietf.org/rfc/rfc1519.txt
[13] R. Gummadi, S. Gribble, S. Ratnasamy, S. Shenker, and I. Stoica, "The impact of dht routing geometry on resilience and proximity," in *Proceedings of SIGCOMM '03*, 2003.
[14] F. Dabek, J. Li, E. Sit, J. Robertson, M. F. Kaashoek, and R. Morris, "Designing a dht for low latency and high throughput," in *Proceedings of USENIX Symposium on Networked Systems Design and Implementation*, 2004.
[15] T. Locher, S. Schmid, and R. Wattenhofer, "equus: A provably robust and locality-aware peer-to-peer system," in *Proc. 6th IEEE Int'l Conf. on Peer-to-Peer Computing (P2P'06)*, 2006.
[16] S. Ratnasamy, M. Handley, R. Karp, and S. Shenker, "Topologically-aware overlay construction and server selection," in *Proceedings of IEEE INFOCOM'02*, 2002.
[17] R. Tian, Y. Xiong, Q. Zhang, B. Li, B. Y. Zhao, and X. Li, "Hybrid overlay structure based on random walks," in *Proceedings of 4th International Workshop on Peer-to-Peer Systems (IPTPS)*, 2005.
[18] Q. Zhang, F. Yang, W. Zhu, and Y. Q. Zhang, "Construction of locality-aware overlay network: moverlay and its performance," *IEEE Journal on Selected Areas in Communications (J-SAC)*, vol. 22, no. 1, Jan 2004.
[19] L. Garcs-Erice, K. W. Ross, E. W. Biersack, P. Felber, and G. Urvoy-Keller, "Topology-centric look-up service," in *Proceedings of the 5th International Workshop NGC '03*, 2003.
[20] R. A. Ferreira, A. Grama, and S. Jagannathan, "Enhancing locality in structured peer-to-peer networks," in *Proceedings of the 10th IEEE International Conference on Parallel and Distributed Systems (ICPDS)*, 2004.
[21] B. Krishnamurthy and J. Wang, "On network-aware clustering of web clients," in *Proceedings of SIGCOMM '00*. ACM Press, 2000, pp. 97–110.
[22] L. Garces-Erice, K. W. Ross, E. W. Biersack, P. Felber, and G. Urvoy-Keller, "Topology-centric look-up service," in *Proc. 5th Int'l Workshop on Networked Group Communications (NGC'03)*, 2003.
[23] J. Xiong, Y. Zhang, P. Hong, and J. Li, "Chord6: IPv6 based topology-aware chord," in *Proc. IEEE Int'l Conf. on Autonomic and Autonomous Systems / Int'l Conf. on Networking and Services (ICAS/ICNS)*, 2005.
[24] A. Medina, A. Lakhina, I. Matta, and J. Byers, "BRITE: An approach to universal topology generation," in *Proc. Int'l Workshop on Modeling, Analysis and Simulation of Computer and Telecommunications Systems (MASCOTS'01)*, August 2001.
[25] J. Zhao and J. Lu, "Solving overlay mismatching of unstructured p2p networks using physical locality information," in *Proc. 6th IEEE Int'l Conf. on Peer-to-Peer Computing (P2P'06)*, 2006.
[26] C.-M. Huang, T.-H. Hsu, and M.-F. Hsu, "Network-aware p2p file sharing over the wireless mobile networks," *IEEE Journal on Selected Areas in Communications (J-SAC)*, vol. 25, no. 1, Jan 2007.
[27] Y. Rekhter, B. Moskowitz, D. Karrenberg, G. J. de Groot, and E. Lear, "RFC 1918: Address allocation for private internets," IETF, 1996. [Online]. Available: http://www.ietf.org/rfc/rfc1918.txt
[28] M. Freedman, M. Vutukuru, N. Feamster, and H. Balakrishnan, "Geographic locality of ip prefixes," in *Proceedings of Internet Measurement Conference (IMC)*, 2005.
[29] IANA, "Ipv4 address space," IANA, January 2006. [Online]. Available: http://www.iana.org/assignments/ipv4-address-space
[30] F. Dabek, R. Cox, F. Kaashoek, and R. Morris, "Vivaldi: a decentralized network coordinate system," in *Proceedings of SIGCOMM '04*. ACM Press, 2004, pp. 15–26.
[31] T. D. Team, "DIMES: Distributed Internet Measurements & Simulations," Evergrow. [Online]. Available: http://www.netdimes.org/
[32] D. P. Anderson, J. Cobb, E. Korpela, M. Lebofsky, and D. Werthimer, "SETI@home: an experiment in public-resource computing," *Commun. ACM*, vol. 45, no. 11, pp. 56–61, 2002.
[33] Z. Xu, P. Sharma, S. J. Lee, and S. Banerjee, "Netvigator: Scalable network proximity estimation," HP Labs, Tech. Rep. HPL-2004-28R1, 2006.
[34] I. van Beijnum, "Ipv4 address use report," January 2006. [Online]. Available: http://www.bgpexpert.com/addrspace2005.php
[35] A. Mukherjee, "On the dynamics and significance of low frequency components of internet load," *Internetworking: Research and Experience*, vol. 5, pp. 163–205, December 1994.
[36] M. E. Crovella and R. L. Carter, "Dynamic server selection in the internet," in *Proceedings of the 3rd IEEE Workshop HPCS '95*, 1995.
[37] A. Acharya and J. Saltz, "A study of internet round-trip delay," University of Maryland, Tech. Rep. CS-TR-3736, December 1996.
[38] Y. Zhang, "Characterizing end-to-end internet performance," Ph.D. dissertation, Cornell Univ., August 2001. [Online]. Available: http://www.cs.cornell.edu/yzhang/papers/thesis.ps.gz
[39] H. Zhang, A. Goel, and R. Govindan, "An empirical evaluation of internet latency expansion," *SIGCOMM Comput. Commun. Rev.*, vol. 35, no. 1, pp. 93–97, 2005.
[40] G. Sirer, "Network positioning for wide-area and wireless networks," in *Proceedings of the International Symposium on Distributed Computing*, 2005.

Piotr Karwaczyński received his MSc degree in computer science at the Wrocław University of Technology in 2003 with distinction (specialization: Software Engineering) and currently he is working on his PhD thesis in the field of decentralized systems.

In 2003, he joined the Institute of Applied Informatics, Wrocław University of Technology, as a research assistant. Since October 2003, he has been participating in the EU FP6 project "Dependable Distributed Systems". In 2006 he initiated the cooperation of the university with the international PlanetLab consortium.

Jaka Močnik received his BSc degree in computer science at the University of Ljubljana in 2001 (thesis: Load Distribution in a CORBA Environment). Currently, he is finishing his MSc thesis in the area of dependability of distributed systems.

Since 2005, he is working at XLAB Research, dividing his time between applied and basic research in the area of distributed systems, in particular service-oriented architectures, Grid systems and peer-to-peer networks. Since October 2003, he has been participating in the EU FP6 project "Dependable Distributed Systems."