On the Aging of Landmark-based Coordinates

Xiaoming Zhou and Piet Van Mieghem
Faculty of Electrical Engineering, Mathematics, and Computer Science
Delft University of Technology
P.O. Box 5031, 2600 GA Delft
Email: {X.Zhou, P.VanMieghem}@ewi.tudelft.nl

Abstract— **Many large-scale distributed network applications could benefit from an efficient mechanism to estimate distance in the Internet. Landmark schemes provide such estimates based on distances in a hyperspace in which the hosts are embedded. In this paper, we evaluate the aging effect of landmark-based coordinates using empirical data. That is, to what extent are past measurements a good predictor to estimate the current distance. Our experiments show that based on the data up to 7 days old, the predicted hyperspace-distance function using "triangulation heuristics" correlate relative highly (more than 85%) with the measured distance.**

I. Introduction

An efficient mechanism to estimate distance in the Internet could be beneficial for many large-scale distributed network applications such as nearby server selection and peer-to-peer computing. Landmark schemes provide such estimates based on distances in a hyperspace in which the hosts are embedded [1][4]. In these techniques each of the N nodes measures its distance to a small set of M well-known landmark servers. By conceiving the results as the components of a vector, each node is embedded in a M-dimensional hyperspace. It is assumed that the distance between two network nodes can be estimated by computing the (e.g. Euclidean) distance between their respective coordinate vectors in the hyperspace.

In this paper, we investigate the aging of landmark-based coordinates. Because hosts are not going to continuously update their coordinates, it is interesting to investigate the quality of prediction based on coordinates which have been determined earlier, and to know how often peers should update their coordinates. This knowledge might enhance our ability to manage the network distance estimation.

In this study, "distance" is defined in terms of the network delay experienced in a packet exchange. The network delay directly influences the user experience in many applications. There can be an indirect influence as well because the TCP throughput is roughly inversely proportional to the round-trip time. Our empirical evaluation is based on data from the RIPE NCC TTM project [10]. This data-set contains measurements of the delay between a set of about 80 test-boxes.

The remainder of this paper proceeds as follows. After a discussion of related work in Section 2, Section 3 describes the problem. Section 4 reports on the experimental results. Finally, Section 5 summarizes our conclusions.

II. Related work

Several landmark-based methods have been proposed to estimate the distance between hosts on the Internet. However, none of them investigated the aging effect of landmark-based coordinates.

GeoPing was originally intended to infer geographical locations of nodes from delay measurements to a set of landmarks [1], but later it was also used to infer the delay itself [2]. In the latter form it coincides with the method studied by us, except that it only uses the Euclidean hyperspace distance. A main conclusion of our work [9] is that for predicting the delay, the Euclidean distance function is not the optimal choice. A minor difference is that GeoPing uses the median delay, whereas we use the minimum delay.

Global Network Positioning (GNP) [7] is a different estimation scheme which is also based on landmarks. In this approach not only the peers but also the landmarks are embedded in the hyperspace, based on the measured inter-landmark distances. The assignment of coordinates to landmarks and other nodes is cast in the form of a minimization problem where the objective function quantifies the errors in the distance estimates. This minimization should be solved online, which imposes a computational load on nodes and landmarks. The authors provide an extensive comparison between GNP, the "triangulation heuristics" D_+, D_∞, and D_A that we consider in this paper, and IDMaps (see below). They concluded that D_+ is the best triangulation heuristic for the delay. This is confirmed by our analysis [9]. Lighthouse [5] and SCoLE [6] improve on GNP in the sense that the role of the landmarks is progressively decentralized.

IDMaps [3] is an infrastructure for estimating distances. Their "tracers" are servers similar to our landmarks, except that they actively measure their distances towards other tracers and towards domains on the Internet.

Various overlay networks use distance estimates to construct efficient topologies. S. Ratnasamy *et al.* [8] describe the use of a landmark scheme in a distributed hash-table named CAN.

In our previous work [9], we have evaluated a simple landmark scheme using empirical data for both delay and hopcount. We investigated various choices for the hyperspace distance function. We found that the delay is easier to predict than the hopcount.

III. Problem description and definitions

In this paper, we study the hyperspace distance[1] to estimate the network distance. Let $d(A, B)$ be the network distance between node A and B. Note that our "network distance" is not a mathematical distance function. In particular the triangle inequalities may not hold.

The M landmark servers are denoted as $L_1, ..., L_M$. To each node X_i we assign a coordinate vector

$$\boldsymbol{x}_i \equiv (\boldsymbol{x}_{i1}, \boldsymbol{x}_{i2}, ..., \boldsymbol{x}_{iM})$$

where $\boldsymbol{x}_{ik} = d(X_i, L_k)$ is the measured distance between the node i and the landmark k. The hyperspace distance between the coordinate vectors $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ is written as $D(\boldsymbol{x}_i, \boldsymbol{x}_j)$, to distinguish it from the network distance d. We consider the correlation $D(\boldsymbol{x}_i, \boldsymbol{x}_j)$ with $d(X_i, X_j)$ as the measure of quality of the prediction scheme. We study the same functionals as in our previous work [9]:

$$\begin{aligned}
D_q(\boldsymbol{x}_i, \boldsymbol{x}_j) &= \left(\sum_{k=1}^{M} (\boldsymbol{x}_{ik} - \boldsymbol{x}_{jk})^q\right)^{1/q}, q > 0 \\
D_+(\boldsymbol{x}_i, \boldsymbol{x}_j) &= \min_{k=1,...,M} (\boldsymbol{x}_{ik} + \boldsymbol{x}_{jk}) \\
D_A(\boldsymbol{x}_i, \boldsymbol{x}_j) &= (D_+ + D_\infty)/2 \\
D_G(\boldsymbol{x}_i, \boldsymbol{x}_j) &= \sqrt{D_+ D_\infty}
\end{aligned}$$

Their linear correlation coefficients of the predicted hyperspace distance with actual network distance d are defined by

$$r(D, d) = \frac{cov(D, d)}{\sqrt{var(D) \cdot var(d)}}$$

When using the hyperspace distance $D(t)$ with the data obtained at time t to estimate the network distance $d(t + \Delta t)$ at time $t + \Delta t$ (Δt is called a time lag), the linear correlation coefficients of the predicted hyperspace distance $D(t)$ with the network distance $d(t + \Delta t)$ are defined by

$$\rho(t, \Delta t) = r(D(t), d(t + \Delta t))$$

IV. Experiment results

Our data are provided by RIPE NCC TTM project. The TTM infrastructure consists of approximately 80 measurement boxes scattered over Europe (and a few in the US and Asia). Between each pair of measurement boxes, IP packets of a fixed length (100 bytes), called probe-packets, are continuously transmitted with interarrival times of about 30 seconds, resulting in a total of about 2880 probe-packets per day. The sending measurement box generates an accurate time-stamp synchronized via GPS in each probe-packet, while the receiving measurement box reads the GPS-time of the probe-packet on arrival. The end-to-end delay is defined as the difference between these two time-stamps and has an accuracy of about 10 μs. We have analyzed the data collected by TTM from February 8, 2004 to February 14, 2004, and the data collected on May 15, 2003 and on January 31, 2004. We only considered the minimal delays available in all the days, resulting in 73 active boxes, where 62 hosts are located in Europe, 7 in the US, 2 in Japan, and 1 each in Australia and New Zealand.

For each sender-destination pair we computed the minimum end-to-end delay over 24 hours (that is approximately 2880 probe-packets), in order to know the congestion-free delay. The RIPE TTM differs from other infrastructures like PingER and AMP in that it measures one-way delays rather than RTTs. In this paper, however, we do not care about asymmetry, and we always consider the symmetrized network distance, defined as the sum of the network distances in both directions. For the delay, the result can be considered as a round-trip time. We omitted pairs for which the delay in one of the directions is missing, leaving in total 4521 pairs. These measurements provide us with the network distance matrix $d(A, B)$.

In the experiment during more than a week period, 10 of all the test-boxes are assigned the status of landmark. We choose them randomly except that 6 are located in the EU, 1 is located in the Asia and 3 are located in the US. The remaining test-boxes are referred to as “peers”. We repeated the experiments with $M = 5$ and $M = 15$ landmarks respectively. In the following part A, we investigate the quality of a landmark scheme over one week, while in the part B, we study the time dependence of coordinates.

A. The quality of a landmark scheme over one week

This part investigates the quality of a landmark scheme over a week. We consider the measurement data that spans a week from Monday (February 8, 2004) to Sunday (February 14, 2004). During each day of the experiment, for each pair of peers, the current network distance $d(A, B)$ was calculated and was compared to the distance predicted by the landmark scheme (using 5, 10 and 15 landmarks): that is the hyperspace distance $D(\boldsymbol{x}_A, \boldsymbol{x}_B)$ for six different hyperspace norms: D_1, D_2, D_∞, D_+, D_A, and D_G. Figure 1 plots the correlation coefficients $\rho(t, 0)$ for the estimated (i.e. hyperspace) distance versus the measured distance at the same day, for the six hyperspace distance functions. On the vertical axis gives the linear correlation coefficient $\rho(t, 0)$, on the horizontal axis is the day.

Note that due to the dynamic property of the Internet distance, $\rho(t, 0)$ for six hyperspace distance functions are not always the same during a week. The results in Figure 1 show that during a week, the best results for the delay estimation are always obtained using D_G and D_+, and they seem to possess a relatively stable correlation over time. The difference between the results obtained using D_G and D_+ is only within 2%. Moreover, the D_G and D_+ hyperspace distances give the highest correlations, in particular higher than for the Euclidean distance D_2. These analysis confirm the conclusion in [9], which is based on data measured 9 months earlier (May 15, 2003).

[1] A relatively new approach to represent a network distance matrix is to map the network nodes into a k-dimensional (e.g. Euclidean) hyperspace. In some papers (like [11]), the term "embedding" is also used to denote this hyperspace mapping.

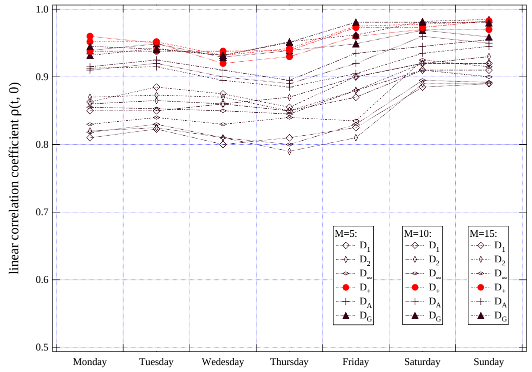

Fig. 1. The correlation coefficients for the estimated (i.e. hyperspace) distance versus the measured distance, for six hyperspace distance functions.

B. Estimate the distance using the past data

Hosts are not going to continuously update their coordinates. In this part, we investigate the quality of prediction based on coordinates determined earlier. The central question addressed in this part is: how often should peers update their coordinates? More precisely, with a time lag between embedding and evaluation, how does the hyperspace distance function correlate with the actual distances between hosts?

To answer above questions, we took the measured distance on February 14, 2004 as measured distance $d(A, B)$. We predicted the distances based on data measured on earlier time (for six hyperspace distance functions). For each peer, the correlation coefficients $\rho(t, \Delta t)$ for the estimated (i.e. hyperspace) distance versus $d(A, B)$ was computed. Due to space limitation, Figure 2 (see the last page) only shows an example of scatterplots for the measured distance versus the estimated (i.e. hyperspace) distance based on the data measured in 1 day earlier ($\Delta t = 1$). Each data point corresponds to a pair of peers. On the horizontal axis is their actual network distance $d(A, B)$ measured on February 14, on the vertical axis their distance predicted by the landmark scheme, that is their hyperspace distance $D(\boldsymbol{x}_A, \boldsymbol{x}_B)$, using the previous delay data measured on February 13. The solid lines shown are the least-squares fitted line, and the diagonal $D = d$. The inset also shows the linear correlation coefficient $\rho(t, \Delta t)$.

Note that Internet distance can change due to routing policy, routing updates or changes of the topology. Figure 3 presents the scatterplots of the distance measured on February 14 and some earlier measured distances. The results show that most end-to-end distances are stable in our database. In a number of cases, higher capacity links were chosen due to routing updates between source-destination pairs, which leads to shorter end-to-end delays. Some outliers in Figure 3 show this change. For example, the routings that from a host located in Sofia, Bulgaria to other test-boxes have been completely changed since February 2004, and the one-way delay has been reduced by a factor of 10 times smaller.

For a better visualization, Figure 4 shows the correlation coefficient $\rho(t, \Delta t)$ as a function of the time lag Δt. This

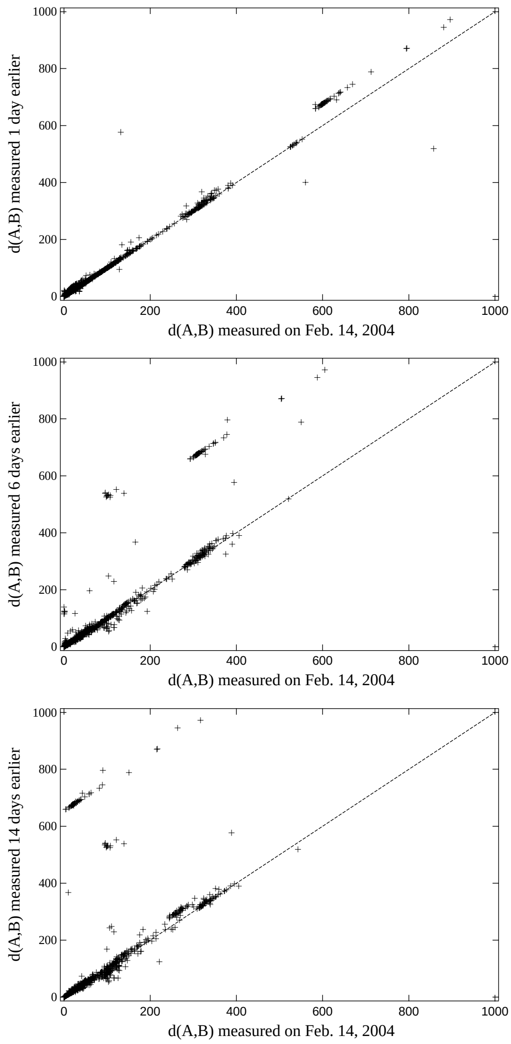

Fig. 3. Scatterplots current distance (measured on Feb. 14, 2004) versus previous measured distance in the database. The plotted line is the diagonal

experiment involves $M = 10$ landmarks. The plot shows that based on the data measured up to a week earlier, the predicted hyperspace-distance function using D_G and D_+ correlate strongly (more than 0.85) with the measured distance, and in particular, exceeds that of the Euclidean distance D_2. The $\rho(t, \Delta t)$ is much lower (about 0.2) for the prediction based on data measured 14 days or more days earlier. We also observed similar results for $M = 5$ and $M = 15$.

The complete results are summarized in Table I, where the first, second and third number in each cell is the linear correlation coefficient $\rho(t, \Delta t)$ for $M = 5$, 10 and 15 landmarks, respectively. Table I conforms the observation that based on the data measured within a week, the predicted hyperspace-distance function using D_G and D_+ correlate strongly with the measured distance.

To measure how well a predicted distance (based on data measured on earlier time) matches the corresponding measured distance, we use two metric relative errors that are defined as:

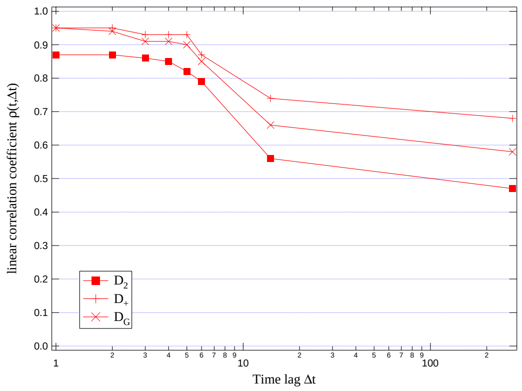

Fig. 4. The correlation coefficient for the measured distance versus the estimated hyperspace distance based on data measured on eariler time for D_2, D_G and D_+. The landmarks is 10.

$e(D_G, d) = |\frac{D_G - d(A,B)}{\min(D_G, d)}|$ and $e(D_+, d) = |\frac{D_+ - d(A,B)}{\min(D_+, d)}|$.

Figure 5 shows the cumulative distribution function (CDF) of the ratios of $e(D_G, d)$ and $e(D_+, d)$ for $M = 10$ for all measurement in our data sets. The closer this ratio to 0, the better the estimation. We observed that based on the data within a week earlier, between 94% and 98% of estimates using D_+ fall within a factor of 1 of the real distances; while 90% and 94% for D_G. However, based on the data measured 9 months earlier, only about 80% of estimates using D_+ (or D_G) fall within a factor of 1 of the real distances. A prediction mechanism can potentially be extremely inaccurate with respect to the relative error metrics, but the prediction scheme using D_G and D_+ based on the past data (e.g. within a week) can be relatively accurate to the measured distance. We also observed similar results for $M = 5$ and $M = 15$.

V. Conclusions

In this paper, we studied how often should peers update their coordinates. More precisely, we investigated the quality of landmark schemes over one week using empirical delay data. Our experiments show that the best results for the delay estimation are obtained using D_G and D_+, and they seem to possess a relatively stable correlation over time. Moreover, the D_G and D_+ hyperspace distances give the highest correlations, and exceed that of the Euclidean distance D_2. We also investigated the quality of prediction based on coordinates which have been determined earlier. Our experimental results suggest that based on data up to 7 days old, the predicted hyperspace-distance function using D_G and D_+ correlate relative highly with the measured distance. Moreover, those predicted hyperspace distance can be relative accurate with respect to the measured distance.

Acknowledgements

We are grateful to Stijn van Langen for very useful discussions and comments. We would also like to thank Henk Uijterwaal, Mark Santcroos and Ruben van Staveren (RIPE) for the use of the RIPE measurement system. Part of this work has been funded by the NWO SAID project.

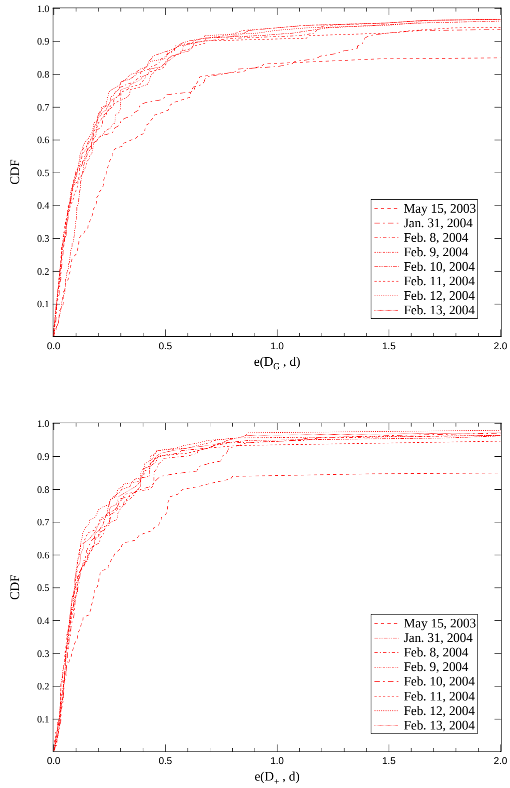

Fig. 5. Cumulative distribution function (CDF) of the ratios of $e(D_G, d)$ and $e(D_+, d)$ for 10 landmarks in our experiment

References

[1] V. N. Padmanabhan and L. Subramanian, *An Investigation of Geographic Mapping Techniques for Internet Hosts*, Proc. ACM Sigcomm, San Diego, CA, USA (2001).

[2] K. Lakshminarayanan and V. N. Padmanabhan, *Network Performance of Broadband Hosts: Measurements & Implications*, Microsoft Technical Report (2003).

[3] P. Francis, S. Jamin, C. Jin, Y. Jin, D. Raz, Y. Shavitt, and L. Zhang, *IDMaps: A Global Internet Host Distance Estimation Service*, IEEE/ACM Transactions on Networking 9, 525 (2001).

[4] J. D. Guyton and M. F. Schwartz, *Locating Nearby Copies of Replicated Internet Servers*, Proc. ACM Sigcomm, Cambridge, MA, USA (1995).

[5] M. Pias, J. Crowcroft, S. Wilbur, T. Harris, and S. Bhatti, *Lighthouses for Scalable Distributed Location*, Proc. of the 2nd International Workshop on Peer-to-Peer Systems, Berkeley, CA, USA (2003).

[6] M. Szymaniak, G. Pierre and M. van Steen, Scalable Cooperative Latency Estimation, Proceedings of the 10th International Conference on Parallel and Distributed Systems (ICPADS2004), Newport Beach, CA, USA, July 2004

[7] T. S. E. Ng and H. Zhang, *Towards Global Network Positioning*, Proc. ACM Sigcomm Internet Measurement Workshop, San Francisco, CA (2001); *Predicting Internet Network Distance with Coordinates-Based Approaches*, Proc. IEEE Infocom, New York, NY, USA (2002).

[8] S. Ratnasamy, M. Handley, R. Karp, S. Shenker, *Topology-Aware Overlay Construction and Server Selection*, Proc. IEEE Infocom, New York, NY, USA (2002).

[9] S. van Langen, X. Zhou and P. Van Mieghem, 2004, *On the Estimation of Internet Distances Using Landmarks*, International Conference on Next Generation Teletraffic and Wired/Wireless Advanced Networking (NEW2AN'04), St. Petersburg, Russia, February 2-6 (2004).

[10] RIPE Test Traffic Measurements, http://www.ripe.net/ttm.

[11] Y. Shavitt and T. Tankel, *On the Curvature of the Internet and its usage for Overlay Construction and Distance Estimation*, Proc. IEEE Infocom, Hong Kong, CN (2004)

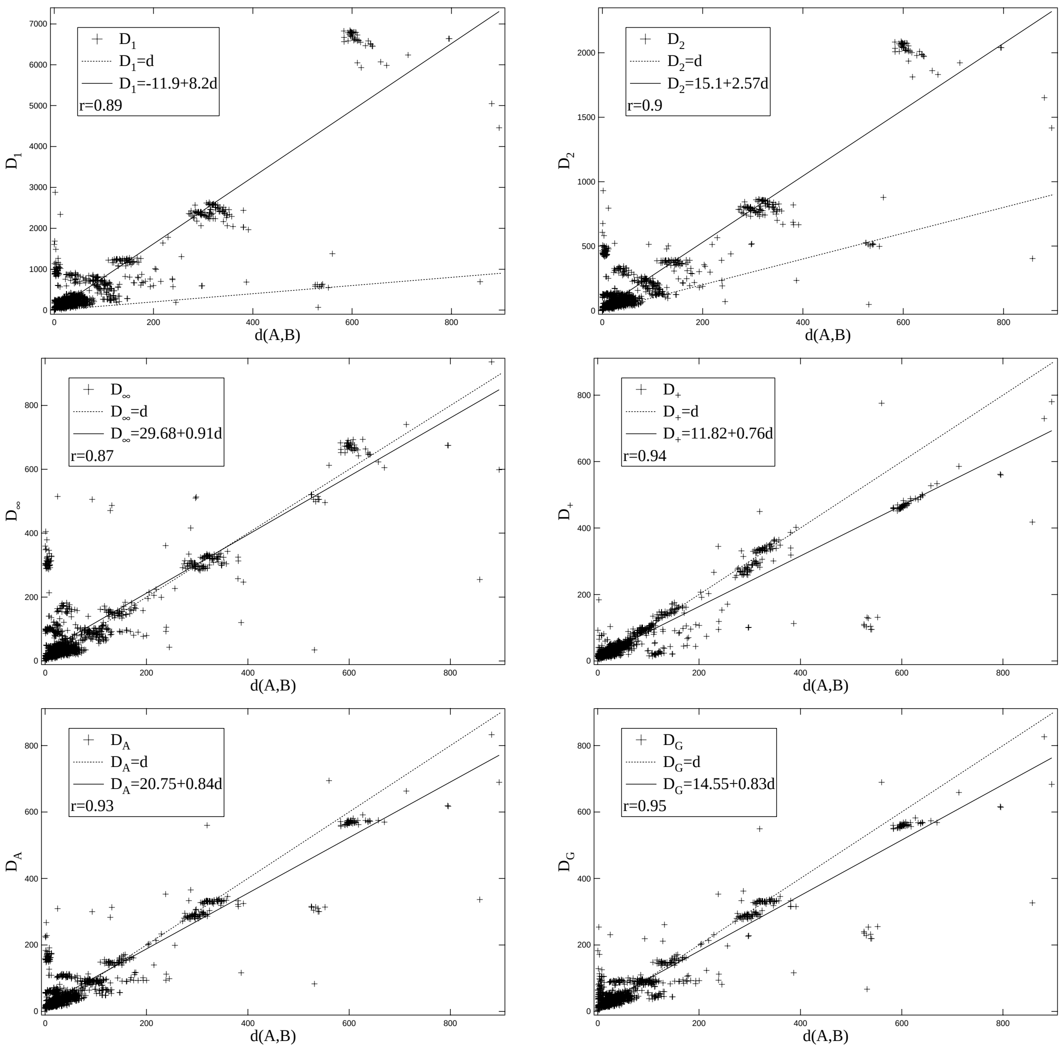

Fig. 2. Scatterplots for the estimated (i.e. hyperspace) distance (using data from 1 day eariler) versus the measured distance, for six hyperspace distance functions. The number of landmarks is 10. The plotted lines are the diagonal D = d, and a least-squares fit

	4 days earlier	**3 days earlier**	**2 days earlier**	**1 day earlier**
D_1	0.79, 0.85, 0.84	0.81, 0.87, 0.86	0.87, 0.91, 0.9	0.9, 0.89, 0.89
D_2	0.81, 0.85, 0.84	0.82, 0.86, 0.85	0.88, 0.9, 0.89	0.9, 0.9, 0.9
D_∞	0.81, 0.75, 0.74	0.82, 0.84, 0.82	0.86, 0.87, 0.86	0.88, 0.87, 0.87
D_+	0.94, 0.94, 0.93	0.93, 0.93, 0.91	0.97, 0.94, 0.95	0.97, 0.94, 0.93
D_A	0.9, 0.87, 0.86	0.89, 0.87, 0.85	0.94, 0.9, 0.89	0.95, 0.93, 0.93
D_G	0.92, 0.91, 0.91	0.92, 0.91, 0.9	0.96, 0.95, 0.94	0.97, 0.95, 0.95
	9 months earlier	**14 days earlier**	**6 days earlier**	**5 days earlier**
D_1	0.34, 0.38, 0.4	0.41, 0.48, 0.40	0.73, 0.79, 0.78	0.75. 0.82, 0.81
D_2	0.38, 0.41, 0.43	0.46, 0.52, 0.44	0.74, 0.79, 0.78	0.77, 0.82, 0.81
D_∞	0.48, 0.47, 0.49	0.56, 0.56, 0.51	0.76, 0.82, 0.8	0.79, 0.83, 0.82
D_+	0.69, 0.68, 0.68	0.74, 0.74, 0.66	0.87, 0.87, 0.86	0.93, 0.93, 0.91
D_A	0.58, 0.57, 0.59	0.65, 0.66, 0.6	0.84, 0.82, 0.82	0.88, 0.86, 0.85
D_G	0.58, 0.58, 0.59	0.66, 0.66, 0.6	0.85, 0.85, 0.85	0.9, 0.9, 0.89

TABLE I

CORRELATION COEFFICIENTS FOR THE ESTIMATED DELAY DATA.