

Similarity Search in a Hybrid Overlay P2P Network

Mouna Kacimi, Kokou Yétongnon
University of Bourgogne
Laboratoire LE2I
Sciences et Techniques 21078 Dijon Cedex France
kacimi@khali.u-bourgogne.fr
kokou.yetongnon@u-bourgogne.fr

Abstract

P2P systems are increasingly used to discover and share various data between users. The performance of a P2P based information retrieval system is determined by the efficiency of locating information and the precision with which the retrieved data corresponds to the submitted query. In this paper, we address the problem of similarity search by providing a combined peer organizing and data organizing scheme for approximate similarity search in a Hybrid Overlay P2P Network. Through extensive simulations, we show the efficiency of the similarity search in our approach using a density-based algorithm to cluster data and peers in a high dimensional feature space.

1 Introduction

Information search is increasingly important in many applications, particularly in web based distributed environments where users and applications can share various contents. With the evolution communication technologies, the number of accessible unstructured information repositories has increased rapidly. Traditional search techniques based on immediate lookup and exact matching of keywords are no longer sufficient for many emerging applications such as image retrieval and data mining. New models based on the unifying concepts of *similarity searching* or *proximity searching* are needed for discovering and retrieving similar or close objects to a given query. Similarity searching is particularly important in fully distributed networks such as P2P systems in which various routing schemes are used to submit queries to a group of relevant nodes. Many search techniques have been proposed for P2P systems relying on their underlying overlay infrastructure.

Generally, two broad categories of P2P can be distinguished: *unstructured systems* and *structured systems*. Unstructured P2P overlay networks organize peers randomly,

while the structured systems impose a topology on the peers and place data not at random peers but at specific locations. Information searching in unstructured P2P systems are based on progressively flooding requests in the network by broadcasting them to directly connected peers. Several methods have been used to improve the performance of the basic flooding search. Mechanisms such as Time-To-Live (TTL) are used to reduce network load by limiting the scope of the search. Another mechanism for reducing network load is the random walk [5] which, at each step of the search, forwards a query to a randomly chosen neighbor until the required object is found [3].

Clustering techniques can also be used to reduce the scope of query flooding. Clustering involves the creation of links on top of unstructured P2P overlay networks to organize peers according to their common properties or interests. Several parameters can be used to cluster peers, such as network related information [1, 4, 13], application needs [19, 29], peer characteristics [11, 16], and similarities between peers contents [2, 6, 14, 20–22, 27]. Many methods are used to measure the similarity between peers. For example, Sripanidkulchai et al [14] take into account the query traces over the P2P network [14]. Hang et al [22, 25] generate signature vectors based on low-level features describing peer content. Other techniques presented in [2, 20, 21, 27] associate peers with semantic descriptions that can be simple keyword-based annotations, schema or ontologies. Mainly, clustering helps limiting the flooding, thus, the query is sent only to peers that are more likely to have relevant data. We note that similarity issues have been addressed by some clustering techniques such as Firework query model presented in [22, 25] and SON system [27]. However, clustering approaches organizes peers according to their largest data interests providing an approximate description of peers' content. Thereby, the interests of data objects that represent a small fraction of the peer content are ignored and might not be reachable.

Structured P2P systems are based on DHT (Distributed

Hashing Table) [9, 10, 15, 17, 18, 24, 26, 30]. DHT systems assign from the same key space, a unique identifier to each peer called *NodeID* and a unique identifier to each data object called a *key*. Data objects are mapped by the overlay network protocol to unique live peer having the closest identifier to the data object key. Each peer maintains a routing table consisting of its neighboring peers' *NodeIDs* and IP addresses. A lookup query is identified by a unique key and is forwarded in a progressive manner to peers having *NodeIDs* that are closer to the query key. Each DHT-based system has a different organization schemes for the data objects, the key space and routing strategies. DHT techniques allow an efficient data access with a complete lookup and reduce the hops number to locate data. However, they are more appropriate for exact queries. Thus, the main challenge for those systems is to process complex queries such as similarity, approximate and range selections. This challenge was recently addressed in [7, 8, 23, 28] by adding a layer on top of the existing DHT systems to process complex queries.

In this paper we propose a Hybrid Overlay Network that combines the characteristics of both clustering and DHT techniques for more precise peers' content description which results in more accurate similarity search. Data and peers are organized in a high feature space describing their content. The main contributions of our paper are:

- Organizing peers to limit the flooding overhead and send queries only to the relevant peers
- Organizing data objects in specific locations by placing similar data objects in neighboring locations. Thereby, complex queries such as range and neighboring queries are processed efficiently.

In the next section, we introduce the Hybrid Overlay Network (HON) for organizing data and peers in a high-dimensional feature space. Section 3 presents the similarity search in HON-P2P system. Section 4 presents the simulations results. And finally section 5 concludes the paper.

2 Hybrid Overlay Network

The Hybrid Overlay Network (HON) combines data organization and peers organization techniques to perform an efficient similarity search based on range and neighboring queries. The data contents of peers are represented by *n*-element vector called Feature Vector (FV), where each element define a particular feature or attribute associated with data object (e.g., color for an image, concept or key word for a text document). Since each data object is described by a feature vector, it can be seen as a point in a *n*-dimensional feature space.

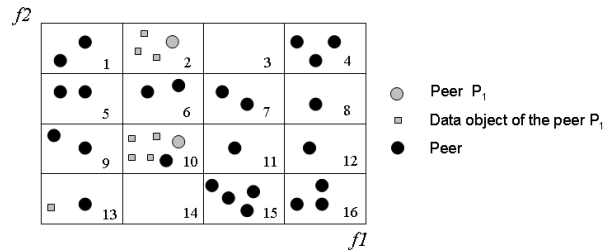


Figure 1. Data and peers organization

We define a partition of the feature space into cells as presented in our previous work [12], and use the distribution of data objects over the cells as the basis for computing query similarities to data objects and peers. Thus, two peers are considered similar if their contents are distributed on the same sub regions of the feature space. Two steps are required to construct the Hybrid Overlay Network (HON):

The first step consists in organizing data objects in the feature space. Each data object is described by a feature vector and corresponds to one point in the feature space. Therefore, it is mapped to one cell. The number of data objects in each cell is recorded using a cell density measure. This notion of cell density is used for mapping a peer to a cell and can be used later to create clusters.

The second step consists in organizing peers in the feature space. Each peer is mapped to a set of cells containing its objects. The mapping is done using a threshold value T . A peer is mapped to a cell only if it has a number of objects higher than T in the cell. Figure 1 shows the partition cells of a 2-dimensional feature space using the features f_1 and f_2 and the distribution of data objects of the peer P_1 . The data objects of P_1 are mapped to the cells 2, 10 and 13. Using a threshold equals to 1, the peer P_1 is mapped only to the cells 2 and 10 because the number of its data objects in those cells is higher than 1.

The cell granularity represents the size of a cell. We have evaluated in section 4 the impact of cells granularity on the similarity search efficiency. The results show that the lower the cell granularity is the more accurate the similarity between data objects. A high cell granularity, maps data peers to few cells which makes larger the flooding space and gives less precise and similar results. Low granularities assure an efficient similarity search but they tend to generate a high number of cells. The high increase of cells number may leads to several problems. The main problem is the increase of the search path length. A high number of hops might be required to reach the destination since the query is routed from a cell to the adjacent one and no shortcuts are available in our approach. Another problem of high cells number is the maintenance cost increase. Additional costs are required to build and update indexes maintained by each

peer when peers join and leave the network. To conserve low granularities for better similarity search and avoid the problems of cells number explosion, cells can be grouped into clusters to reduce the maintenance cost and the search path length. We assume that each cluster is managed by one super peer which maintains neighboring contacts to its neighboring clusters. Each peer holds contacts to the super peers managing its relevant clusters.

Cells belonging to the same cluster should share similar properties for better performance. For different objectives, several criteria can be used to group cells into the same cluster. A cluster can group together a set of adjacent cells to perform the range query. Another method is to group cells that are frequently communicating or asked together to reduce the search path length. In our approach, we use a density-based algorithm that cluster cells according to two parameters: cells adjacency and cells density. The objectives of the density-based algorithm are:

- Cluster adjacent cells for retrieving similar objects.
- Build clusters according to cells density to provide a high recall. Grouping cells with high densities increases the number of retrieved similar objects.

Density-Based Algorithm Cell density is the number of objects the cell contains. The density-algorithm works in the following way:

- 1- Compute the density of cells
- 2- Select the unmarked cells that have the highest density
- 3- A cell can be in one of three conditions:
 - If there is no cluster adjacent to the cell, it forms a new cluster
 - If there is one cluster adjacent to the cell, it joins the adjacent cluster
 - If there is two adjacent clusters the cell joins the cluster having less density for load balancing. Note that the density of a cluster represents the number of objects it contains.
- 4- Mark the processed cells.
- 5- Repeat the process until all the cells are marked.

3 Similarity search

Let a query object Q be described by the coordinates $[f_{q1}, f_{q2}, \dots, f_{qn}]$ in the feature space, where f_{qi} is the i_{th} feature value of Q . The query Q represents one point in the feature space called the query point Q_P (see figure 2). When a query Q is initiated, the requesting peer maps the query to the cell containing the query point Q_P where the

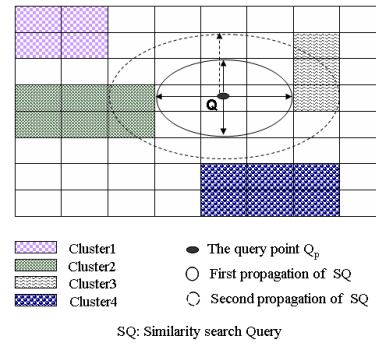


Figure 2. Example of similarity search

required object resides called *target cell*. Then, the search takes the form of different actions, depending on if the target cell is found or not.

If the target cell is found, the requesting peer checks its cluster-index to extract the cluster address to which the target cell belongs. Then, the relevant super peer floods all the peers contained in the target cell to find the required object. When a peer receives the query Q , it computes the distance between the query Q and each of its objects belonging to the target cell. When the distance is less than a predefined *Similarity Threshold* ST , the object is sent to the requesting peer. The distance between a query and an object is defined as follows:

Definition (Query-Object Distance)

Assuming that an object O is described by a set of features : $D_o = [f_{o1}, f_{o2}, \dots, f_{ok}, \dots, f_{on}]$, and a query Q is described by a set of features $D_q = [f_{q1}, f_{q2}, \dots, f_{qk}, \dots, f_{qn}]$. The distance between query Q and object O is given by
$$\sum_{i=1}^n \frac{|f_{qi} - f_{oi}|}{n}$$
.

If the target cell is not found, which means that the cell is empty, we then find the closest regions in the feature space to the target cell by generating a *Similarity search Query* SQ . The similarity search query SQ is a range query having as lower and upper bounds l and u where the middle point $(l+u)/2$ is the query point Q_P . The similarity search query is propagated recursively to the adjacent cells until at least one cluster intersect the query SQ . The figure 2 shows an example of a query Q mapped to an empty cell. A similarity search query SQ then is generated and propagated to the adjacent cells. The query SQ in this example does not intersect any cluster because all the adjacent cells are empty. Therefore, the query SQ is extended with larger range values and propagated to the next adjacent cells. The second propagation of the query SQ intersects Cluster2 and Cluster3. Only the cells of Cluster2 and 3 covered by the query SQ are considered as *target cells*. These cells are then

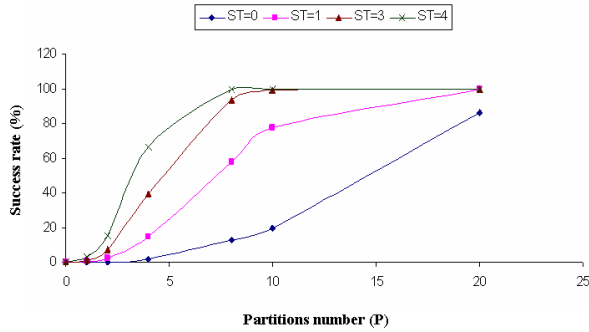


Figure 3. The impact of cell granularity on the success rate

queried to return most similar objects to the initial query Q . Once the target cells are defined, the query SQ is then processed in the same manner described in the first case. Note that the similarity threshold ST for the query SQ is higher than the one of the initial query Q and it varies according to the user needs.

4 Evaluations

Our simulation consists in evaluating the performance of similarity search in HON-P2P system using the density-based clustering. We run 10,000 peers with 1,500,000 objects following a uniform distribution where the average number of objects per peer is equal to 150. The feature space is described using 4 features and composed of 10,000 cells. To measure the similarity search we simulate 5,000,000 queries over the network.

In this simulation we focus on three metrics to evaluate the similarity search:

- *The success rate*: the percentage of the most similar responses to the query object Q called relevant responses. The success rate is computed by $R \times 100 / K$, where R is the number of the relevant responses and K is the total number of responses.
- *The recall*: what fraction of the relevant responses has been retrieved? If a search only retrieves one hundred relevant responses out of three thousand that are available, that search has a low recall. If it retrieves all the available responses to the query Q , it has a high recall. Let TR be the total number of the relevant responses for a query Q and RR be the number of the retrieved responses. The recall is computed by $RR \times 100 / TR$.
- *The query scope*: what fraction of peers in the system is involved in query processing? A smaller query scope increases system scalability. The query scope is computed by $QP \times 100 / P$, where P is the total number of peers and

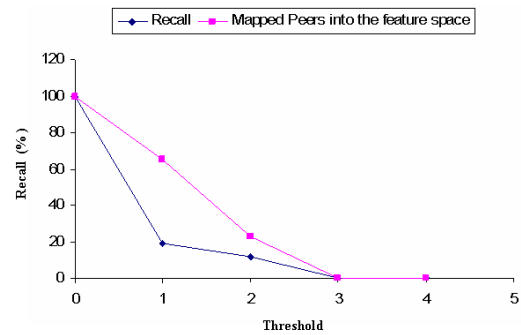


Figure 4. The recall according to threshold value

QP is the average number of peers involved in the query processing.

Figure 3 shows that the success rate depends on two parameters: the cell granularity and similarity threshold ST . The success rate reaches high values with low granularities compare to high granularities. In addition, when the similarity threshold increases, it provides a higher success rate. In our simulation, the range values of low level features have as lower and upper bounds f_{min} and f_{max} , where $f_{min}=0$ and $f_{max}=40$. As shown in figure 3 a partition equals to 1 and a similarity threshold equals to 4 provide only 2,96% of success rate, while a partition equals to 20 with similarity threshold equals to 0 allow 86,44% of success rate.

The recall in HON-P2P system depends on the used threshold value to map peers to cells. If the threshold value T is equal to 0, peers are mapped to all the cells where their objects reside. In this case, a query Q will get the answers from all the peers in the system containing at least one object in the target cell. Therefore, the recall reaches 100% assuring a complete lookup. If the threshold value T increases as shown in figure 4, a peer having a number of objects in the target cell less than T will not be requested and its objects cannot be reached. Consequently, no complete lookup implies a recall decrease. In the figure 4, we notice that the recall decreases when the threshold value increases.

The query scope depends on two parameters. First, the threshold value T . As presented before, the increase of the threshold leads to a low recall due to the fact that not all the peers having the relevant answers can be reached. Thus, the number of peers involved in the query processing decreases with the increase of the threshold T . The figure 5(b) shows the impact the threshold T on the query scope. Second, the cell granularity G . A high granularity increases the num-

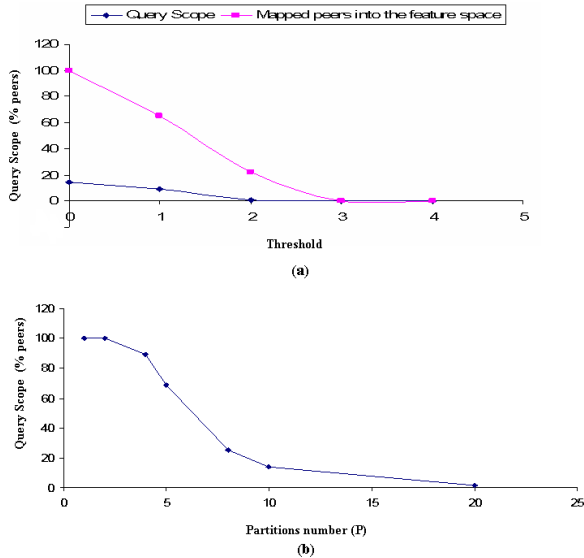


Figure 5. The query scope according the cell granularity and the threshold value

ber of peers containing in each cell, thus the number of the requested peers increases as shown in figure 5(b).

5 Conclusion and future work

In this paper we have presented a Hybrid Overlay Network that organizes data and peers in a high dimensional feature space to perform the similarity search. A clustering algorithm based on cells densities was introduced to decrease the system maintenance cost imposed by the high number of cells. We have presented the simulation results showing the high success rate and recall the HON-P2P system can achieve, and the different parameters affecting the similarity search efficiency. Our future work consists in studying the maintenance cost and failure consequences in HON-P2P system.

References

- [1] A. Bestavros and S. Mehrotra. Dns-based internet client clustering and characterization. Technical Report BUCS-2001-012 MA 02215, Boston University, Computer Science Department, Boston, June 2001.
- [2] A. Crespo and H. Garcia-Molina. Semantic overlay networks for p2p systems. Technical report, Computer Science Department, Stanford University, October 2002.
- [3] L. Adamic, R. Lukose, A. Puniyani, and B. Huberman. Search in power law networks. 64(E):46135–46143, 2001.
- [4] A. Agrawal and H. Casanova. Clustering hosts in p2p and global computing platforms. In *Proceedings of the 3rd International Symposium on Cluster Computing and the Grid (CCGRID)*, pages 367–373, 2003.
- [5] E. Cohen and S. Shenker. Replication strategies in unstructured peer-to-peer networks. In *Proceedings of the 2002 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 177–190, 2002.
- [6] S. B. Handurukande, A.-M. Kermarrec, F. L. Fessant, and L. Massoulié. Exploiting semantic clustering in the edonkey p2p network. In *Proceedings of the 11th ACM SIGOPS European Workshop*, 2004.
- [7] M. Harren, J. M. Hellerstein, R. Huebsch, B. T. Loo, S. Shenker, and I. Stoica. Complex queries in dht-based peer-to-peer networks. In *Proceedings of IPTPS02*, 2002.
- [8] R. Huebsch, J. M. Hellerstein, N. Lanham, B. T. L. S. Shenker, and I. Stoica. Querying the internet with pier. *Proceedings of 19th International Conference on Very Large Databases (VLDB)*, pages 321–332, 2003.
- [9] I. Clarke, O. Sandberg, B. Wiley, and T. W. Hong. Freenet: A distributed anonymous information storage and retrieval system. In *Lecture Notes in Computer Science*, 2009:311–320, 2001.
- [10] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan. Chord: A scalable peer-to-peer lookup service for internet applications. *ACM SIGCOMM*, pages 149–160, 2001.
- [11] JXTA. <http://www.jxta.org/>, 2004.
- [12] M. KACIMI, Y. MA, R. CHBEIR, and K. YETONGNON. Hon-p2p: A cluster-based hybrid overlay network for multimedia object management. *The 11th International Conference on Parallel and Distributed Systems, IEEE Computer Society*, 2005.
- [13] B. Krishnamurthy and J. Wang. On network-aware clustering of web clients. *Proceedings of the conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, pages 97–110, 2000.
- [14] K. Sripanidkulchai, B. Maggs, and H. Zhang. Efficient content location using interest-based locality in peer-to-peer systems citation. *INFOCOM*, 2003.
- [15] J. Kubiawicz, D. Bindel, P. Eaton, Y. Chen, D. Geels, R. Gummadi, S. Rhea, W. Weimer, C. Wells, H. Weatherpoon, and B. Zhao. Oceanstore: An architecture for globalscale persistent storage. In *Proceedings of ACM SIGPLAN Notices*, 35(11):190–201, November 2000.
- [16] A. Loser, W. Nejdl, M. Wolpers, and W. Siberski. Information integration in schema-based peer-to-peer networks. In *Proceedings of the 15th Conference On Advanced Information Systems Engineering*, 2003.
- [17] D. Malkhi, M. Naor, and D. Ratajczak. Viceroy: A scalable and dynamic emulation of the butterfly. In *Proceedings of the 21st annual ACM symposium*, pages 183–192, 2002.
- [18] G. S. Manku, M. Bawa, and P. Raghavan. Symphony: Distributed hashing in a small world. In *Proceedings of the 4th USENIX Symposium on Internet Technologies and Systems*, 2003.
- [19] S. Marti, P. Ganesan, and H. Garcia-Molina. Sprout: P2p routing with social networks. *International Conference on*

Extending Database Technology (EDBT), pages 425–435, 2004.

- [20] W. Nejdl, M. Wolpers, W. Siberski, A. Loser, I. Bruckhorst, M. Schlosser, and C. Schmitz. Semantic overlay clusters within super-peer networks. *In Proceedings of the International World Wide Web Conference*, pages 33–47, 2003.
- [21] W. Nejdl, M. Wolpers, W. Siberski, C. Schmitz, M. Schlosser, I. Brunkhorst, and A. Loser. Super-peer-based routing and clustering strategies for rdf-based peer-to-peer networks. *Proceedings of the twelfth international conference on World Wide Web*, pages 536–543, 2003.
- [22] C. Ng and K. Sia. Peer clustering and firework query model. *Proceedings of 11th World Wide Web Conference*, 2002.
- [23] P. Rösch, K.-U. Sattler, C. Weth, and E. Buchmann. Best effort query processing in dht-based p2p systems. *Proc. of the 1st IEEE International Workshop on Networking Meets Databases (NetDB)*, 2005.
- [24] A. Rowstron and P. Druschel. Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems. *IFIP/ACM International Conference on Distributed Systems Platforms (Middleware)*, pages pages 329–350, 2002.
- [25] K. C. Sia, C. Ng, C. Chan, S. Chan, and L. Ho. Bridging the p2p and www divide with discover - distributed content-based visual information retrieval. *The Twelfth International World Wide Web Conference (WWW)*, 2003.
- [26] S.Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker. A scalable content-addressable network. *In Proc. ACM SIGCOMM*, 2001.
- [27] I. Tatarinov, Z. Ives, J. Madhavan, A. Halevy, D. Suciu, N. Dalvi, X. Dong, Y. Kadiyska, G. Miklau, and P. Mork. The piazza peer data management project. *SIGMOD Record*, 3:47–52, 2003.
- [28] P. Triantafillou and T. Pitoura. Towards a unifying framework for complex query processing over structured peer-to-peer data networks. *Workshop on Databases, Information Systems and Peer-to-Peer Computing (DBISP2P)*, 2003.
- [29] H. J. Wang, Y. Hu, C. Yuan, Z. Zhang, and Y. Wang. Friends troubleshooting network: Towards privacy-preserving, automatic troubleshooting. *IPTPS04*, 2004.
- [30] B. Y. Zhao, L. Huang, J. Stribling, S. C. Rhea, A. D. Joseph, and J. D. Kubiatowicz. Tapestry: A resilient global-scale overlay for service deployment. *IEEE Journal on Selected Areas in Communications*, 22(1):41–53, January 2004.