SRI: Exploiting Semantic Information for Effective Query Routing in a PDMS *

Federica Mandreoli, Riccardo Martoglia and Simona Sassatelli
DII - University of Modena and Reggio Emilia, Italy
{fmandreoli, rmartoglia, sassatelli}@unimo.it

Wilma Penzo
DEIS - University of Bologna, Italy
wpenzo@deis.unibo.it

ABSTRACT

The huge amount of data available from Internet information sources has focused much attention on the sharing of distributed information through Peer Data Management Systems (PDMSs). In a PDMS, peers have a schema on their local data, and they are related each other through semantic mappings that can be defined between their own schemas.

Querying a PDMS means either flooding the network with messages to all peers or take advantage of a routing mechanism to reformulate a query only on the *best* peers selected according to some given criteria. As reformulations may lead to semantic approximations, we deem that such approximations can be exploited for locating the *semantically best directions* to forward a query to.

In this paper, we propose a distributed index mechanism where each peer is provided with a Semantic Routing Index (SRI) for routing queries effectively. A fuzzy-oriented model for SRI is presented where operations for creating and maintaining SRIs are well-founded. In addition, we show how SRIs can be employed in the query processing phase with the aim of reducing the space of reformulations. Finally, we conduct a series of meaningful experiments showing the effectiveness of the proposed approach.

Categories and Subject Descriptors

E.1 [**Data**]: Data Structures—*distributed data structures*; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*search process*

General Terms

Management

*This work is partially supported by the Italian Council co-funded project WISDOM.

WIDM'06, November 10, 2006, Arlington, Virginia, USA.

Keywords

Query routing, Semantics, Data-sharing P2P systems

1. INTRODUCTION

The ever-growing and widespread availability of data from Internet information sources has placed great interest on the potential of information sharing. This awareness has led to successful systems such as Napster [3] and Gnutella [2], just to mention a few. These are P2P systems, i.e., systems with distributed computing capabilities, where each peer exchanges information and services directly with other peers of the system.

On the other hand, the huge number of data sources spread over the network has focused the attention on the problem of *where* to find *relevant* information. To this end, the Semantic Web community has spent much work on defining techniques for providing data sources with semantic information aiming at describing the knowledge offered to the network. In this scenario, P2P systems have evolved towards Peer Data Management Systems (PDMSs) where each peer is enriched with a schema that represents the peer's domain of interests, and semantic mappings are locally established between peers' schemas [10, 17, 4]. A semantic overlay network is thus put at advanced search mechanisms disposal for effective data retrieval, in that the semantic information can be exploited to locate relevant information over the huge amount of data available in the network.

A PDMS is thus a decentralized architecture for web-scale data sharing: peers are autonomous as to the data they store locally and to the semantic description and conceptual organization they provide for the data they want to share with other peers. Let us consider Figure 1 as a sample scenario of a PDMS concerning data about operas. Each peer is provided with a schema, for instance XML-based, whereas bold lines denote semantic mappings between pairs of peers. In order to query a peer in a PDMS, its own schema is used for query formulation, and query answers can come from any peer in the network that is connected through a semantic path of mappings [21]. As an example, let us consider the following query, posed on the schema of Peer1: "Retrieve the main singers of the opera entitled *Aida*". In order to be answered by Peer3, which is directly connected to Peer1, such query is to be reformulated according to the semantic mappings established between the schemas of the involved peers. However, due to the heterogeneity of the schemas,

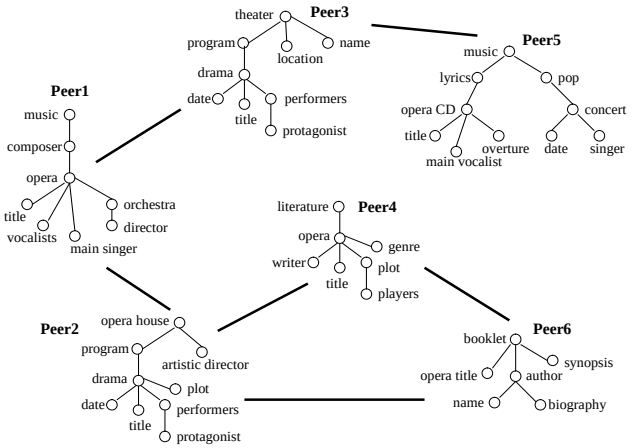


Figure 1: Reference example

the reformulation of a query may lead to some semantic approximation and, consequently, the returned data may not exactly fit with the query conditions. For instance, `main singer` in Peer1's schema could be reformulated in a non-exact way on Peer3's `performers`. To this purpose, semantic mappings can be conveniently extended with a score, thus giving a measure of the semantic compatibility occurring between the involved portions of schemas. As such scores reflect the relevance of peer's data to a query, we deem that semantic mappings can be exploited in the searching phase to suggest a direction towards the semantic paths which *better* satisfy the query conditions.

Indeed, an important aspect to be considered is that a PDMS underlies a potentially very large network able to handle huge amounts of data. In this context, any relevant peer may add new answers to a given query and different paths to the same peer may yield different answers [21]. For this reason, *query routing*, i.e. the process of selecting the most promising peers, is a fundamental issue for querying distributed resources. In particular, in a semantic web perspective, a query posed over a given peer should be forwarded to the most relevant peers that offer semantically related results among its immediate neighbors first, then among their immediate neighbors, and so on. For instance, in the reference example, the Peer1's neighbors Peer2 and Peer3 might be considered mirroring peers as to the portion of the schemas involved in the query above; as to the second step of query reformulation, Peer5 is more relevant than Peer4 and Peer6, since it deals with lyric music data, instead of written operas. For these reasons, the answers obtained from path Peer3-Peer5 fit better the query conditions than those from paths Peer2-Peer4-Peer6 and Peer2-Peer6.

When a query is forwarded through a semantic path, it undergoes a multi-step reformulation which may involve a chain of semantic approximations. To this purpose, our proposal is to exploit such approximations for selecting the direction which is *more likely* to provide the best results to a given query. Our perspective is knowledge-based, in that the routing of a query is guided by the semantic mappings between the peers. However, coming back to our example, the information provided by the semantic mappings stored in Peer1 is not enough for deciding which is the best path. In fact, being Peer2 and Peer3 mirrors, the semantic approximation of the query would be measured as identical for both directions. Therefore, broadly speaking, some kind of information about the relevance of the whole semantic paths should be available in the network, maintained up-to-date, and easily accessible for query routing purposes. To this end, we borrow the idea of a distributed-index mechanism from the literature [7] which maintains indices at each node. In our proposal, a *Semantic Routing Index* (SRI) summarizes, for each concept of its peer's schema, the semantic approximation "skills" of each subnetwork reachable from its immediate neighbors, and thus gives a hint of the relevance of the data which can be reached in each path. For instance, the Peer1's SRI will contain two entries, one for the upward subnetwork and one for the downward one. The semantic knowledge stored in a SRI is summarized on the available directions in order to maintain the size of the semantic index proportional to the number of neighbors, thus scaling well in a PDMS scenario.

In this paper, we study SRIs for semantic query routing and we evaluate their effectiveness. In particular, the contribution of this paper are:

- In Section 2 we extend the notion of semantic mapping and semantic paths with a score, expressing the grade of uncertainty which naturally arises from establishing a correspondence between semantic concepts. To this purpose we rely on the fuzzy set theory [12].
- On the basis of this fuzzy-oriented view of mappings, in Section 3 we introduce SRIs as distributed indices for query routing, and we give a fuzzy interpretation of the operations necessary to create and maintain them.
- In Section 4 we perform an evaluation of the effectiveness of SRIs through a set of query routing experiments on a wide variety of scenarios.
- Finally, in Section 5 we relate to other approaches in the literature and we discuss future extensions to our work.

2. SEMANTIC PEERS

In this section, we introduce the basic concepts our routing mechanism relies on. They are defined in a fuzzy theoretical framework. Fuzzy set theory has been widely applied in contexts where uncertainty of description is intrinsic in the nature of the data, most notably in the case of multimedia data [8, 19]. We deem that these principles can provide a valid support to deal with the semantic approximation originated by the heterogeneity of the schemas in a PDMS.

We denote with $\mathcal{P}$ a set of peers. Each peer $p_i \in \mathcal{P}$ stores local data, modelled upon a local schema S_i which can be, for instance, an ontology, a relational, or a XML schema. This makes a peer p_i a *semantic* peer, in that its local schema S_i describes the semantic content of its underlying data. Without loss of generality, we consider a peer schema S_i as a set of semantic concepts $\{C_{i_1}, \ldots, C_{i_{m_i}}\}$, each one understanding, for instance, an ontology class, or a relational table, or an XML schema element.

2.1 Semantic Mappings

Peers are pairwise connected in a semantic network through semantic mappings between peers' schemas. For our query

routing purposes, we abstract from the specific format that semantic mappings may have. For this reason, we consider a simplified scenario, and we assume directional, pairwise and one-to-one semantic mappings. The approach we propose can be straightforwardly applied to more complex mappings relying on query expressions as proposed in [10, 17, 4].

A semantic mapping can be established from a source schema S_j to a target schema S_i, and it defines how to represent S_i in terms of S_j's vocabulary. In particular, it associates each concept in S_i to a corresponding concept in S_j according to a *score*, denoting the *degree of semantic similarity* between the two concepts. A formal definition of semantic mapping can be given according to a fuzzy interpretation, and it relies on the concept of fuzzy relation [12].

DEFINITION 1 (SEMANTIC MAPPING). *A* semantic mapping *from a source schema S_j to a target schema S_i, not necessarily distinct, is a fuzzy relation $M(S_i, S_j) \subseteq S_i \times S_j$ where each instance (C, C') has a membership grade denoted as $\mu(C, C') \in [0, 1]$ and indicating the strength of the relation between C and C'. This fuzzy relation satisfies the following properties: 1)it is a 0-function, i.e., for each $C \in S_i$, it exists exactly one C' in S_j such that $\mu(C, C') \geq 0$; 2) it is reflexive, i.e., given $S_i = S_j$, for each $C \in S_i$ $\mu(C, C) = 1$.*

Without loss of generalization, we assume that the self mapping $M(S, S)$ is the identity relation. Notice that a non-mapped concept has membership grade 0. A sample tuple of the semantic mapping between S_1 and S_3 of the reference example is $M(S_1, S_3)(\texttt{main singer}, \texttt{protagonist}) = 0.7$.

Whenever a peer joins a PDMS, it selects a small subset of peers as its *neighboring* peers, computes and stores the corresponding mappings in its local repository. To this end, semi-automatic techniques can be employed. Since they are not in the focus of this paper, we refer the reader to [14, 15]. Semantic mappings are used for query reformulation: When a querying peer p_i forwards the query q to one of its neighbors, say, p_j, q must be reformulated into q' so that it refers to concepts in the p_j's schema. To this end, p_i uses the semantic mapping $M(S_i, S_j)$. In this context, reformulation amounts to unfolding [21].

2.2 Semantic Paths

A semantic path is a chain of semantic mappings connecting a given pair of peers. Through the reformulation of a query along the mappings composing a semantic path, the PDMS can access data on remote peers. As local semantic mappings may involve semantic approximations, the semantic approximation given by a semantic path can be obtained by composing the fuzzy relations understood by the involved mappings. This relies on the notion of *generalized composition* of binary fuzzy relations [12].

DEFINITION 2 (COMPOSITION OF MAPPINGS). *Given a t-norm[1] I and the semantic mappings, $M(S_i, S_j) \subseteq S_i \times S_j$ and $M(S_j, S_k) \subseteq S_j \times S_k$, the I-composition of $M(S_i, S_j)$ and $M(S_j, S_k)$ is the semantic mapping $M(S_i, S_j) \circ^I M(S_j, S_k) \subseteq S_i \times S_k$ defined by: $[M(S_i, S_j) \circ^I M(S_j, S_k)](C, C'') = I[M(S_i, S_j)(C, C'), M(S_j, S_k)(C', C'')]$, $\forall C \in S_i, C'' \in S_k$, with $C' \in S_j$.*

[1]A t-norm I is a binary operation on $[0, 1]$ that is monotone, commutative, associative, and it satisfies the boundary condition $I(a, 1) = a$ for all a in $[0, 1]$.

The composition of more complex mappings require specific algorithms [21]. However, as we will see later, we are not properly interested in the instances of the resulting semantic mapping but rather on their membership grades.

DEFINITION 3 (SEMANTIC PATH). *Given a t-norm I and a sequence of mappings $< M(S_1, S_2), \ldots, M(S_{k-1}, S_k) >$ connecting peer p_1 with peer p_k, the path $P_{p_1 \ldots p_k} \subseteq S_1 \times S_k$ is the semantic mapping $M(S_1, S_2) \circ^I \ldots \circ^I M(S_{k-1}, S_k)$.*

The composition function should capture the intuition that the longer the chain of mappings, the lower the grades, thus denoting the accumulation of semantic approximations given by a sequence of connected peers. In order to obtain such effect of semantic attenuation due to the chain of mappings from C_1 to C_k in the schema of a peer p_k which is far away from p_1, several alternatives exist for the t-norm I [12]. For instance, a possible choice for the t-norm I is the *algebraic product* $I(\mu, \mu') = \mu * \mu'$. In fact, given that the arguments are grades in $[0, 1]$, their algebraic product is still in $[0, 1]$, and it is lower than or at most equal to its arguments.

Given $M(S_1, S_3)(\texttt{main singer}, \texttt{protagonist}) = 0.7$ and $M(S_3, S_5)(\texttt{protagonist}, \texttt{main vocalist}) = 0.5$ in the reference example, their composition based on the algebraic product yields to the following instance of the semantic path $P_{\text{Peer1, Peer3, Peer5}} \subseteq S_1 \times S_5 : (\texttt{main singer}, \texttt{main vocalist}) = 0.35$.

2.3 Generalized Semantic Mappings

The query execution process starts from the querying peer which reformulates the query over its immediate neighbors, then over their immediate neighbors and so on. Thus, from a multi-step reformulation point of view, whenever a query posed over peer p_i is reformulated over peer p_j, the query is moving from p_i to the subnetwork rooted at p_j and it might follow any of the semantic paths originating at p_j. In order to model the semantic approximation of the p_j's subnetwork w.r.t. the p_i's schema, the semantic approximations given by each path in the p_j's subnetwork are aggregated into a measure reflecting the relevance of the subnetwork as a whole.

To this end, the notion of semantic mapping is generalized as follows. Let p_j^Δ denote the set of peers in the subnetwork rooted in p_j, S_j^Δ the set of schemas $\{S_{j_k} | p_{j_k} \in p_j^\Delta\}$, and $P_{p_i \ldots p_j^\Delta}$ the set of paths from p_i to any peer in p_j^Δ. The generalized mapping relates each concept C in S_i to a set of concepts C^Δ in S_j^Δ taken from the mappings in $P_{p_i \ldots p_j^\Delta}$, according to an *aggregated score* which expresses the semantic similarity between C and C^Δ. In this context, a concept in S_i can be associated to more than one concept in a schema S_{j_k} in S_j^Δ, since more than one path may exist between p_i and p_{j_k}. The following definition formalizes the notion of aggregation of the semantic paths starting from p_i and ending in any peer in p_j^Δ.

DEFINITION 4 (GENERALIZED SEMANTIC MAPPING). *Let p_i and p_j be two peers, not necessarily distinct, and g an aggregation function. A* generalized semantic mapping *between p_i and p_j is a fuzzy relation $M(S_i, S_j^\Delta)$ where each instance (C, C^Δ) is such that:*

- *C^Δ is the set of concepts $\{C_1, \ldots, C_h\}$ associated with C in $P_{p_i \ldots p_j^\Delta}$, and*
- *$\mu(C, C^\Delta) = g(\mu(C, C_1), \ldots, \mu(C, C_h))$.*

As to the function g, the following properties, which express the essence of the notion of aggregation [12], must hold: 1) g is *monotonic increasing* in all its arguments; 2) g is a *continuous* function; 3) g respects the *boundary conditions* $g(0,\ldots,0) = 0$ and $g(1,\ldots,1) = 1$. Moreover, aggregating operations on fuzzy sets are usually expected to satisfy two additional requirements: 4) g is a *symmetric* function of all its arguments, that is, $g(a_1,\ldots,a_m) = g(a_{\pi(1)},\ldots,a_{\pi(m)})$ for any permutation π on $[1,m]$; 5) g is an *idempotent* function, that is, $g(a,\ldots,a) = a$ for all $a \in [0,1]$.

The aggregation function g should be chosen conveniently to model the *semantic aggregation of semantic grades*. In fact, each resulting grade for a given concept should be representative of the semantic approximation given by the peer and its own subnetwork. Several choices are possible for g, for instance functions such as the min, the max, any generalized mean (e.g., harmonic and arithmetic means), or any ordered weighted averaging (OWA) function (e.g., a weighted sum) [12]. For instance, the generalized semantic mapping $M(S_1, S_3^{\Delta})$ between Peer1 and Peer3 of the reference example relates `main singer` with {`protagonist`, `main vocalist`} related to the paths Peer1-Peer3 and Peer1-Peer3-Peer5, respectively, whose membership grade based on the arithmetic means is $(0.7+0.35)/2$. As a further choice, we experienced a modified version of a function which is commonly used in the field of discrete choice analysis with application to travel demand [5]. In travel demand applications, it is often the case that the actual alternatives from which a decision maker chooses are unidentifiable, and aggregated geographical zones are used as the alternatives [5]. This is very close to our view, in that a peer should be able to choose a direction on the basis of the aggregated information about the paths which can be explored by following that direction. Our proposal is to adapt the function used in travel demand applications, which relies on the concept of *utility*, i.e., the amount of advantages in making a choice among several alternatives, for modelling the aggregation of the membership grades $\mu_{j_1},\ldots,\mu_{j_h}$. For convenience of notation each μ_{j_l} is an abbreviation for the grade $\mu(C, C_{j_l})$, with $l = 1\ldots h$, shown in Definition 4. The proposed function U satisfies the properties of an aggregation function:

$$U = \overline{\mu} + \frac{1}{\nu} \ln \left[\frac{1}{h} \sum_{l} e^{\nu(\mu_{j_l} - \overline{\mu})} \right] \quad (1)$$

where $\overline{\mu} = 1/h \sum_l \mu_{j_l}$ and ν is a positive scale parameter. As shown by the experiments, U proved to be a good function for aggregation as it adjusts the average value $\overline{\mu}$ with a measure of the variance among the elemental alternatives, in that it is particularly sensitive to the presence of elemental alternatives having high grades. For instance, it is able to distinguish the case $[0.9, 0.5, 0.1]$ from the case $[0.6, 0.5, 0.4]$ which have the same mean, but have values $U = 0.552$ and $U = 0.503$, respectively.

The Lemma below easily follows from the properties that an aggregation function must satisfy.

LEMMA 1. *A* generalized semantic mapping *between p_i and p_j is a fuzzy relation $M(S_i, S_j^{\Delta})$ which satisfies the following properties: 1) it is a 0-function, i.e., for each $C \in S_i$ it exists exactly one tuple C^{Δ} in the range such that $\mu(C, C^{\Delta}) \geq 0$; 2) it is reflexive, i.e., given $S_i = S_j^{\Delta}$, for each $C \in S_i$, $\mu(C, C) = 1$.*

SRI_{Peer1}	opera	main singer	title	...
Peer1	1.0	1.0	1.0	...
Peer2	0.6	0.4	0.3	...
Peer3	0.7	0.6	0.6	...

Figure 2: Sample SRI

3. SEMANTIC ROUTING INDICES

The membership grades given by generalized semantic mappings can be exploited for a wise propagation of a given query formulated over a peer p towards the most promising direction in the network, i.e., towards p's neighbors whose subnetworks are the most semantically related to the query. To this purpose, each peer p maintains a matrix which contains the membership grades between its schema and the schemas of its neighbors. This matrix is used as a routing index and it is named *Semantic Routing Index (SRI)*. More precisely, if p has n neighbors, its SRI has $n+1$ rows, where the first row refers to the knowledge on the local schema of peer p.

DEFINITION 5 (SEMANTIC ROUTING INDEX). *Let p be a peer with schema S=$\{C_1,\ldots,C_m\}$ and neighbors $p_1,\ldots,p_n$. The p's* Semantic Routing Index *is a matrix SRI of $n+1$ rows and m columns, such that each entry $SRI[i][j]$, for $i = 1\ldots n$ $j = 1\ldots m$, is the membership grade $\mu(C_j, C_j^{\Delta})$ of the instance (C_j, C_j^{Δ}) of the generalized semantic mapping $M(S, S_i^{\Delta})$.*

A sample of a portion of the Peer1's SRI of the reference example is shown in Fig. 2. The space required for storing a SRI at a peer is proportional to the number of the peer's neighbors, and thus quite modest w.r.t. the number of peers which usually join a PDMS. This makes our distributed-index mechanism scalable in a P2P context. Now, the main issue is how SRIs are used for routing queries in the network, and on how they are created and maintained. This is the subject of the following sections.

3.1 SRIs for Query Routing

When a peer p needs to forward a query q, it accesses its own SRI for determining the neighboring peers which are most semantically related to the concepts in q. For the sake of clarity, we start simple, and we assume the query q refers to a single concept C. The choice of the semantically best neighboring peers is done by evaluating the column of its SRI corresponding to C. In particular, the highest values in this column make the corresponding neighbors to be the selected peers. For instance, given the query `main singer` on Peer1 whose SRI is shown in Fig. 2, Peer3 will be preferred to Peer2. In the general case of a complex query involving more concepts, the choice of the best neighbors is given by applying scoring rules which, for each neighboring peer p_i, combines the corresponding grades in the SRI for all the corresponding concepts in q. How these values can be effectively combined is beyond the scope of this paper, and, as to this point, several works can be exploited (e.g., [8, 19]). Here, we make the simplifying assumption that an overall score is somehow obtained for a complex query, and we defer a deeper study of this issue to future work. Once the best neighboring peer p_i is found, the semantic mapping $M(S, S_i)$ is exploited to unfold the query q in q'. q' is then routed towards the subnetwork p_i^{Δ}, where, starting from p_i

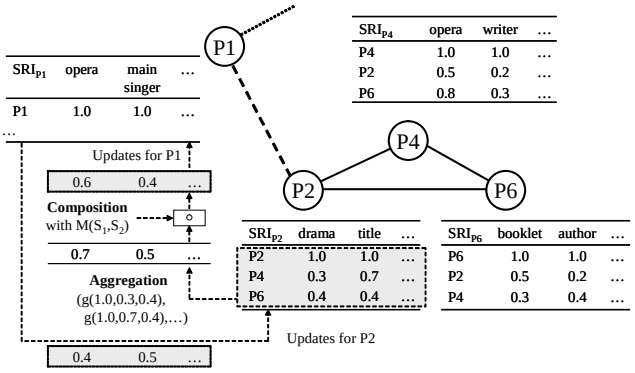


Figure 3: SRI evolution

which in turn evaluates q' and returns its local results to p, the process possibly reiterates.

3.2 Spreading Semantic Information in the Network

Since SRIs summarize the semantic information offered by the network, they change whenever the network itself changes. This may occur in response to either the joining/leaving of peers, or to changes in peers' schemas which possibly involve changes in the semantic mappings. We first focus our attention on the evolution of the PDMS's topology.

SRIs evolution is managed in an incremental fashion as follows. As a base case, the SRI of an isolated peer p having schema S is made of the single row $[1, \ldots, 1]$, i.e., it contains the membership grades of the concepts in S in the self mapping $M(S, S)$. This row expresses the semantic approximation offered by the subnetwork rooted in p, yet made of the only peer p. A simplification of the process of Peer1's SRI update when Peer1 connects to Peer2 is shown in Fig. 3 (P1 and P2 in the figure). When a peer connects to another peer, each one *aggregates* its own SRI SRI by rows, according to an aggregation function g. The result of this aggregation operation is a tuple $SRI^g = [\mu_1, \ldots, \mu_m]$. Each μ_j is the membership grade of concept C_j in the schema S of the peer to the fuzzy relation obtained by the aggregation of the SRI's rows, i.e., $\mu_j = g(SRI[0][j], \ldots, SRI[n][j])$ for $j = 1 \ldots m$. The so obtained fuzzy relation SRI^g and the schema S are then sent to the other peer.

After a peer, say p_i, receives such knowledge from the other peer, say p_j, a semantic mapping $M(S_i, S_j)$ is established between S_i and S_j. Then, p_i extends its SRI SRI_i with a new row for p_j. The membership grades of this newly created row are obtained in two steps: 1) $M(S_i, S_j)$ is composed with the aggregated SRI provided by p_j to obtain a fuzzy relation which expresses the extension of the semantic paths originating from p_j (represented by the aggregated SRI) with the connection between p_i and p_j; 2) the so obtained fuzzy relation is then aggregated with $M(S_i, S_j)$ to include the semantic path connecting p_i with p_j. More precisely,[2] $SRI_i[j][k] = g(M(S_i, S_j)(C_k, C'_k), M(S_i, S_j)(C_k, C'_k) \circ^I SRI^g[k])$, for $j = 1 \ldots m$.

Afterwards, both peers p_i and p_j need to inform their own reverse neighbors that a change occurred in the network and thus they have to update their SRIs accordingly. To this end, each peer, say p_i, sends to each reverse neighbor p_{i_k} an aggregate of its SRI, excluding the p_{i_k}'s row, i.e., $\mu_j = g(SRI_i[0][j], \ldots, SRI_i[k-1][j], SRI_i[k+1][j], \ldots, SRI_i[n][j])$. When p_{i_k} receives such aggregated information, it updates the i-th row of its SRI by recomputing the membership values as discussed above.

As the values stored in the SRIs are computed incrementally, we have to show that they actually correspond to the membership values of the generalized semantic mappings.

THEOREM 1. *Whenever the aggregation function g is associative and the composition function I is distributive w.r.t the aggregation function g, the process described above is correct, i.e. when applied to the Semantic Routing Index of peer p_i, $SRI_i[j]$ is the generalized semantic mapping $M(S_i, S_j^{\triangle})$ between p_i and p_j.*

Due to the lack of space, we only give a sketch of the proof. The proof is by induction on the construction process and it relies on the properties of the composition and aggregation functions. Notice that in order to make the function U shown in Section 2.3 associative, it suffices to maintain the number of aggregated paths as additional information.

Disconnections are treated in a similar way as connections. When a node disconnects from the network, each of its neighbors must delete the row of the disconnected peer from its own SRI and then inform the remaining neighbors that a change on its own subnetwork has occurred by sending new aggregates of its SRI to them. A similar procedure applies in case of modifications of the semantic knowledge maintained at each peer, for instance when a new concept is added to the peer's schema. When many changes occur in the PDMS, a careful policy of updates propagation may be adopted. For instance, when changes has a little impact on its SRI, a peer may also decide not to notify the network. This would reduce the amount of exchanged messages as well as the computational costs due to SRI manipulation. We are aware that the definition of such policies, recommended for highly dynamic PDMSs, is a fundamental issue, so we plan to deal with it in future work.

4. SRI IN ACTION

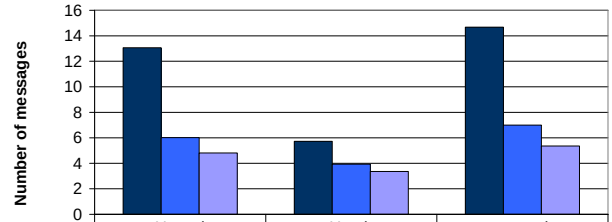
In this section we discuss a selection of experiments we performed to test the effectiveness of SRIs for query routing.

4.1 Experimental Setting

For our experiments we used a simulation framework able to reproduce the main conditions characterizing a PDMS environment. In particular, we employed SimJava 2.0, a discrete, event-based, general purpose simulator, which allowed us to evaluate the impact of exploiting SRIs for query routing. Through this framework we modelled scenarios corresponding to networks of semantic peers, each with its own schema describing a particular reality. We chose peers belonging to different semantic categories, where the schemas of the peers in the same category describe the same topic from different points of view. As in [21], the schemas are derived from real world-data sets, collected from many different available web sites, such as the DBLP Computer Society Bibliography and the ACM SIGMOD Record and enlarged with new schemas created by introducing structural and terminological variations on the original ones. Then,

[2]Note that, because of the boundary condition of the t-norm I used for composition, and being g an idempotent function, the base case of connection of an isolated peer p_j to a peer p_i results in $SRI_i[j][k] = M(S_i, S_j)(C_k, C'_k)$.

we distributed these schemas in the network in a clustered way, i.e. the schemas belonging to the same semantic category are more likely to belong to peers connected through semantic mappings. This reflects realistic scenarios where nodes with semantically similar content are often clustered together. As to the topology of the semantic network of mappings connecting the peers, we tested our techniques on different alternatives: We started with tree networks and then explored more realistic and uncontrolled ones generated with the BRITE topology generator tool [1]. In these last cases, in order to avoid the presence of cyclic paths in the SRI updates propagation, when a peer connects to the network a cycle detection mechanism based on global unique identifiers, as in [7], is adopted.



	U-prod	U-sub	sum-sub
Random	13.06	5.73	14.68
SemMap	6.02	3.93	7
SRI	4.81	3.36	5.36

Figure 4: Selection of functions

4.2 Effectiveness Evaluation

In order to evaluate the effectiveness of exploiting SRIs for query routing, we simulated the querying process by instantiating different queries on randomly selected peers and concepts and propagating them until a stop condition is reached. We considered two alternatives: stopping the querying process when a given number of peers (*messages*) has been queried and measuring the quality of the results (*satisfaction*) or, in a dual way, stopping when a given satisfaction is obtained and measuring the required number of messages. Satisfaction is a specifically introduced quantity that grows proportionally to the goodness of the results returned by each queried peer. Each contribution is computed by composing the semantic mappings scores of the traversed peers. The search strategy employed is the depth-first search (DFS): Each peer receiving a query produces an answer on its local schema, then selects its most promising neighbor among the unvisited ones and forwards the query to it. In this section, we compare our neighbor selection mechanism based on SRIs (*SRI*) with a mechanism where the selection of the next neighbor to be visited is based on the semantic mapping values (*SemMap*) and with a baseline corresponding to a random strategy (*Random*). Notice that all the results we present are computed as a mean on several hundreds query executions.

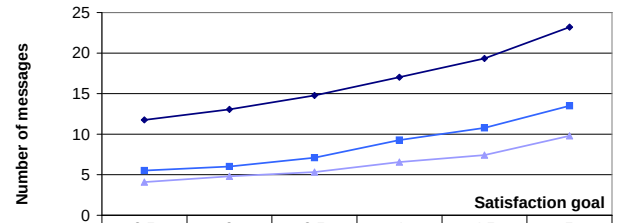
We start by considering tree topologies and a satisfaction goal as stop condition. Our first aim is to evaluate if the information provided by the SRIs, built using different combinations of the proposed functions for aggregation and composition, is useful to perform a good routing mechanism. In Figure 4 we show three scenarios describing the behaviour of the routing mechanism exploiting the Random, SemMap and SRI neighbor selection. Besides the ones presented in the previous sections, we also considered the sum for aggregation and a variant of the difference for composition (sub in the figure). The three scenarios differ for the choice of functions and the satisfaction goal, but in each of them we have similar results: The SemMap strategy is by far better than the Random strategy, but the SRI one requires even fewer messages to reach the goal. In particular, the first scenario involving the U and the product functions shows the higher number of saved messages (from 6.02 to 4.81, which leads to an improvement of 25%); thus, for the following tests we will refer to this pair of functions.

The next step is to deepen the analysis of the behaviour of the routing strategies when we gradually vary the stop conditions. Figure 5-a shows the trend of the number of required messages for a given satisfaction goal, while, from a dual perspective, Figure 5-b shows the trend of the obtained satisfaction at a given message limit. From both points of view, the Random strategy is outdistanced by the other two. Further, the difference between the SemMap and SRI performance appears closer at the initial part of the graphs but becomes increasingly more significant at growing stop conditions. This means that SRIs are indeed able to discriminate better subnetworks to explore and consequently increase the satisfaction at each step in a more substantial way. Nevertheless, tree topologies may not be considered completely realistic for a PDMS setting and may facilitate the SRI routing process, because in such kind of topologies different subnetworks are not overlapped and it is consequently probably easier to identify the better ones.

For these reasons, we deepened our tests by introducing realistic network topologies, also involving redundant and cyclic paths. The results referring to these situations are shown in Figure 6, which presents the graphs of Figure 5 for the new environments. Even though the distance between SRIs and SemMap is slightly reduced due to the complications introduced, we can see that, despite the new unfavourable scenarios, the trend of these graphs are roughly similar to the previous ones. Specifically, even in this case, we can observe that the SRI curves are clearly separated from the others, reflecting that the SRIs' ability to identify the best subnetworks to explore facilitates a faster retrieval of the highest ranked results.

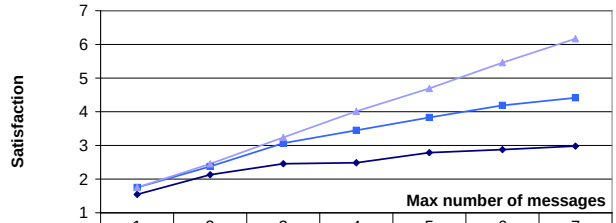
Figure 7 shows another typology of experiments aiming to verify how the performances of the SRI routing mechanism are affected when we vary the update horizon limiting the part of network the SRI scores summarize. The curves represent the messages trend for growing values of the horizon, starting from the baseline 1 which corresponds to the SimMap case, and for two different satisfaction goals. Observing the graph, it is clear that increasing the horizon allows us to perform a better routing mechanism, because it relies on more precise information. In particular, from our tests, horizon extensions up to 8 clearly provide significant benefits, then the results appear to stabilize (see Figure 7). Notice that the use of an horizon, limiting the updates propagation for the SRIs scores, clearly introduces a kind of approximation on the information stored, but it is also useful in limiting the SRI maintenance costs. Therefore, due to the above considerations, since higher horizons lead to larger propagation costs, we consequently estimate that an horizon value of 8 represents a good trade-off (this is also the value at which all tests in this section have been performed).

Finally, the last type of test we present explores a possible



	2.5	3	3.5	4	4.5	5
Random	11.76	13.06	14.78	17.02	19.34	23.21
SemMap	5.51	6.02	7.11	9.28	10.78	13.51
SRI	4.09	4.81	5.32	6.57	7.43	9.79

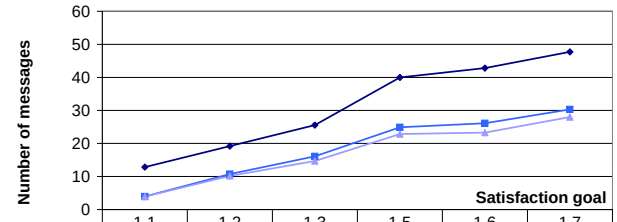
(a) Required messages trend for a given satisfaction goal



	1	2	3	4	5	6	7
Random	1.5432	2.1322	2.4567	2.4876	2.7852	2.8754	2.9811
SemMap	1.7529	2.3756	3.0668	3.4495	3.8308	4.1891	4.4143
SRI	1.7529	2.4559	3.2389	4.0119	4.6949	5.4579	6.1709

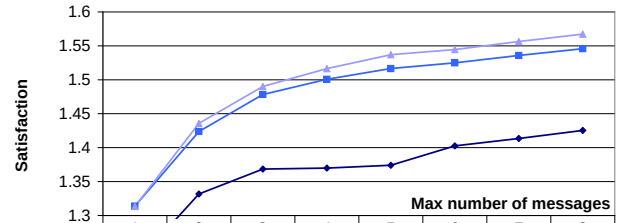
(b) Obtained satisfaction trend at a given message limit

Figure 5: Tree networks results



	1.1	1.2	1.3	1.5	1.6	1.7
Random	12.84	19.17	25.57	39.99	42.8	47.73
SemMap	3.93	10.74	16.09	24.89	26.13	30.27
SRI	3.91	10.18	14.7	22.84	23.29	28

(a) Required messages trend for a given satisfaction goal



	1	2	3	4	5	6	7	8
Random	1.2281	1.3317	1.3685	1.3699	1.3741	1.4026	1.4135	1.4254
SemMap	1.3139	1.4238	1.4783	1.5008	1.5165	1.5249	1.5358	1.5458
SRI	1.3133	1.4359	1.4902	1.5164	1.5369	1.5445	1.5564	1.5673

(b) Obtained satisfaction trend at a given message limit

Figure 6: Real networks results

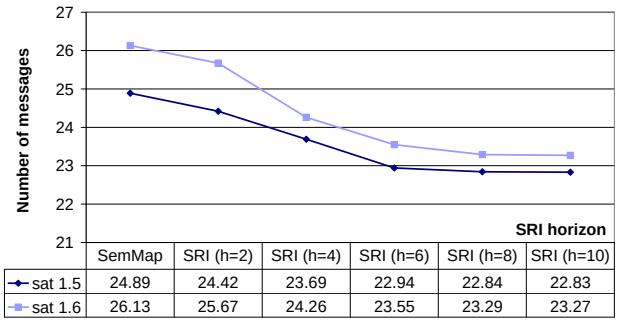
enhancement on the routing process, involving a mechanism of pruning on the selection of the paths to follow: We introduce a threshold for the selection of neighbors to which propagate the queries, preventing the exploration of those paths characterized by small mapping and/or SRI scores and consequently leading to uninteresting results. The graph shows the results for both SemMap and SRI routing strategy, expressing the number of messages necessary to reach a given satisfaction goal when we apply three different pruning thresholds. Notice that when we use a zero threshold, and consequently apply no pruning mechanism, we obtain the same results presented earlier in the section. As can be seen, increasing the threshold leads to significant savings in the number of messages for both strategies. However, performing pruning on the basis of SRIs appears to perform better in every situation, showing a small but consistent number of saved messages w.r.t SemMap.

5. RELATED WORK AND CONCLUDING REMARKS

As envisioned by the Semantic Web, the need of complementing the Web with more semantics has spurred much efforts towards a rich representation of data. To this end, knowledge representation languages (e.g., XML, RDF, and OWL) has flourished in recent years. In this view, peer data management systems (PDMSs) have been introduced as a solution to the problem of large-scale sharing of semantically rich data [10]. Indeed, a key challenge when querying a large set of peers is query routing, i.e., the capability of selecting a small subset of relevant peers to forward a query to. Much research work in the P2P area has focused on this issue [20, 11, 7, 6, 9, 13, 16, 18, 22]. Some of these works discuss id/keyword-based search of documents [20, 6, 16], some assume a common vocabulary/ontology is shared by peers in the network [7, 9], some address scalability of query routing by means of a properly tailored super-peer topology for the network [18], or by adapting their own semantic topology according to the observation of query answering [22].

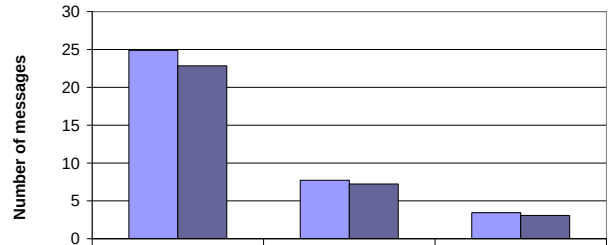
Most of these proposals are based on IR-style and machine-learning techniques [7, 6, 13, 16, 22]. Basically, they utilize measures that rely on keyword statistics, on the probability of keywords to appear into documents, on the number of documents that can be found along a path of peers, on caching/learning-from the number of results returned for a query. Then, all of them (but [7]) provide routing techniques which either assume distributed indices which are indeed conceptually global [20, 16], or support completely decentralized search algorithms which, nevertheless, exploit information about neighboring peers only. More precisely, the only work [7] proposes a routing mechanism which does not limit the peer's capability of selecting peers to the information available at a 1-hop horizon, rather it extends this view by using summaries of subnetworks' content to provide a *direction* to send a query to.

Nevertheless, querying a PDMS is different than querying a P2P system, primarily because of the presence of heterogeneous schemas at the peers. On the other hand, the novelty in a PDMS lies in its ability to exploit the transitive relationships among such schemas for query answering [11, 10].



	SemMap	SRI (h=2)	SRI (h=4)	SRI (h=6)	SRI (h=8)	SRI (h=10)
sat 1.5	24.89	24.42	23.69	22.94	22.84	22.83
sat 1.6	26.13	25.67	24.26	23.55	23.29	23.27

(a) ...SRI horizon



	Pruning off	Pruning med	Pruning high
SemMap	24.89	7.73	3.45
SRI	22.84	7.2315	3.09

(b) ...pruning

Figure 7: Additional results for real networks: Impact on required messages of...

In this scenario, our work aims to support query routing in a PDMS, and it appears to be the first having this purpose. The main differences between our proposal and the P2P techniques discussed above are: 1) We do not assume any global characterization of documents in the network; 2) We move in a PDMS scenario, then assuming the presence of schemas describing the content of peers' data, as well as pairwise semantic relationships between the peers' schemas; 3) We make a schema-based rather than a key(word)-based search; 4) inspired to [7], we rely on fully distributed semantic indices, called SRIs, which summarize the *semantics* (rather than the number of documents as in [7]) that can be retrieved following a given direction in the network; 5) in order to cope with schema heterogeneity, we rank (subnetwork of) peers according to the *semantic similarity* occurring between concepts in the peers' schemas. The experiments we conducted on a simulation environment led to very encouraging results.

As a future work, SRIs could be integrated in a more general framework together with other approaches such as [7, 16] which are orthogonal to ours, and which cover complementary aspects such as knowledge on quantitative information, as well as on novelty of results, so as to blend different dimensions a peer can be queried on. Then, as also stated in [22], the *best* peer has been understood as a peer that has the most knowledge. Other aspects one might include in the evaluation of peers are properties like latency, costs, etc.

6. REFERENCES

[1] BRITE. `http://www.cs.bu.edu/brite/`.

[2] Gnutella. `http://www.gnutella.com/`.

[3] Napster. `http://www.napster.com/`.

[4] M. Arenas, V. Kantere, A. Kementsietsidis, I. Kiringa, R. Miller, and J. Mylopoulos. The hyperion project: from data integration to data coordination. *SIGMOD Record*, 32(3):53–58, 2003.

[5] M. Ben-Akiva and S. Lerman. *Discrete Choice Analysis: Theory and Application to Travel Demand*. The MIT Press, 1985.

[6] B. Cooper. Using Information Retrieval Techniques to Route Queries in an InfoBeacons Network. In *Proc. of DBISP2P*, 2004.

[7] A. Crespo and H. Garcia-Molina. Routing Indices for Peer-to-Peer Systems. In *Proc. of ICDCS*, 2002.

[8] R. Fagin. Combining Fuzzy Information: an Overview. *SIGMOD Record*, 31(2):109–118, 2002.

[9] P. Haase, R. Siebes, , and F. van Harmelen. Peer Selection in Peer-to-Peer Networks with Semantic Topologies. In *Proc. of ICNSW*, 2004.

[10] A. Halevy, Z. Ives, J. Madhavan, P. Mork, D. Suciu, and I. Tatarinov. The Piazza Peer Data Management System. *IEEE TKDE*, 16(7):787–798, July 2004.

[11] M. H. K. Aberer, P. Cudré-Mauroux. The Chatty Web: Emergent Semantics Through Gossiping. In *Proc. of WWW*, 2003.

[12] G. J. Klir and B. Yuan. *Fuzzy Sets and Fuzzy Logic: Theory and Applications*. Prentice Hall, 1995.

[13] G. Koloniari and E. Pitoura. Content-Based Routing of Path Queries in Peer-to-Peer Systems. In *Proc. of EDBT*, 2004.

[14] J. Madhavan, P. A. Bernstein, A. Doan, and A. Y. Halevy. Corpus-based schema matching. In *Proc. of ICDE*, 2005.

[15] S. Melnik, H. Garcia-Molina, and E. Rahm. Similarity Flooding: A Versatile Graph Matching Algorithm and Its Application to Schema Matching. In *Proc. of ICDE*, 2002.

[16] S. Michel, M. Bender, P. Triantafillou, and G. Weikum. IQN Routing: Integrating Quality and Novelty in P2P Querying and Ranking. In *Proc. of EDBT*, 2006.

[17] W. Nejdl, B. Wolf, S. Staab, and J. Tane. EDUTELLA: Searching and Annotating Resources within an RDF-based P2P Network. In *Proc. of WWW Intl. Workshop on the Semantic Web*, 2002.

[18] W. Nejdl et al. Super-Peer-Based Routing and Clustering Strategies for RDF-Based Peer-to-Peer Networks. In *Proc. of WWW*, 2003.

[19] W. Penzo. Rewriting Rules To Permeate Complex Similarity and Fuzzy Queries within a Relational Database System. *IEEE TKDE*, 17(2):255–270, 2005.

[20] I. Stoica, R. Morris, D. Karger, F. Kaashoek, and H. Balakrishnan. Chord: A Scalable Peer-To-Peer Lookup Service for Internet Applications. In *Proc. of SIGCOMM*, 2001.

[21] I. Tatarinov and A. Halevy. Efficient Query Reformulation in Peer Data Management Systems. In *Proc. of SIGMOD*, 2004.

[22] C. Tempich, S. Staab, and A. Wranik. REMINDIN': Semantic Query Routing in Peer-to-Peer Networks Based on Social Metaphors. In *Proc. of WWW*, 2004.