The Knowledge Engineering Review, Vol. 23:1, 51–72.
doi:10.1017/S0269888907001257 Printed in the United Kingdom

Semantically routing queries in peer-based systems: the H-Link approach

STEFANO MONTANELLI and SILVANA CASTANO

Dipartimento di Informatica e Comunicazione, Università degli Studi di Milano. Via Comelico 39, 20135 Milano, Italy;
e-mail: castano@dico.unimi.it
e-mail: montanelli@dico.unimi.it

Abstract

A challenging issue to advance the existing P2P semantic routing protocols is related to the capability of developing mechanisms for focused selection of the query recipients by taking into account a semantically rich description of the context of each peer. In this article, we present the *H-Link semantic routing approach* designed to exploit the results of an ontology matchmaking process for providing a semantic overlay network where peers having similar contexts are recognized and interlinked as *semantic neighbors*. In particular, H-LINK aims at advancing the existing semantic routing protocols by combining ontology-based peer context descriptions and ontology matching techniques for providing query forwarding on a real semantic basis, in a completely decentralized way.

1 Introduction

The need of sharing data and resources to foster semantic collaboration is a key issue at the current stage of development of P2P networks and open distributed systems in general (Androutsellis-Theotokis & Spinellis, 2004). In this scenario, peers join the system by providing their own context and need to cooperate by matching their respective knowledge, usually described through an ontology, with the aim to discover similar partners and thus to enforce effective resource sharing. In order to provide scalable infrastructures for peer communications, P2P semantic routing protocols are being proposed with the aim of selecting query recipients on a semantic basis, by identifying those peers that are most likely to provide matching results according to the query content (Staab *et al*., 2004; Mandreoli *et al*., 2006; Haase *et al*., 2007). However, most of the existing approaches rely on a rather simplifying assumption of a global repository of knowledge where mappings among the distributed peer resource descriptions are maintained. This centralized mapping knowledge is then exploited to select the best recipients for a given query by the routing mechanism. In some other approaches, the knowledge model supported by the peers is kept quite poor (i.e. metadata rather than ontologies) in order to reduce the complexity of the matching process. In such cases, syntactic matching techniques (e.g. string- and keyword-based techniques) are generally employed to compute the similarity among the resource descriptions of the different peers, thus leading to a poor accuracy in query recipient selection (Tsoumakos & Roussopoulos, 2003; Borch & Vognild, 2004; Zeinalipour-Yazti *et al*., 2005). A challenging issue to advance the state of the art is thus related to the capability of developing a routing mechanism where a semantically rich description of the context of each peer is explicitly taken into account for selecting the query recipients without relying on a shared global repository of predefined mapping knowledge.

In this article, we present the *H-Link semantic routing mechanism* designed to exploit the results of an ontology matchmaking process for providing a semantic overlay network where peers having similar contexts are recognized and interlinked as *semantic neighbors*. In particular, H-Link aims at advancing the existing semantic routing protocols by combining ontology-based peer context descriptions and ontology matching techniques for providing query forwarding on a real semantic basis, in a completely decentralized way. Furthermore, H-Link aims at enforcing scalability in query forwarding by introducing a *credit-based* mechanism where the approximate number of desired replies is specified rather than the non-scalable number of hops to cross.

The article is organized as follows. In Section 2, we discuss the role of ontologies and context for enforcing effective semantic routing and we introduce the foundational aspects of H-Link. Details regarding the use of ontologies and ontology matching techniques in H-Link are provided in Section 3. The H-Link semantic routing approach is then presented in Section 4. In Section 5, experimental results on H-Link are provided. Furthermore, related work and original contributions of H-Link are discussed in Section 6. Finally, concluding remarks and future work are presented in Section 7.

2 The role of ontologies and peer contexts in semantic query routing

In the literature related to P2P systems, the notion of context is mostly related to the idea of formally representing the peer perspective when joining the network for sharing purposes (O'Brien & Abidi, 2006). Furthermore, in more recent schema-based P2P systems, the use of ontologies is becoming a key feature for providing an expressive representation of peer resources. In this respect, the peer context is used to advertise the peer interests to the network in order to foster the identification of nodes capable of providing knowledge that matches a given target concept of interest (Nejdl *et al*., 2002; Broekstra *et al*., 2003; Halevy *et al*., 2004; Castano *et al*., 2006d). Furthermore, the peer context can be used to support the formation of semantic communities of peers with similar interests (Bonifacio *et al*., 2003; Castano & Montanelli, 2005). In such systems, the notion of peer context usually coincides with a (formal) representation of the peer knowledge (e.g. an ontology) regarding the peer shared resources.

Regarding more specifically P2P semantic query routing, the notion of peer context is only partially supported. As we will discuss in detail in Section 6, most of the existing P2P semantic routing approaches provide a rather implicit definition of peer context, often identified as the set of keywords associated with the shared resources/files (Joseph, 2002; Borch & Vognild, 2004; Zeinalipour-Yazti *et al*., 2005). In some recent approaches to semantic routing, the use of ontologies for peer-knowledge representation, and thus of peer context representation, is also being appearing (Staab *et al*., 2004; Mandreoli *et al*., 2006). In this respect, the H-Link semantic routing mechanism uses a peer context explicitly defined in form of a *peer ontology*. In fact, the peer ontology in H-Link provides a formal description of the peer context in terms of (1) the local knowledge the peer intends to share with the other nodes (i.e. the semantic description of the resources to share) and (2) the network knowledge of the peer, that is the knowledge about its semantic neighbors progressively discovered as long as interactions occur in the network. Such a network knowledge is exploited in H-Link for enforcing semantic query routing by selecting as query recipients the peers with the highest chance to provide interesting query replies.

2.1 Main features of H-Link

The H-Link mechanism has been conceived to work in a peer-based system where peers act as independent agents and no centralized authorities are defined to manage a comprehensive view of the resources shared by all the nodes of the system. The key idea of H-Link is to exploit the results of knowledge discovery interactions to train the behavior of the routing mechanism. To this end, peers are connected through matching-based *confidence* measures that keep track of the semantic affinity among the ontologies of different peers. As a result, peers are organized in a semantic overlay network where peers having similar knowledge are interlinked as *semantic neighbors*. The following main features characterize the H-Link mechanism.

- *Dynamic knowledge discovery approach.* In H-Link, peers interact by submitting discovery queries with the aim to identify relevant partners with respect to one or more target concepts of interest. Receiving a discovery query, a peer evaluates whether it is capable of providing concepts matching the target request. According to the results of the matching process, the list of discovered matching concepts and associated semantic affinity values is replied to the requesting peer. In H-Link, the replying nodes are linked by the requesting peer as semantic neighbors and the returned semantic affinity values are exploited to set the level of confidence of each semantic neighbor with respect to the discovered matching concepts. This way, as long as a peer learns about the network contents through discovery queries, its network knowledge gradually evolves to reflect its newly acquired semantic neighbors.
- *Peer context and ontologies.* In H-Link, both queries and peer resources are specified in terms of ontological descriptions. In particular, each query contains a list of target concept(s) of interest with possible properties and semantic relations that further specify the request. Furthermore, each peer joining the system provides a peer ontology describing the peer context, namely the knowledge the peer brings to the network and the knowledge the peer perceives from the network. As the use of peer ontologies is a foundational aspect of H-Link, a detailed description of the peer ontology definition will be provided in Section 3.1.
- *Ontology matching techniques.* In H-Link, each peer is capable of performing ontology matching through the use of a semantic matchmaker. Ontology matching is employed by a peer during the knowledge discovery process in order to assess whether it can provide matching concepts in reply to an incoming discovery query. Furthermore, ontology matching is also exploited in H-Link to select the recipients of a query according to the expected semantic affinity between the query contents and the peer ontology of the receiving peer. The H-Match semantic matchmaking system is currently used to support H-Link (Castano *et al.*, 2006c). As ontology matching techniques are a foundational aspect of H-Link, a detailed description of the ontology matching techniques of H-Match will be provided in Section 3.2.

2.2 *Motivating and running example*

As an example of dynamic knowledge discovery with H-Link, we consider the scenario of Figure 1. In this scenario, each peer is independent and joins the system by providing its own peer ontology. In Figure 1, we suppose that peer A is interested in discovering peers capable of providing resources semantically related to the publishing domain. To this end, peer A composes and submits to the system a discovery query Q1 containing the target concepts of interest Publication and Book with the properties year and author, respectively. Moreover, Book is specified as a subclass of Publication. Receiving the query Q1, a peer (i.e. peer B, peer C, and peer D) invokes its semantic matchmaker (i.e. H-Match) to compare the description of the target concepts in the query against its own peer ontology. According to the results of the matching process, peer B and peer D send back to the requesting peer A a ranked list of concepts matching the target, and for each entry, the calculated semantic affinity value SA, with $SA \in [0, 1]$. In particular, peer B replies with the Volume matching concept since SA(Book, Volume) = 0.83 in H-Match, while peer D sends back two matching concepts, namely Newspaper and Magazine, with SA(Publication, Newspaper) = 0.67 and SA(Book, Magazine) = 0.539, respectively. On the other hand, peer C does not reply to peer A as no matching concepts are identified. The query replies represent the discovered knowledge of peer A that can be exploited to decide whether to further interact with the answering peers in order to access their resources for data sharing.

Without H-Link, the discovery process relied on conventional P2P infrastructures and associated routing protocols for query propagation in the network (e.g. flooding). In the following sections, we show how the discovered knowledge can be further exploited in H-Link for semantic routing purposes by enforcing query forwarding according to peer ontology similarities rather than to the mere network topology.

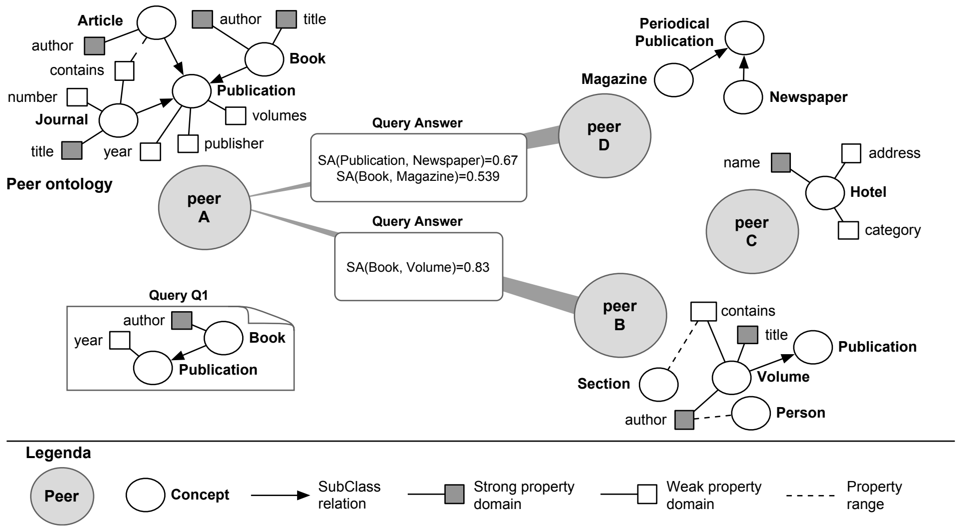

Figure 1 Example of dynamic knowledge discovery in H-LINK

3 Peer ontology architecture and matching techniques

In this section we discuss the reference architecture of a peer ontology and we present the H-MATCH ontology matchmaking techniques.

3.1 The peer ontology architecture

The context knowledge of a peer is described through a peer ontology. At the conceptual level, the peer ontology is organized in a two-layer architecture where the upper layer represents the *content knowledge* and the lower layer represents the *network knowledge* of a peer.

The content knowledge layer. The content knowledge layer of a peer ontology describes the resources the peer brings to the network. In abstract terms, the content knowledge layer contains concepts C, properties P, and semantic relations SR defined as follows.

- *Concept*. A concept $c \in C$ is defined as a pair of the form $c = (n_c, P_c)$, where n_c is the concept name, and P_c is a possibly empty set of properties of c.
- *Property*. A property $p \in P$ is defined as a pair of the form $p = (n_p, PC_p)$, where n_p is the property name, and PC_p is a set of property constraints. Each property constraint associates a property p with a concept c, by specifying the minimal cardinality and the property value v_p of p in c. A property constraint $pc_p \in PC_p$ is a 3-tuple of the form $pc_p = (c, k_p, v_p)$, where c is a concept, $k_p \in \{0,1\}$ is the minimal cardinality associated with p when applied to c, and v_p is the value associated with p when applied to c, and can be a datatype $\mathrm{d}t_p$ or a reference name. We call strong properties the properties with $k_p = 1$, and weak properties the ones with $k_p = 0$.
- *Semantic relation*. A semantic relation $sr \in \mathrm{SR}$ is defined as a binary relation of the form $sr = (c, c', rn)$, where c and c' are concepts and rn is the name of the relation holding between them.

The content knowledge layer is defined according to the constructs of the ontology specification language adopted for peer ontology description. In concrete H-LINK applications, we considered Semantic Web peer ontologies specified with the OWL standard language (Smith *et al.*, 2004). In this case, a concept corresponds to an OWL class, a property is defined through an ObjectProperty or a DatatypeProperty according to the corresponding range (i.e. an OWL class or a datatype, respectively), a property constraint is translated with OWL restrictions, and a semantic relation can be equivalentClass or subClassOf.

The network knowledge layer. The network knowledge layer of a peer ontology describes the knowledge that the peer has of the semantic neighbors it has interacted with. This layer is seen as a set of neighbor peers NP that are connected with the concepts in the content knowledge layer through affinity links AL.

- *Neighbor peer*. A neighbor peer is an ontological description of the properties of a semantic neighbor that has been identified during the knowledge discovery process. A neighbor peer $np \in$ NP is defined as a triple of the form $np = (n_{np}, P_{np}, sns_{np})$, where n_{np} is the unique identifier of the neighbor peer in the network, P_{np} is a set of properties describing the network features of the peer (e.g. IP address, bandwidth), and $sns_{np} \in [0,1]$ is the measure of the '*semantic neighborness*' of np, which provides a comprehensive estimation of the np relevance computed on the basis of the level of semantic affinity of the query replies returned by np during past interactions.
- *Affinity link*. An affinity link is a relation holding between a neighbor peer np and a concept c to capture the fact that np is capable of providing matching concepts for c. An affinity link $al \in$ AL is defined as a triple of the form $al = (c, np, cf)$, where $c \in C$ is a concept in the content layer, $np \in$ NP is a neighbor peer in the network layer, and $cf \in [0, 1]$ is the confidence value of al which corresponds to the semantic affinity value computed between between c and the matching concepts in peer ontology of the peer np.

Example. As an example, in Figure 2, we consider a portion of the peer ontology of the peer A after the knowledge discovery interaction described in Figure 1. In the example, we assume to deal with an OWL ontology and we rely on a graph-based representation where the graph nodes denote concepts and properties, while the edges denote the semantic relations between concepts. In this example, peer B and peer D have answered to query Q1, then the corresponding neighbor peers are inserted in the network knowledge layer. According to the query reply of peer B, an affinity link with a confidence value of 0.83 is defined to connect the neighbor peer B with the Book concept in the content knowledge layer. As two matching concepts are returned in the query answer of peer D (i.e. Newspaper, Magazine), two affinity links are defined by connecting the neighbor peer D with the concepts Publication and Book in the content knowledge layer, and by setting a confidence value of 0.067 and 0.539, respectively. The values of semantic neighborness associated with peer B and peer D are 0.83 and 0.61, respectively. The values of confidence and semantic neighborness are calculated on the basis of the semantic affinity values of the concepts returned by peer B and peer D in their respective query replies. A detailed description of confidence and semantic neighborness computation is provided in Section 4 where we discuss the process of network knowledge definition.

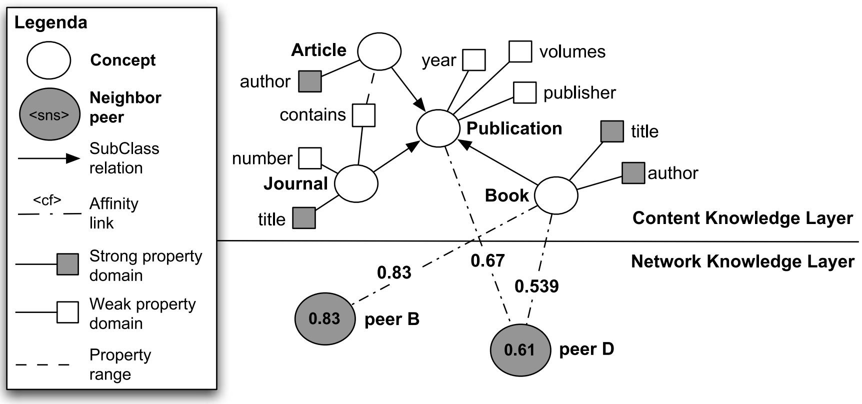

Figure 2 A portion of the peer ontology of peer A

3.2 *Ontology matching with H-Match*

To determine confidence and semantic neighborness measures, H-LINK relies on the H-MATCH ontology matchmaking system. H-MATCH performs ontology matching by deploying four different *matching models* spanning from surface to intensive matching, with the goal of providing a wide spectrum of metrics suited for dealing with many different matching scenarios that can be encountered in comparing concept descriptions of real ontologies. H-MATCH takes two ontologies as input and returns the mappings that identify matching concepts in the two ontologies, namely the concepts with the same or the closest intended meaning. In H-MATCH we perform similarity analysis through a combination of affinity metrics to determine a measure of semantic affinity in the range [0,1]. A threshold-based mechanism is enforced to set the minimum level of semantic affinity required to consider two concepts as matching concepts. Given two concepts c and c', H-MATCH calculates a semantic affinity value SA(c,c') as the linear combination of a linguistic affinity value LA(c,c') and a contextual affinity value CA(c,c'). The H-MATCH linguistic affinity provides a measure of similarity between two ontology concepts c and c' computed on the basis of their linguistic features (i.e. concept names). For the linguistic affinity evaluation, H-MATCH relies on a thesaurus of terms and terminological relationships automatically extracted from the WordNet lexical system. The H-MATCH contextual affinity provides a measure of similarity by taking into account the contextual features of the ontology concepts c and c'. The context of a concept can include properties, semantic relations with other concepts, and property values. The context can be differently composed to consider different levels of semantic complexity, and four matching models, namely, *surface*, *shallow*, *deep*, and *intensive*, are defined to this end. In the surface matching, only the linguistic affinity (i.e. no contextual features at all) between the concept names of c and c' is considered to determine concept similarity. In the shallow, deep, and intensive matching, also contextual affinity is taken into account to determine concept similarity. In particular, the shallow matching computes the contextual affinity by considering only the properties of c and c'. Deep and intensive matching extend the depth of concept context for the contextual affinity evaluation of c and c', by considering also semantic relations with other concepts (deep matching model) as well as property ranges (intensive matching model), respectively. A comprehensive semantic affinity value SA(c, c') is evaluated as the weighted sum of the linguistic affinity value and the contextual affinity value, that is:

$$\mathrm{SA}(c,\, c') = W_{\mathrm{LA}}{\cdot}\mathrm{LA}(c,\, c') + (1-W_{\mathrm{LA}}){\cdot}\mathrm{CA}(c,\, c') \tag{1}$$

where W_{LA} is a weight expressing the relevance assigned to the linguistic affinity in the semantic affinity evaluation process.

Finally, a *matching policy* MP is defined in H-MATCH to configure the current execution of the matchmaker for a given matching case. A matching policy is a 4-tuple of the form $\mathrm{MP} = \langle mm, W_{\mathrm{LA}}, t\rangle$, where: $mm \in \{$surface, shallow, deep, intensive$\}$ denotes the matching model to be used for H-MATCH execution, $W_{\mathrm{LA}} \in [0,\, 1]$ denotes the linguistic affinity weight, and $t \in (0,\, 1]$ denotes the matching threshold.

Considerations on H-Match. H-MATCH has been extensively tested on several real ontology matching cases in order to evaluate the matching models with respect to performance and quality of results (Castano *et al.*, (2006a, 2006c). By analyzing the obtained results, we note that the most accurate and precise results are achieved using the deep and intensive matching models, provided that the ontology descriptions are detailed enough. On the other side, we note that the best performance in terms of computation time are achieved with the surface and shallow matching models. For semantic routing purposes, the computation time of the semantic affinity evaluation is a crucial factor and needs to be performed as fast as possible in order to avoid bottlenecks. To this end, possible lacks in matching precision and accuracy can be admitted in turn of rapid response time during the semantic affinity evaluation. For this reason, the shallow matching model is selected as a default in H-LINK. We note that a number of ontology matching tools and techniques are currently available (Giunchiglia & Shvaiko, 2003; Noy & Musen, 2003; Euzenat *et al.*, 2004;

Ehrig & Staab, 2004). However, the advantage of H-Match is that it is specifically conceived to work in open distributed systems by relying on its flexible matching models that allow to dynamically configure the tradeoff between performance and accuracy according to the requirements of the considered matching scenario.

Provided that a dynamic and flexible configuration is supported, other existing matching tools can however be used for satisfying the ontology matching purposes of H-Link (Shvaiko, 2006). In the remainder of the article, we assume to work with H-Match for supporting semantic routing in H-Link.

4 The H-Link semantic routing approach

The H-Link semantic routing mechanism is based on the idea of exploiting the network knowledge layer of a peer ontology and of using the H-Match matchmaking system for providing semantic query routing support. Formally, we can define H-Link in terms of its constituent elements as follows.

Definition of H-Link. The H-Link mechanism HL is defined as a 4-tuple of the form $\mathrm{HL} = \langle Q, O, \mathrm{MP}, N_{cr} \rangle$ where Q is a discovery query, O is a peer ontology, MP is a matching policy for the configuration of the semantic matchmaker (i.e. H-Match), and N_{cr} is the number of credits attached to the query Q which represents the number of replies that the requesting peer wishes to receive as answers to Q.

In order to illustrate the H-Link mechanism, we consider a discovery query Q with a target concept tc and we distinguish two different roles[1]:

- *Requesting peer*. A requesting peer p submits a discovery query Q to the network in order to identify relevant partners for subsequent resource sharing. Through H-Match, the matching concept list MCL is determined, that is $\mathrm{MCL} = \{\langle c_1, SA(tc, c_1)\rangle \ldots \langle c_n, SA(tc, c_n)\rangle\}$, where $c_1 \ldots c_n \in O$ are matching concepts for Q and $SA(tc, c_1) \ldots SA(tc, c_n)$ are the corresponding semantic affinity values. Peer p also sets the *number of credits* N_{cr} that will be distributed to the query Q recipients. Therefore, H-Link is invoked by passing the list MCL to select the semantic neighbors for query Q submission.
- *Receiving peer*. A receiving peer r receives a discovery query Q together with the number of credits nc_r assigned to r. The peer r needs to evaluate whether matching concepts can be provided back to peer p. To this end, H-Match is invoked by peer r and the list MCL of matching concepts is still produced as a result. If $\mathrm{MCL} \neq \varnothing$, the peer r sends MCL back to peer p by consuming one credit, otherwise no reply is sent back to peer p and all the received credits are still available for forwarding. If at least one credit is available, H-Link is invoked by peer r to select the semantic neighbors for query Q forwarding; otherwise the propagation mechanism stops. Each query is identified through a unique timestamp. The timestamp is exploited for enabling a peer to discard duplicate incoming queries. Query replies are returned to the requesting peer by following the reverse query path in order to avoid a sudden burst of incoming messages as suggested in Borch and Vognild (2004).

4.1 H-Link routing mechanism

H-Link is invoked for both query Q submission and forwarding. The number of credits available for distribution is indicated with A_{cr} where (1) $A_{cr} = N_{cr}$ for the requesting peer p when Q is submitted, (2) $A_{cr} = n_{cr}$ for each receiving peer r that has not consumed any credit for Q reply, and (3) $A_{cr} = n_{cr} - 1$ for each receiving peer r that has consumed one credit for Q reply.

[1] For the sake of clarity, we consider the case of a single target concept in the query. The H-LINK semantic routing mechanism can be easily extended to consider the case of more than one target concept. Furthermore, we assume that the peer ontology of the requesting peer already contains some neighbor peers in its network knowledge layer.

Three main steps define H-LINK: *selection of semantic neighbors*; *ranking of semantic neighbors*; *distribution of credits*.

1. **Selection of semantic neighbors.** The network knowledge layer of the peer ontology is accessed to select the neighbor peers (together with the associated confidence values) which are connected to the concepts in MCL through an affinity link. A list SNL of semantic neighbors is returned as a result. A semantic neighbor $sn \in \mathrm{SNL}$ is described in the form $sn = \langle n_{np}, \{c_1,\ cf_1 \ldots c_m,\ cf_m\}\rangle$, where n_{np} is the identifying label of the neighbor peer featuring sn, while $c_1 \ldots\ c_{m1} \in \mathrm{MCL}$ are the concepts of MCL connected to np through an affinity link, and $\{cf_1 \ldots cf\}$ are the corresponding confidence values.
2. **Ranking of semantic neighbors**. The semantic neighbors in SNL are ranked with respect to the query target tc by combining the confidence values associated to the semantic neighbors in SNL and the semantic affinity values in MCL. The harmonic mean is exploited to this end. Given a semantic neighbor $sn \in \mathrm{SNL}$, the ranking value r_{sn} corresponds to the following formula:

$$r_{sn} = \frac{\sum_{i=1}^{m} \frac{2 \cdot cf_i \cdot SA(tc,c_i)}{cf_i \cdot SA(tc,c_i)}}{m} \tag{2}$$

   Finally, a ranked list $\mathrm{RSNL} = \{\langle sn_1,\ r_{sn_1}\rangle \ldots \langle sn_k,\ r_{sn_k}\rangle\}$ of semantic neighbors with the corresponding ranking value is returned as a result. A threshold mechanism can be used to rule out the semantic neighbors with a ranking value lower than a predefined threshold t.
3. **Distribution of credits**. The semantic neighbors in RSNL determine the recipients of the query Q. Available credits A_{cr} are proportionally distributed to the semantic neighbors in RSNL according to their ranking value. Then, the number of credits nc_{sn} assigned to the semantic neighbor $sn \in \mathrm{RSNL}$ is computed as follows:

$$nc_{sn} = \left\lfloor \frac{A_{cr}}{\sum_{\forall sn_i \in \mathrm{RSNL}} r_{sn_i}} \cdot r_{sn} \right\rfloor \tag{3}$$

   Finally, a query distribution list $\mathrm{QDL} = \{\langle sn_1,\ nc_{sn_1}\rangle \ldots \langle sn_k,\ nc_{sn_k}\rangle\}$ is returned as a result.

Considerations. In H-LINK, the harmonic mean is used to combine the confidence values associated to the peers in SNL and the semantic affinity values in MCL. Such a choice is motivated by the idea to push the ranking value of the semantic neighbors that present similar values of confidence in SNL and semantic affinity in MCL. This way, when calculating the comprehensive ranking value of a semantic neighbor, we succeed in balancing the impact of the semantic affinity computed on the fly for the current query and the impact of the confidence values derived from past interactions.

A possible side effect of the H-LINK mechanism is due to the fact that credits are distributed on the basis of the knowledge discovered during past interactions. This means that the knowledge of new peers joining the system is hardly discovered and it is not considered for semantic neighbor selection. H-LINK deals with this by introducing a perturbation in the credit distribution phase. As proposed in Zeinalipour-Yazti *et al.* 2005), a small set of random peers is picked and it receives a percentage of the credits available for distribution. As a result, a larger part of the network is explored with the aim to discover additional knowledge and to include new peers in the semantic routing process. We will evaluate the impact of random peers in credit distribution with respect to the default H-LINK semantic routing protocol in Section 5.

Moreover, some special cases can occur in H-LINK for which appropriate solutions are discussed in the following.

Case 1: $\mathrm{MCL} = \emptyset$. When $\mathrm{MCL} = \emptyset$, that is the peer ontology does not contain concepts matching the target query, the default H-LINK invocation with selection and ranking of semantic neighbors can not be executed. In this case, credits are proportionally distributed according to the semantic neighborness of the peers in the network knowledge layer. This choice is motivated by the assumption that a peer with high semantic neighborness is also highly queried and maintains a populated network knowledge layer in its peer ontology. Then, peers with high semantic neighborness can be exploited as relays with the aim of increasing the chances to reach peers with matching knowledge. As another possible solution, we are working on a *back-propagation* strategy to address query forwarding when MCL $= \emptyset$. In the back-propagation strategy, when a peer r receives a discovery query Q and the credits nc_r from a requesting peer nc_r, the query propagation is stopped and credits nc_r are returned to peer p. This way, peer p can update its network knowledge layer by reducing the confidence measures of peer r for the query target concepts and the returned credits nc_r can be reassigned (1) by increasing the number of credits distributed to other query recipients and (2) by distributing credits to those semantic neighbors that were previously excluded due to low ranking values. The back-propagation strategy for credit distribution introduces additional complexity in the query forwarding process. We plan to evaluate the effectiveness of such a strategy in future simulation experiments. The back-propagation strategy is not viable when the query is submitted by the requesting peer. In case that the requesting peer obtains the MCL $= \emptyset$, only the query distribution strategy based on semantic neighborness is feasible.

Case 2: insufficient credits. The nc_r credits received with a query Q by a peer r can be not sufficient to assign at least one credit to each $sn \in$ RSNL. In this case, credits are assigned starting from the semantic neighbors with the higher ranking value. As a result, the semantic neighbors with low ranking value can be excluded by QDL, thus reducing the number of query Q recipients. In case that a query Q duplicate is received, the attached nc_r credits are used to accomplish the credit distribution phase by considering the previously excluded semantic neighbors. For this reason, each query is cached together with the associated QDL until a predefined deletion timeout is expired.

Case 3: empty network knowledge layer. Such a situation may occur during peer bootstrapping after joining the system and during normal peer activities when all the semantic neighbors are temporarily disconnected. In this scenario, credits are proportionally assigned to each directly connected peer when a query is submitted to the system.

4.2 Building the network knowledge

Receiving replies in response to a discovery query submitted with H-Link, a requesting peer has to populate the network knowledge layer of its peer ontology according to the received matching concepts and associated semantic affinity values. In particular, a neighbor peer is inserted in the peer ontology for each peer that replied to a discovery query.

We consider a peer p submitting a discovery query Q containing a single target concept tc. The peer r replies to the query Q by providing a list of matching concepts $c_1 \ldots c_n$ and associated semantic affinity values $\mathrm{SA}(tc, c_1) \ldots \mathrm{SA}(tc, c_n)$. Receiving such a query reply, a neighbor peer np is defined in the peer p ontology to denote that peer r is a semantic neighbor, and an affinity link is defined to connect np to the concept tc. For defining an affinity link, two different scenarios can occur:

- The target concept tc is in the content knowledge layer of the peer p ontology. In this case, an affinity link $al = (tc, np, cf)$ is defined, and the confidence value cf is computed as the average mean of the received semantic affinity values for tc, namely $cf = \frac{\sum_{i=1}^{n} SA(tc,c_i)}{n}$.
- The target concept tc is not in the content knowledge layer of the peer p ontology. In this case, the affinity link is not defined. As an alternative, the peer p can decide to evolve its peer ontology by including the definition of the concept tc through ontology evolution techniques (Castano *et al.*, 2006b). As a result, the affinity link is defined according to the previous method.

The same peer p actions are performed when more than one target concept tc is contained in the query Q. In this case, a different affinity link can be defined between the neighbor peer np and each target concept tc, according to the matching results returned in the query reply by peer r.

When an affinity link is inserted/updated, the semantic neighborness sns_{np} of the corresponding neighbor peer *np* is modified. Currently, sns_{np} is computed as the average mean of the confidence values associated to the affinity links connected to *np*.

Considerations on the computation of confidence values. In the network knowledge layer of a peer ontology, confidence and semantic neighborness allow to link peers with similar knowledge at different level of granularity (i.e. concept and peer level). Moreover, confidence values introduce a flexible and extensible mechanism that can be improved according to the specific peer requirements. For instance, a peer can decide to restrict the creation of new affinity links only for a selected set of concepts of interest. This way, only a limited number of concepts are connected to neighbor peers, thus fostering the efficiency of peers with very focused interests. As another possible option, the creation of affinity links can be restricted only to the received matching concepts whose semantic affinity value SA is over a given threshold. This way, only highly expert semantic neighbors are stored in the peer ontology.

In H-Link, confidence values are computed by relying on the semantic affinity evaluations produced by the matchmaker during the knowledge discovery phase. Such a choice is aimed at supporting query propagation by exploiting semantics-based criteria, like semantic affinity, rather than topology-based parameters, like peer distance and bandwidth. However, the computation of confidence measures can be extended to combine additional metrics and to provide a more accurate evaluation of the semantic neighborness of each peer. In this respect, possible extensions to the current techniques for confidence computation are being investigated and some preliminary ideas are discussed in Section 7.

4.3 Application example

As an example of H-Link semantic routing, we consider the peer B of Figure 1. Peer B intends to submit to the system the query Q2 described in Figure 3(a) with total number of credits to distribute $N_{cr} = 5$. The peer B uses H-Match to compare the query Q2 against its peer ontology (see Figure 3(b)). As a result, the following semantic affinity values are returned by H-Match:

SA(Book, Volume) = 0.79
SA(Book, Publication) = 0.49

By invoking h-link, we find that:

$$\begin{aligned}&\text{MCL} = \{\langle \text{Volume},\ 0.79\rangle\langle \text{Publication},\ 0.49\rangle\}\\ &\text{SNL} = \{\langle \text{peer A},\ \{\text{Volume},\ 0.74\}\rangle,\ \langle \text{peer E},\ \{\text{Publication},\ 0.81\}\rangle\}\\ &\langle \text{peer F},\ \{\text{Volume},\ 0.875,\ \text{Publication},\ 0.62\}\rangle\end{aligned}$$

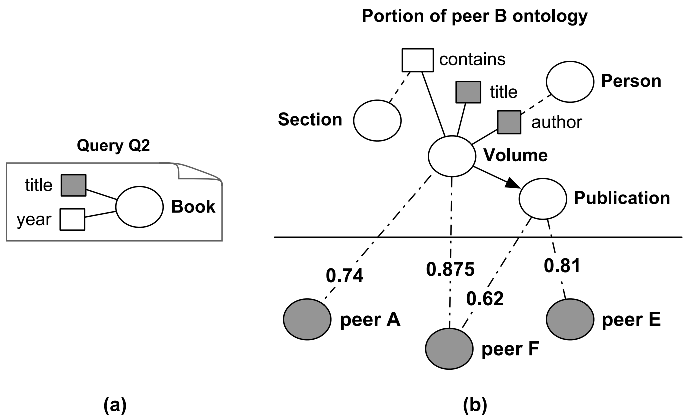

Figure 3 (a) The query Q2 example and (b) a portion of the peer B ontology

Table 1 Example of semantic neighbor ranking and credit distribution

Semantic neighbor	Ranking value	Assigned credits
Peer A	0.764	2
Peer E	0.611	1
Peer F	0.689	2

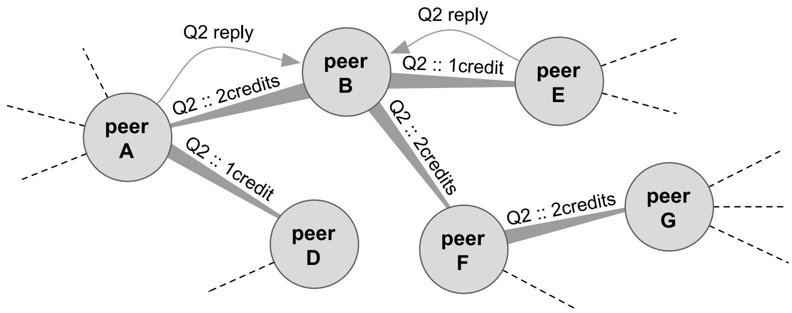

Figure 4 The H-Link routing schema for query Q2

On the basis of such results, H-Link computes the ranking of the semantic neighbors in SNL and assigns the corresponding number of credits, as shown in Table 1.

The query Q2 is then submitted to the selected semantic neighbors together with the assigned number of credits. As shown in the routing schema of Figure 4, peer A receives the query, consumes one credit for replying to peer B, and forwards the query Q2 to peer D by assigning the last remaining credit. Peer E consumes the unique credit received and stops the forwarding process, while the peer F forwards all the received credits to peer G as no reply is sent back to peer B.

As a final remark, we stress that H-Link is employed during the discovery phase for identifying the query recipients with the higher chances to provide matching replies. After the discovery phase, we assume that conventional *search queries* are submitted by the requesting peer B to point-to-point interact with one or more of the previously discovered peers for actual data acquisition. Further details regarding data sharing and semantic collaboration techniques in ontology-based P2P systems are provided in Castano *et al.* (2005).

5 Experimental results

The main goals of the experimentation is to assess the performance of the H-Link semantic routing mechanism. An experimentation by simulation has been performed to this end by relying on the Neurogrid P2P simulator.

5.1 Neurogrid configuration

In order to perform the H-Link experimentation, the Neurogrid P2P simulator has been configured according to the following global settings:

- #nodes. Initializing a simulation run, the network is generated by creating the number of nodes indicated by the #nodes parameter.
- #connections per node. Each node is connected with a subset of the other nodes. The initial number of node connections is randomly defined in the range indicated by the #connections per node parameter. The number of node connections can vary during the simulation according to the H-Link results.

- #concepts per node. Each peer is equipped with a randomly generated peer ontology. The size of the peer ontology is measured in terms of the number of concepts belonging to the ontology that is randomly defined in the range indicated by the #concepts per node parameter.
- max network layer size. Each peer maintains a vector for storing affinity links and corresponding neighbor peers. The size of such vector is defined according to the max network layer size parameter. Since each node has a limited space for storing neighbor peers, only the affinity links with the highest confidence values are maintained.
- #concepts per query. Each query is randomly defined and submitted by peers with equal probability. The number of concepts belonging to the query is randomly defined in the range indicated by the #concepts per query parameter.
- #queries. A simulation run finishes when the number of queries propagated in the system reaches the limit defined in the #queries parameter.
- #credits. Each query is associated to an initial number of credits defined in the #credits parameter.

For peer ontology and query generation, we have considered two reference OWL ontologies from the publishing domain (i.e. Ka[2], Portal[3]). An extraction procedure has been developed to pick out a random-size ontology region starting from a given concept of the reference ontologies. This way, we can provide both random generation and semantic homogeneity of peer ontologies. The same procedure is exploited also for query definition.

The H-MATCH matchmaking system is used for providing semantic affinity evaluation among queries and peer ontologies during the simulation process. For what concern the role of H-MATCH in the simulations, we have produced a file containing all the H-MATCH results and corresponding affinity values obtained from the complete comparison of the two reference ontologies. Such a file is exploited by the Neurogrid simulator in order to decide whether a peer is relevant for a given query. In other words, when a query is propagated in the network, the Neurogrid simulator checks in the file whether the concepts in the query have matching entries with the concepts in the ontology of each receiving peer. The choice of using pre-computed ontology mappings instead of their run-time computation with H-MATCH is motivated by the goal of better focusing on the network behavior of H-LINK rather then on its performance in terms of response times. Furthermore, the use of pre-computed mappings simplifies recall calculation that is one of the H-LINK evaluation metrics. However, a run-time computation of ontology mappings would only have a minimal impact on the routing process. In fact, the H-MATCH performance is mainly affected by the size of the ontologies to be compared and experiments show that H-MATCH scales quite well when domain ontologies composed of one hundred and more concepts are considered (Castano *et al.*, 2006c). We note that ontologies of such size are commonly suited for the goal of describing peer resources in a P2P network that is the typical application scenario of H-LINK.

5.2 *Simulation goals*

Simulations are devoted to measure *generated traffic* and *recall* of H-LINK. By generated traffic we mean the overall number of messages routed during a complete simulation run. Furthermore, we measure recall according to the classical definition of Information Retrieval (Salton, 1989), namely as the ratio of the number of matching concepts retrieved by a H-LINK query to the total number of matching concepts that was expected according to the file of matching results. In the simulation, the generated traffic will be analyzed through the Avg Traffic indicator which describes the average number of messages required by each query. Moreover, recall will be analyzed through the Min recall, Max recall, and Avg recall indicators, which describe the minimum, maximum, and average recall values obtained by a query, respectively.

[2] http://protege.stanford.edu/plugins/owl/owl-library/ka.owl
[3] http://www.aktors.org/ontology/portal

The experiments are devoted to illustrate the obtained results for what concern (1) H-LINK scalability, (2) H-LINK performance when compared with the Gnutella protocol, and (3) the impact of random credit distribution on the performance of the default H-LINK mechanism. The choice of a comparison with Gnutella is due to the fact that both H-LINK and Gnutella define an overlay network posed on top of an unstructured P2P infrastructure. Apart from the architectural similarities, we have chosen Gnutella since its routing protocol is well-known and it is frequently considered as a reference example. Some other P2P routing protocols have been also considered for a comparison with H-LINK and we plan to perform additional experiments in future work (see Section 7).

The following results are expected and will be verified and discussed after experiment presentation.

- The main contribution of H-LINK is expected in terms of scalability and traffic reduction.
- Recall is positively affected by the use of more accurate matching models (i.e. deep, intensive).
- Random credit distribution contributes to reach a larger part of the network and to discover additional peers.

As a final remark, we stress that simulation experiments have been performed by gradually varying the configuration parameters (i.e. #nodes, #connections per node, #queries, #credits). For the comparison of H-LINK with Gnutella, we have also repeated the simulation by relying on different H-MATCH models (i.e. surface, shallow, deep, intensive) with a standard matching configuration (i.e. $W_{LA} = 0.5$ and $t = 0.5$). When not differently specified, the H-MATCH intensive model is used for generating confidence values. In the following, we discuss some selected results where the H-LINK properties are more evident. However, a complete report on the simulation results is provided in Montanelli (2006).

5.3 H-Link scalability results

The H-LINK scalability is evaluated by measuring traffic and recall when the number of peers joining the system is a growing variable. The simulation has been performed according to the following configuration:

- #nodes varies in the range [100–1000] with an increment of 180 nodes for each simulation run.
- #connections per node is increased according to the #nodes parameter and varies in the range [3–10].
- #queries = 5000.
- A complete simulation has been performed for each value of #credits varying in the range [12–24] with an increment of 4 credits.

In Figure 5, the simulation results for #credits =24 are reported. We observe that very interesting results are obtained by H-LINK in terms of generated traffic. It is important to stress that such results are provided by considering the worst case scenario where the completely random

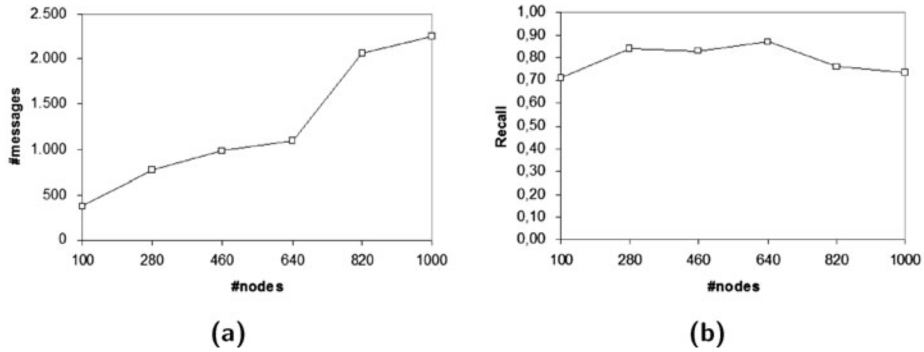

Figure 5 Evaluation of H-LINK scalability for #credits = 24: (a) Traffic (b) Recall

generation of both queries and peer ontologies is performed. In future experiments, we plan to exploit additional distribution models with the aim at investigating whether further improvements in H-Link scalability are still possible when more realistic distribution models are considered (e.g. the Zipf model (Korfhage, 1997)). On the other hand, we observe that recall lies in the range [0.71, 0.87] and thus it is not significantly affected by the variation in the #nodes parameter. A similar consistent behavior in both generated traffic and recall values can be observed in the other simulations where different values of the #credits parameter are employed (see Montanelli (2006) for further details).

5.4 *Comparison with Gnutella*

The comparison of H-Link with Gnutella has been evaluated by measuring traffic and recall when the query scope (i.e. number of credits for H-Link and TTL for Gnutella) is a growing variable. The simulation has been performed according to the following configuration:

- #nodes = 500
- A complete simulation has been performed for each value of #connections per node varying in the range [3–10].
- #queries = 5000.
- #credits varies in the range [27–54] with an increment of 9 credits for each simulation run. The #credits parameter is proportionally set according to the value in #connections per node. For the Gnutella simulation, the #credits parameter is replaced by the TTL parameter which varies in the range [3–6].

In Figure 6, the simulation results for #connections per node = 8–10 are reported. In Figure 6, we also compare the H-Link results obtained by varying the H-Match model from surface to intensive matching. We note that excellent results are obtained by H-Link in terms of generated traffic. Moreover, we also note that the variation in the adopted matching model does not significantly affect the performance of H-Link when the generated traffic is considered. For what concern the recall values, the optimal behavior of Gnutella is motivated by the fact that the simulation of Figure 6 works with #nodes = 500 and #connections per node = 8–10. Under such a configuration, Gnutella floods the network and succeeds in reaching a large part of either relevant or irrelevant nodes, thus retrieving all the available concepts matching the target query. On the other hand, H-Link presents very interesting results in terms of recall values, which lie in the range [0.82, 0.96] when the intensive matching model is used. Furthermore, we note that the choice of a specific matching model has an impact on the H-Link recall values. According to the results of Figure 6(b), the deep and the intensive matching models are suggested when a

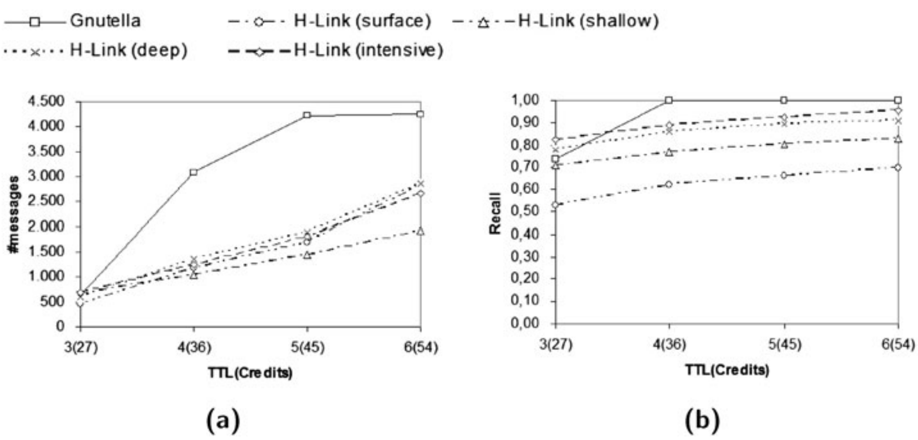

Figure 6 Comparison of H-Link with Gnutella for #connections per node 8–10: (a) Traffic (b) Recall

more accurate behavior of H-LINK is required. However, we stress that more accurate H-MATCH models also imply higher computation time as discussed in Section 3.2. For generic P2P applications, the shallow matching model is selected as a default in that it is sufficiently accurate while keeping the matching computation quite limited. For a comprehensive comparison of H-LINK and Gnutella, both generated traffic and recall need to be considered as a whole. This way, we can observe that there is a tradeoff between traffic reduction and accuracy of results. In this respect, H-LINK claims to succeed in considerably reducing traffic while preserving an important level of accuracy in results at the same time.

5.5 Impact of random credit distribution

H-LINK has been implemented in the Neurogrid P2P simulator as two different versions, namely H-Link-D and H-Link-R. The H-Link-D version implements the default H-LINK algorithm as presented in Section 4. On the opposite, the H-Link-R version supports the random distribution of a subset of the available credits. In other words, a portion of the available credits are randomly assigned during the H-LINK credit distribution phase in order to discover the knowledge of the new peers joining the system. The goal of this test is to measure the impact of random credit distribution on the default H-LINK algorithm by observing traffic and recall when the number of queries is a growing variable. The simulation has been performed according to the following configuration:

- #nodes = 1000
- #connections per node is randomly chosen in the range [8–10].
- #queries varies in the range [1000–10 000] with an increment of 1000 queries for each simulation run.
- #credits = 24.

The simulation results for the considered configuration are reported in Figure 7. We note that a slight increase in traffic can be observed when H-Link-R is applied. This is due to the fact that some of the available credits are wasted on irrelevant paths, thus confirming the benefits of the default H-Link-D mechanism for traffic reduction. Furthermore, we note that the default H-Link-D algorithm still provides better results in terms of recall on the average case. By reading the poor results of the H-Link-R algorithm, we stress that the role of the random credit distribution is to enlarge the query scope for handling the dynamic peer behavior in terms of join/leave operations. In the simulation, join and leave operations are not considered, thus the real effects of random credit distribution are only partially visible. However, we observe that the H-Link-R algorithm can be positively employed when specific peer behaviors occur. In particular, random credit distribution is suggested in highly dynamic P2P scenarios where a large number of join/leave operations are expected per time unit and frequent changes in peer ontologies cause the

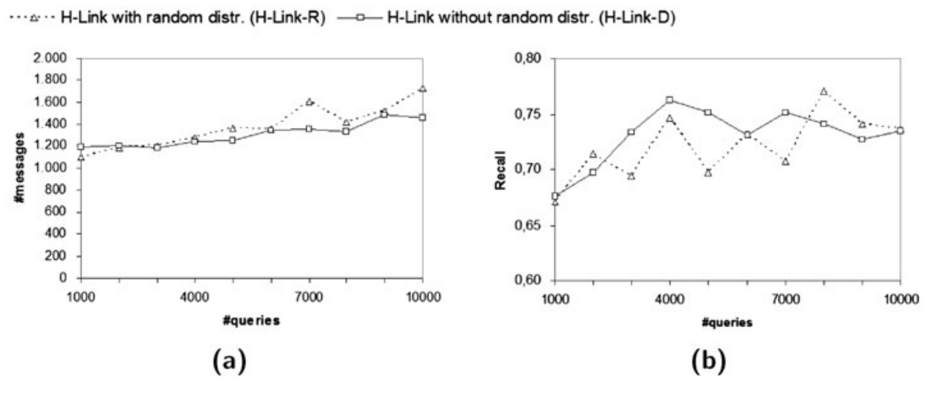

Figure 7 Impact of H-LINK random credit distribution: (a) Traffic (b) Recall

modest population of the network knowledge layer, thus negatively affecting the performance of the default H-Link-D algorithm.

5.6 *Final considerations*

According to the expected results presented in Section 5.2, we note that H-Link simulations confirm most of the hypothesis. In particular, the main contribution of H-Link regards traffic reduction. Such an interesting result also characterizes some other P2P semantic routing approaches (e.g. Joseph, 2002; Staab *et al*., 2004), confirming that query propagation on a semantic basis is an effective solution when compared with traditional techniques, like Gnutella. In the case of H-Link, the benefits in terms of traffic reduction are also supported by interesting values in terms of recall and scalability. In our opinion, two main reasons lead to such a result. On one side, the use of ontology matching techniques has a high impact on recall measures where H-Link is near to 52% in the worst case and reaches 100% in the best case. On the other side, the use of credits represents a first attempt to enable peers, and thus users, to supervise the query propagation process through an application-oriented parameter rather than network-oriented ones, like TTL.

For what concern the H-Match models, the simulations fully confirm the expected results. In particular, more accurate matching models (i.e. deep and intensive) provide higher recall values. However, fairly good results in terms of recall are obtained also with less accurate matching models (i.e. surface and shallow). As discussed in Section 5.4, the choice of the correct H-Match matching model is a key aspect when peer computational resources are a critical factor. In this respect, it is important to note that a peer can switch from one H-Match model to another on the fly according to the current peer status in terms of available bandwidth and workload.

Finally, we stress that also the expected impact of the random credit distribution is indirectly confirmed by the simulations. Furthermore, we observe the H-Link-R version can be adopted to improve the effectiveness of the learning phase for discovering the peer semantic neighbors. As discussed in Section 5.5, H-Link-R is suited to work in highly dynamic P2P scenarios for handling frequent join/leave operations and peer ontology modifications.

6 Related work

The main limits that affect traditional P2P routing protocols derive from two opposite goals that need to be pursued. On one side, queries need to reach the largest number of peers in order to increase the chances to locate the target file. On the other side, the need of scalability requires that the generated number of messages should be as lower as possible. When the P2P network is targeted to knowledge sharing, query routing becomes even more complex than traditional file-sharing systems due to the different levels of semantic complexity that need to be considered in query processing (Ehrig & Sure, 2004). In this respect, content-based and semantic routing techniques are being proposed to improve the tradeoff between the need to maximize the effectiveness of the discovery phase while minimizing the network traffic of a P2P system. In the following, we analyze and we compare with H-Link the main existing approaches for P2P semantic routing. In particular, we have considered the approaches where the idea of peer context is supported in some way to describe the knowledge provided by a single and independent node. In this respect, the considered approaches are classified in two main categories according to the adopted peer-knowledge representation approach, namely *metadata-based* and *ontology-based* semantic routing techniques. For a complete classification and critical review of the existing P2P semantic routing approaches, the reader can refer to Montanelli (2006).

Metadata-based semantic routing techniques. Metadata-based sematic routing approaches support an implicit definition of peer context since each node participates to the system by providing a set of files to share together with a set of corresponding metadata (e.g. title, author) and keywords. In this respect, the context of a peer p is defined by the set of keywords associated to the shared files provided by p. In such kind of P2P systems, syntactic matching techniques (e.g. string-based and keyword-based) are defined to evaluate the similarity of a peer file with a given target request. Existing metadata-based semantic routing approaches can be distinguished according to the level of adaptivity characterizing the query forwarding mechanism. As an interesting example of metadata-based sematic routing approach, the Intelligent Search Mechanism (ISM) is presented in Zeinalipour-Yazti. *et al.* (2005) as a novel mechanism for information retrieval in P2P networks. For each query, a peer exploits the replies of past queries and estimates which of the known peers are more likely to return relevant results for the current query, and propagates the request to those peers only. Searches occur when a peer submits to the system a query containing one or more target keywords of interest. Each receiving node matches the target keywords in the query with the keywords associated to its own documents and replies by returning those documents that are associated to at least one matching keyword. In order to match queries and document keywords, traditional similarity metrics can be used, such as the *cosine* similarity function (Baeza-Yates & Ribeiro-Neto, 1999) that is proposed by the authors. Each peer maintains a profiling structure where it stores the answer observations related to the queries that were routed through it. Profiling structures represent an element of locality to be exploited for selection of query recipients. A forwarding mechanism based on TTL is defined to enlarge the query scope when required. Similarly to ISM, the concept of 'good' peer is defined in Ramanathan *et al.* (2002) as the key element of an automatic interest-based mechanism where peers with similar interests are close and the similarity between peer interests decreases as the distance among the peers increases. To measure the interest similarity, each peer calculates the 'percQueryHits' value which measures the ratio of the hits provided by each of its immediate neighbors (i.e. directly connected peers) to the total number of collected query replies. The percQueryHits values are then exploited by a requesting peer to select the most responsive neighbors as query recipients. Syntactic matching techniques, such as string-based techniques, are then used to determine the query reply by comparing the target keywords of a query with the document metadata in the repository of each receiving peer. As another example, the Seers search protocol is defined in Borch and Vognild (2004). In the Seers infrastructure, a XML meta-document is used to describe each shared resource and a matching policy is provided to define how tags in different meta-documents should be matched against others. Furthermore, a transmission policy is used to describe how a query with a target meta-document should be propagated in the network. In particular, each peer associates a ranking value to its neighbors. For each neighbor, the ranking value is computed on the basis of (1) the neighbor state (i.e. active, inactive, stale), and (2) the neighbor interests that are discovered by observing the meta-documents routed by the neighbor. The Seers search protocol has been refined in Borch (2005) where the Socialized.Net infrastructure is presented to support a mechanism for query forwarding based on preferences and reputation for peer ranking. Finally, the Neurogrid system is proposed in Joseph (2002) as an adaptive and decentralized search system. In such an approach, semantic routing is intended as content-based query forwarding, and a learning mechanism is defined to dynamically adjust the relevance of known peers for each query. In particular, each peer maintains (1) a list of keywords associated to its shared files and (2) a frequency indicator associating a keyword with the number of hits-per-remote-peer collected in previous interactions. The frequency indicator is used to rank the remote peers for query recipient selection.

Ontology-based semantic routing techniques. Ontology-based sematic routing approaches support an explicit definition of peer context since each node participates to the system by providing an ontology describing its shared resources. In this respect, the ontology of a peer p defines the context of p and semantic-based matching techniques are defined to evaluate the similarity of a peer context with respect to a given target request. Existing ontology-based semantic routing approaches can be distinguished according to the supported ontology specification language and to the level of semantic complexity exploited by the adopted matching techniques. As an interesting example of ontology-based sematic routing approach, the REMINDIN' (Routing Enabled by Memorizing INformation about Distributed INformation) algorithm is proposed in

Staab *et al.* (2004). Such an approach is defined in the framework of the SWAP platform where (1) each peer provides a local node repository (i.e. an ontology) containing the RDF(S) statements that describe either data or conceptual information, and (2) the SeRQL language is adopted for query formulation (Broekstra *et al.*, 2003). Furthermore, the SWAP storage model allows a peer to measure the confidence value of another remote peer p in terms of the number of relevant answers provided by p for each argument r. Such a confidence value is updated when new interactions are issued with the peer p on topics related to r. Receiving a query, a peer replies to a requesting node with the RDF(S) statements that satisfy the SeRQL query and that are contained in its local node repository. After the reply, a peer can forward the query to other peers in the network according to a TTL-based mechanism. In this respect, REMINDIN' is invoked to select the most promising query recipients by ranking remote peers according to their stored confidence values. The REMINDIN' approach has been refined in Loser *et al.* (2005) where advanced techniques for peer confidence computation are illustrated. As another example, P2P Semantic Link Networks (P2PSLN) are presented in Zhuge *et al.* (2004). In a P2PSLN, different types of semantic links can be established between two peers (i.e. equal to, similar to, reference, implication, subtype, sequential, empty, and null) according to a set of predefined reasoning rules. Each peer defines its own XML Schema (source schema) describing the contents to share and adopts SOAP-based messages to communicate with the other members of the network. When a peer joins a P2PSLN, it acquires the XML Schema of a subset of network nodes (target schema) and analyzes them recursively through cycle analysis and functional dependency analysis with the intention to identify the semantic mappings between the source and the target elements. Upon receiving a query, a peer will autonomously forward the request to relevant peers according to its semantic links as well as the similarity between elements and structures of peer schemas. Similar to P2PSLN, the 'routing by mapping' mechanism is proposed in Mandreoli *et al.* (2006) to address semantic query propagation in a Peer Data Management System (PDMS) (Halevy *et al.*, (2004). To this end, each peer provides an OWL ontology describing the informative contents of its sources and two adjacent peers (i.e. neighbors) are logically connected through a set of semantic mappings that keeps track of the semantic affinity among the ontology concepts of the two peers. In order to discover the semantic mappings between two peers, schema matching operations based on terminological and structural comparisons are performed on the respective peer ontologies. Each resulting mapping is finally characterized by a numerical score in the range [0,1] computed through a fuzzy-based similarity function. To support query submission/forwarding, a Semantic Routing Index (SRI) is maintained by each peer as a matrix where columns are the concepts of the local ontology and rows contain the neighbor peers as well as a summarized indication of their capability to provide relevant results for each local concept. In Haase *et al.* (2007), the authors define a peer selection approach based on a shared ontology and an expertise advertisement mechanism. In such an approach, each peer picks out the recipients of a given query by selecting the known peers whose expertise is similar to the query subject. The system is characterized by a common ontology that provides a shared conceptualization for the domain of interest of all the system nodes. Each peer defines and subsequently advertises a semantic description of its expertise that represents the knowledge base of the peer according to the common ontology. As a result of advertisement, each peer acquires a list of nodes with similar expertise that is exploited to rank query recipients according to the semantic similarity between peer expertise and the query subject.

Original contribution of H-Link. Metadata-based semantic routing techniques have been mostly developed as P2P applications of Information Retrieval techniques. In this context, the focus is on improving the efficiency of the data retrieval procedures when applied to a P2P environment. To this end, adaptive query routing approaches are implemented in metadata-based semantic routing systems. This means that query recipients are dynamically selected according to the results of past interactions or by relying on the expertise estimations associated to the remote peers and calculated through some heuristic metric (e.g. number of query hits, peer reputation). One of the main challenges regarding the existing metadata-based semantic routing systems is related to the use of more expressive peer context representations and semantic-based matching techniques for evaluating the peer relevance with respect to a given query. On the opposite, ontology-based semantic routing techniques are proposed as P2P applications of the recent Semantic Web technologies, such as ontologies and related ontology matching techniques. In this respect, the focus is on providing expressive peer context representations and semantic-based matching techniques. To this end, structural-, linguistic-, and reasoning-based matching techniques are employed in the existing ontology-based semantic routing systems. In this context, one of the main challenges is related to efficiency issues. In particular, there is the need to improve the existing ontology matching techniques for adapting them to work in a P2P environment where performance and response time are critical aspects in order to reduce the risk of peer overloading. In this respect, H-Link has been conceived to merge the features of metadata- and ontology-based techniques by combining ontologies and ontology matching techniques for providing an adaptive P2P routing mechanism. In particular, the main contribution of H-Link is related to the use of independent peer ontologies and to the use of ontology matching techniques to build a network knowledge layer reflecting the gradual learning of semantic neighbors. A further contribution of H-Link regards the use of an ontology matchmaking system for allowing each peer to measure the confidence level of the semantic neighbors in terms of the level of semantic affinity of the matching concepts that a semantic neighbor replies with respect to a given concept of interest.

7 Concluding remarks and future work

In this article, the H-Link semantic routing mechanism has been presented. H-Link has been conceived to work in a peer-based system where each node provides its own context described through a peer ontology. H-Link relies on the results of the knowledge discovery interactions among peers to train the query routing behavior and to identify the peers with similar knowledge that are called semantic neighbors. Query recipients in H-Link are then selected according to confidence and semantic neighborness values associated with the discovered semantic neighbors. This way, H-Link aims at supporting query propagation by exploiting semantics-based criteria, like semantic affinity, rather than topology-based parameters, like peer distance and bandwidth. The results obtained in the experiments show that the H-Link mechanism succeeds in improving the effectiveness of traditional P2P query protocols by providing interesting results in terms of scalability at the same time.

Future work on the H-Link approach will regard three main research directions: *H-Link tuning, H-Link optimization*, and *H-Link application to P2P semantic community management*.

H-Link tuning. Further experiments are required to investigate the H-LINK tuning when different P2P application scenarios are considered. In particular, we plan to study how to dynamically setup the routing mechanism according to the variable network conditions in terms of peer stability. Moreover, we are also interested in analyzing the H-Link behavior when a different level of ontology overlapping among peers is considered.

As another further experiment, we are working on evaluating the H-Link behavior when a different model for peer-knowledge distribution is exploited. In particular, we intend to assess the variation of H-Link performance in terms of scalability when the ZIPF distribution is adopted. As discussed in Schlosser *et al.* (2003), this choice is due to the fact that the ZIPF distribution provides a non-uniform distribution and a more realistic model of peer-knowledge assignment where some concepts are more queried than others and certain concepts have a higher likelihood to be assigned to a peer ontology during the distribution phase.

At the same time, we plan to compare H-Link with other similar P2P semantic routing mechanisms. By considering the approaches presented in Section 6, we note that H-Link is characterized by a number of similarities with the REMINDIN' approach (Staab *et al.*, 2004). In particular, both the systems rely on ontologies and on adaptive techniques for selecting query recipients. In our opinion, the main difference between H-LINK and REMINDIN' consists of a different method for computing peer confidence. Thus, apart from the performance evaluation based on traffic and recall, the experimental comparison of the two approaches is motivated by the possibility to assess the benefits in adopting a semantic matchmaker like H-MATCH in computing peer confidence values.

H-LINK optimization. We are working on some algorithm extensions devoted to improve the current methods for confidence computation and for peer ontology management. In particular, trust-based techniques can be included in H-LINK to this end. Many different approaches have been proposed in the literature to handle trust issues in P2P systems where the inherent open nature of communications may cause malicious peer behaviors (AbdulRahman & Hailes, 1999; Xiong & Liu, 2004). The main existing approaches are based on the idea to advertise reputation information in order to penalize peers with malicious behavior. Information regarding semantic neighbor reputation can be obtained by exploiting an existing trust-based approach and by integrating it in the computation of confidence values. As another possible H-LINK extension, an aging mechanism can be introduced to periodically update confidence measures. As proposed in Joseph (2002), the confidence value *cf* associated to a given affinity link between *c* and *np* can be modified by observing the ratio between the number of relevant replies provided by *np* and the number of queries sent to *np* with a target concept related to *c*. When the ratio has low values, *cf* can be decreased to denote that the original confidence (i.e. semantic affinity) is no more actual. This way, only confirmed affinity links are maintained in the peer ontology, while unreliable confidence values are gradually reduced and finally dropped.

Finally, we are investigating the adoption of network-sniffing techniques for populating the network knowledge of a peer. In the peer bootstrapping, the H-LINK mechanism needs of training phase for locating its semantic neighbors. Network sniffing can be used to speed up H-LINK training by observing other peer queries and related replies for discovering the semantic neighbors.

H-LINK application to P2P semantic community management. In a peer ontology, the network knowledge layer stores the neighbor peers (i.e. semantic neighbors) that are exploited for enforcing the H-LINK semantic routing mechanism. We note that given a concept *c* in the content knowledge layer of a peer ontology, the set of neighbor peers connected to *c* through an affinity link represents a set of semantic neighbors capable of providing knowledge similar to the concept *c*. Such semantic neighbors can be considered as potential participants to a semantic community where the topic of interest regards the concept *c*. In this respect, the network knowledge can be exploited by H-LINK for selecting the semantic neighbors to invite and for aggregating emergent communities of peers on the basis of their common perspective and context on a given concept *c*. Some initial results on this topic are presented in Castano and Montanelli (2005).

Acknowledgements

This article has been partially funded by the ESTEEM (Emergent Semantics and cooperaTion in multi-knowledgE EnvironMents) PRIN project funded by the Italian Ministry of Education, University, and Research and by the EU FP6 BOEMIE (Bootstrapping Ontology Evolution with Multimedia Information Extraction) project.

References

AbdulRahman, A. and Hailes, S. 1999 Relying on trust to find reliable information. In *Proceedings of 1999 International Symposium on Database, Web and Cooperative Systems (DWACOS'99)*, Baden-Baden, Germany.

Androutsellis-Theotokis, S. and Spinellis, D. 2004 A survey of peer-to-peer content distribution technologies. *ACM Computing Surveys* **36**(4), 335–371.

Baeza-Yates, R. and Ribeiro-Neto, B. 1999 *Modern Information Retrieval*. New York: ACM Press Series/Addison Wesley.

Bonifacio, M., Bouquet, P., Mameli, G., and Nori, M. 2003 Peer—mediated distributed knowledge management. In *Proceedings of International Symposium on Agent Mediated Knowledge Management (AMKM 2003)*, Stanford, CA, USA.

Borch, N. 2005 Improving semantic routing efficiency. In *Proceedings of the 2nd International Workshop on Hot Topics in Peer-to-Peer Systems (Hot P2P)*, La Jolla, CA, USA.

Borch, N. and Vognild, N. 2004 Searching variably connected networks. In *Proceedings of the International Conference on Pervasive Computing and Communications (PCC-04)*, Las Vegas, Nevada, USA.

Broekstra, J. *et al.* 2003 A metadata model for semantics-based peer-to-peer systems. In *Proceedings of the 1st WWW International Workshop on Semantics in Peer-to-Peer and Grid Computing (SemPGRID 2003)*, Budapest, Hungary.

Castano, S., Ferrara, A., and Messa, G. 2006a ISLab H-Match results for OAEI 2006. In *Proceedings of International Workshop on Ontology Matching, Co-located with the 5th International Semantic Web Conference (ISWC 2006)*, Athens, Georgia, USA.

Castano, S., Ferrara, A., and Montanelli, S. 2005 Ontology-based interoperability services for semantic collaboration in open networked systems. In *Proceedings of the 1st International Conference on Interoperability of Enterprise Software and Applications (INTEROP-ESA 2005)*, Geneva, Switzerland.

Castano, S., Ferrara, A., and Montanelli, S. 2006b Evolving open and independent ontologies. *Journal of Metadata, Semantics and Ontologies* **1**(4), 235–249.

Castano, S., Ferrara, A., and Montanelli, S. 2006c Matching ontologies in open networked systems: Techniques and applications. *Journal on Data Semantics (JoDS)* **V**, 25–63.

Castano, S., Ferrara, A., and Montanelli, S. 2006d *Web Semantics and Ontology*, chapter Dynamic Knowledge Discovery in Open, Distributed and Multi-Ontology Systems: Techniques and Applications. Idea Group.

Castano, S. and Montanelli, S. 2005 Semantic self-formation of communities of peers. In *Proceedings of the ESWC Workshop on Ontologies in Peer-to-Peer Communities*, Heraklion, Greece.

Ehrig, M. and Staab, S. 2004 QOM—Quick ontology mapping. In *Proceedings of the 3rd International Semantic Web Conference (ISWC 2004)*, Hiroshima, Japan.

Ehrig, M. and Sure, Y. 2004 Ontology mapping—an integrated approach. *In Proceedings of the 1st European Semantic Web Symposium*, Heraklion, Greece.

Euzenat, J., Loup, D., Touzani, M., and Valtchev, P. 2004 Ontology alignment with OLA. In *Proceedings of the 3rd ISWC Workshop on Evaluation of Ontology-based Tools (EON 2004)*, Hiroshima, Japan.

Giunchiglia, F. and Shvaiko, P. 2003 Semantic matching. *Knowledge Engineering Review* **18**(3), 265–280.

Haase, P., Siebes, R., and van Harmelen, F. 2006 Semantic web and peer-to-peer. *In Expertise-Based Peer Selection*. Berlin, Heidelberg: Springer. pp. 125–142.

Halevy, A., Ives, Z., Madhavan, J., Mork, P., Suciu, D., and Tatarinov, I. 2004 The piazza peer data management system. *IEEE Transactions on Knowledge and Data Engineering* **16**(7), 787–798.

Joseph, S. 2002 NeuroGrid: Semantically routing queries in peer-to-peer networks. *In Proceedings of the International Workshop on Peer-to-Peer Computing (Co-located with Networking 2002)*, Pisa, Italy.

Korfhage, R. 1997 *Information Storage and Retrieval*. New York: John Wiley.

Loser, A., Staab, S., and Tempich, C. 2005 Semantic methods for P2P query routing. In *Proceedings of the 3rd German Conference on Multiagent System Technologies (MATES 2005)*, Koblenz, Germany.

Mandreoli, F., Martoglia, R., Penzo, W., and Sassatelli, S.2006 SRI: Exploiting semantic information for effective query routing in a PDMS. In *Proceediings of the 8th ACM International Workshop on Web Information and Data Management (WIDM 2006)*, Arlington, Virginia, USA.

Montanelli, S. 2006 *Emergent Communities for Semantic Collaboration in Multi-Knowledge Environments: Methods and Techniques*. PhD thesis, Università degli Studi di Milano, Italy.

Nejdl, W., Wolf, B., Qu, C., Decker, S., Sintek, M., Naeve, A., Nilsson, M., Palmér, M., and Risch, T. 2002 EDUTELLA: a P2P networking infrastructure based on RDF. In *Proceedings of the 11th International World Wide Web Conference (WWW 2002)*, Honolulu, Hawaii, USA.

Noy, N. F. and Musen, M. A. 2003 The PROMPT Suite: Interactive tools for ontology merging and mapping. *International Journal of Human-Computer Studies* **59**(6), 983–1024.

O'Brien, P. and Abidi, S. 2006 Contextual knowledge sharing in a P2P network. In *Proceedings of the IEEE International Conference on Engineering of Intelligent Systems (ICEIS 2006)*, Islamabad, Pakistan.

Ramanathan, M., Kalogeraki, V., and Pruyne, J. 2002 Finding good peers in peer-to-peer networks. In *IEEE International Parallel and Distributed Processing Symposium*, Fort Lauderdale, FL, USA.

Salton, G. 1989 *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer*. Reading, MA: Addison-Wesley.

Schlosser, M., Condie, T., and Kamvar, S. 2003 Simulating a P2P File-Sharing Network. In *Proceedings of the 1st WWW International Workshop on Semantics in Peer-to-Peer and Grid Computing (SemPGRID 2003)*, Budapest, Hungary.

Shvaiko, P. 2006 Ontology Matching Projects. `http://www.ontologymatching.org/projects.html`.

Smith, M. K., Welty, C., and McGuinness, D. L.(eds.) 2004 OWL Web Ontology Language Guide. World Wide Web Consortium (W3C), `http://www.w3.org/TR/owl-guide/`.

Staab, S., Tempich, C., and Wranik, A. 2004 REMINDIN': Semantic query routing in peer-to-peer networks based on social metaphors. In *Proceedings of the 13th International conference on World Wide Web (WWW 2004)*, New York, NY, USA.

Tsoumakos, D. and Roussopoulos, N. 2003 Adaptive Probabilistic Search (APS) for peer-to-peer networks. In *Proceedings of the 3rd International IEEE Conference on P2P Computing*, Linköping, Sweden.

Xiong, L. and Liu, L. 2004 PeerTrust: Supporting reputation-based trust for peer-to-peer electronic communities. *IEEE Transactions on Knowledge and Data Engineering* **16**(7), 843–857.

Zeinalipour-Yazti, D., Kalogeraki, V., and Gunopulos, D. 2005 Exploiting locality for scalable information retrieval in peer-to-peer networks. *Information Systems* **30**(4), 277–298.

Zhuge, H., Liu, J., Feng, L., and He, C. 2004 Semantic-based query routing and heterogeneous data integration in peer-to-peer semantic link networks. In *Proceedings of the International Conference on Semantics of a Networked World (ICSNW), in Cooperation with ACM SIGMOD 2004*, pp. 91–107, Paris, France.