A space reduction mechanism for the dynamic selection of ontological alignments

Paul Doran, Valentina Tamma, Ignazio Palmisano, and Terry R. Payne

Abstract—Effective communication in open environments relies on the ability of *agents* to reach a mutual understanding of the exchanged message by reconciling the vocabulary (*ontology*) used. Various approaches have considered how mutually acceptable mappings between corresponding concepts in the agents' ontologies may be determined dynamically through argumentation-based negotiation (such as *Meaning-based Argumentation*). However, the complexity of this process is high, reaching $\Pi_2^{(p)}$-complete in some cases. As reducing this complexity is non-trivial, we propose the use of ontology modularization as a means of reducing the space over which possible correspondences are negotiated. The suitability of different modularization approaches as filtering mechanisms for reducing the argumentation search space is investigated. We empirically demonstrate that some modularization approaches not only reduce the number of alignments required to reach consensus, but can also predict those cases where a service provider is unable to fully satisfy a request, without the need for argumentation.

Index Terms—Check keywords

1 Introduction

Effective communication within open and dynamic environments is dependent on the ability of *agents* (i.e. components that *provide*, or *consume* services) to reach a mutual understanding over a set of messages, where no prior assumptions can be made on the vocabulary used to communicate. Unlike small, closed environments (where all the components are known at design time), open, Web-scale environments are typically characterised by large numbers of services which are continually evolving or appearing, and where syntactic and semantic heterogeneity is the norm. Thus, few assumptions can be made about the services on offer at any time, the way in which they are modeled, or the terminology or vocabulary that they use. In such cases, it becomes imperative to specify the explicit vocabularies or ontologies used to facilitate meaningful communication as environments open up, or the heterogeneity of large systems increases. This has been facilitated by the emergence of standards for representing ontologies and optimised reasoners capable of processing them within a tractable timeframe [3].

In addition, transactions should be interpreted by both service providers and consumers based on the underlying semantics of the messages themselves, and thus these agents should resolve any type of mismatch that may exist due to the use of different, but conceptually overlapping ontologies. However, this reconciliation has to be achieved automatically and at run-time (without human intervention) if such components are to transact as the size of the environment grows.

Early systems avoided the problem of ontological heterogeneity by relying on the existence of a shared ontology, or simply assuming that a canonical alignment (i.e., a set of ontology correspondences, als known as mappings), possibly defined at design time, could be used to resolve ontological mismatches. However, such assumptions work only when the environment is (semi-) closed and carefully managed, and no longer hold in open environments where a plethora of ontologies exist. The emergence of different alignment-generation tools [9] has resulted in the existence of multiple, but differing alignments between ontologies, whose suitability can vary depending on the agent's tasks, goals and preferences. Whilst these techniques can be used to reconcile heterogeneous ontologies, they generally operate offline, typically requiring some level of human intervention, and thus are unsuitable for generating correspondences dynamically. However, these alignments can be generated offline and stored for later use within publicly available repositories, such as the *Ontology Alignment Server* (OAS) [10].

Recent approaches have been proposed that rely on negotiation to dynamically resolve ontological mismatches within open environments, by identifying mutually acceptable alignments or shared concepts between different ontologies [6], [12], [17]. However, the use of negotiation to collaboratively search a space of candidate correspondences becomes prohibitively costly as the size of the ontologies grows, and thus a reduction of this search space is highly desirable.

In this paper we explore the use of *Ontology Modularization* as a filtering mechanism for reducing the size of the ontologies used, and hence the size of the search space. Ontology modularization techniques typically produce a subset of ontological definitions (with respect to a supplied signature), known as an *ontology module*.

We examine a small number of techniques proposed that differ on the conditions used to determine the subset of definitions from the original ontology, and use these as a filtering mechanism for alignment negotiation. A framework is presented that integrates the use of modularization with an existing alignment negotiation approach, and the reduction in negotiated alignments is studied. The results demonstrate that

• *P. Doran, V. Tamma, I. Palmisano and T.R. Payne are with the Department of Computer Science, University of Liverpool, Liverpool, L69 3BX UK. E-mail: {pdoran, V.Tamma, I.Palmisano, T.R.Payne}@liverpool.ac.uk*

the number of negotiated alignments can be reduced by an average of 55.14% UPDATE THIS NUMBER (i.e. through eliminating unnecessary alignments for a given transaction). Whilst this reduction is highly dependent on the modularization technique used, the results also demonstrate that some techniques can eliminate the need for negotiation by rapidly identifying cases when no suitable alignments are available.

The paper is organized as follows: ...

2 COMMUNICATION IN OPEN ENVIRONMENTS

As stated in Sect. 1, communication in open environments relies on the agents' ability to reach a mutual understanding. The openness of the environment implies that agents can potentially ask any other agents for information or services, and must be able to do so irrespective of the types of ontological mismatches that can affect the ontologies used to communicate. Assuming that there is always a canonical set of ontology mappings (possibly defined at design time) is an assumption that is becoming increasingly untenable with the evolution of open environments. A number of approaches have tried to overcome this problem in different ways [6], [12], [17].

van Diggelen *et al* [17] propose the dynamic generation of a minimal shared ontology; minimality is evaluated against the ability of the agents to communicate with no information loss. However, this approach uses a limited ontology model whose expressivity allows only simple taxonomic structures, with no properties and only a few restrictions, such as disjointness and partial overlap. The expressive power of this model is non-standard, in that it does not correspond to any of the OWL flavours, and a reformulation of the model in Description Logics (the logical theory underpinning the standard ontology language OWL [15]) is mentioned, but no formal proof appears to be given of the soundness of this reformulation. Therefore, its applicability to real application ontologies expressed in OWL and published on the Web seems limited.

The increased availability of mechanisms for ontology mapping and alignment [9] implies that an agent can potentially deal with several ontology correspondences for any pair of concepts in two ontologies, therefore the problem of enabling dynamic communication between heterogeneous ontologies in open environments can be recast as a collaborative search through the space of all ontology mappings between two ontologies.

Mechanisms for the storing and provisioning of these mappings have been devised, for example Euzenat & Valtchev [10] introduce the notion of an *Ontology Alignment Server* (OAS), that is a dedicated agent able to generate all the potential mappings between two agent ontologies. The availability of these mappings, thus, allows agents to determine, in principle, the correspondences that are most suitable to achieve their goals.

Search mechanisms can be as simple as a brute force approach that selects only those mappings whose level of confidence is above a certain threshold specified for each agent, or can imply some form of negotiation between agents in order to determine those mappings that are mutually acceptable by both agents [6], [12].

Approaches to solve this search problem exploits the use of *argumentation* as a negotiation mechanism to locate mappings that are mutually acceptable by both agents have been described in literature. Laera *et al.* [12] use argumentation as a rational means for agents to select ontology mappings based on the notion of partial-order preferences over the different types of correspondences (e.g. structural vs terminological), using the *Value Based Argumentation framework (VAF)* [2], which prescribes different strengths to arguments on the basis of the values they promote and the ranking given to these values by the agents. Their approach assumed the use of OWL [15] as a common ontology language. Dos Santos *et al.* [6] proposed a variant on this idea, by representing ontology mappings as disjunctive queries in Description Logics.

The complexity of the search through the space of possible alignments can, however, become prohibitive when complex negotiation mechanisms like argumentation are involved, and reach $\Pi_2^{(p)}$-complete [8] [1].

This can make the search costly, especially when it is used to establish a common communication vocabulary (thus constituting the initial phase of any communication or transaction). Thus, a reduction of the search space *before* the argumentation process takes place, thus reducing the time neeeded to complete the process, seems to be an essential step to make the use of argumentation based techniques viable in open environments.

In Sect. 3 we present the *Meaning-based argumentation* [12] approach for negotiating ontology alignments, and then we propose our approach to reduce the space of all candidate mappings by identifying the subset of an agent's ontology that is relevant to the task the agent wishes to perform. Agents can then argue over this subset of the original alignment, thus exploring a search space which is smaller than the original search space, therefore skipping correspondences that are likely to be irrelevant for the current task.

3 ARGUMENTATION-BASED NEGOTIATION OF ONTOLOGY ALIGNMENTS

Sect. 2 discussed the two possible approaches for agent communication in dynamic environments. We argue that a viable way to establish ontology alignments on the fly is through the use of negotiation on the candidate ontology mappings that can be generated between two agent ontologies.

This paper follows the *Meaning-based argumentation* approach proposed in [12] that allows agents with different terminologies to dynamically and automatically reach consensus on the terminology they use to interact. In this approach a *correspodence* (or mapping) is described as a tuple: $m = \langle e, e', n, R \rangle$, where e and e' are the entities between which a relation is asserted, n is a degree of confidence in this correspondence and R is the relation, such as equivalence, holding between e and e' [9]. The approach assumes that there can be many possible mappings between two agent ontologies; these mappings are computed offline and stored by a dedicated agent *Ontology Alignment Service* [10]. Therefore, the OAS

1. This is the complexity of deciding whether an arguments belongs to the preferred extension of an agent, as explained in Sect. 3

provides the set of available *candidate mappings* that agents need to agree over. Hence, the argumentation process acts as a search over the space of all the candidate mappings in order to find those that are mutually acceptable according to the preferences expressed by all the agents involved. It is assumed that the OAS is able to provide a set of justifications G that motivate the existence of a mapping. The agents also have a private threshold value ϵ that is compared against the degree of confidence n that the OAS associates with each mapping. The private pre-ordering of preferences over the types of mappings, and the private threshold are used by the agents to argue in favour or against a correspondence, and to support their arguments with rational motivations based on these preferences.

The argumentation process is based on the *Value-Based Argumentation Framework* (VAF) [1].

Definition 3-1. A *Value-Based Argumentation Framework* (VAF) is defined as $\langle AR, A, \mathcal{V}, \eta \rangle$, where (AR, A) is an argumentation framework (AF), $\mathcal{V}$ is a set of k values which represent the types of arguments and $\eta : AR \rightarrow \mathcal{V}$ is a mapping that associates a value $\eta(x)\mathcal{V}$ with each argument $x \in AR$.

The VAF is an extension of Dung's [7] argumentation framework. , $\mathcal{V}$ is a set of 5 values representing the types of ontological mismatches that can occurr between ontologies, *i.e.* terminological, semantic, structural (internal and external), and extensional[2]. The element η denotes a mapping, $\eta : AR \rightarrow \mathcal{V}$, that associates a value $\eta(x) \in \mathcal{V}$ with each argument $x \in AR$. When arguing over ontology mappings using the VAF, an argument $x \in AR$ is a triple $x = \langle G, m, \sigma \rangle$ where m is a correspondence $\langle e, e', n, R \rangle$; G is the grounds justifying a prima facie belief that the correspondence does, or does not hold; and σ is one of $\{+, -\}$ depending on whether the argument is that m does or does not hold. The grounds justifying correspondences can be determined from the knowledge in ontologies, and this includes both the extensional and intensional OWL ontology definitions.

An argument x is *attacked* by the assertion of its negation (*counter-argument*), thus an argument $y \in AF$ *rebuts* an argument $x \in AF$ if x and y are arguments for the same mapping but with different signs, e.g. if x and y are in the form $x = \langle G_1, m, + \rangle$ and $y = \langle G_2, m, - \rangle$, x counter-argues y and vice-versa. The VAF allows us to associate a level of strength with arguments, to be used in addition to the set of attacks defined A.

In order to allow for capturing the notion of different agents having different perspectives on the same candidate mappings, the notion of *audience* is introduced.

Definition 3-2. An *audience* for a VAF is a binary relation $\mathcal{R} \subseteq \mathcal{V} \times \mathcal{V}$ whose (irreflexive) transitive closure, $\mathcal{R}^*$, is asymmetric, i.e. at most one of (v, v'), (v', v) are members of $\mathcal{R}^*$ for any distinct v, $v' \in \mathcal{V}$. We say that v_i *is preferred to* v_j *in the audience* $\mathcal{R}$, denoted $v_i \succ_{\mathcal{R}} v_j$, if $(v_i, v_j) \in \mathcal{R}^*$. Let $\mathcal{R}$ be an audience, α is a *specific audience* (compatible with $\mathcal{R}$) if α is a *total ordering* of $\mathcal{V}$ *and* $\forall\ v,\ v' \in \mathcal{V},\ \ (v, v') \in \alpha \Rightarrow (v', v) \notin \mathcal{R}^*$

The definition of audience is central to the notion of *acceptability* of an argument, since given a set of arguments, and their respective counter-arguments, the agents in an audience need to consider which of them they should accept. The acceptability of some arguments with respect to some audience, depends on the agents ability to determine a *preferred extension* that represents a consistent position within an argumentation framework, AF, that can be defended against all attacks, and cannot be further extended without causing it to be inconsistent or open to attacks. More formally:

Definition 3-3. Let $\langle AR, A, \mathcal{V}, \eta \rangle$ be a VAF and $\mathcal{R}$ an audience. For arguments x, y in AR, x is a *successful attack* on y *with respect to the audience* $\mathcal{R}$ if: $(x, y) \in \mathcal{A}$ *and* it is *not* the case that $\eta(y) \succ_{\mathcal{R}} \eta(x)$. An argument x is *acceptable to the subset* S with respect to an audience $\mathcal{R}$ if: for every $y \in AR$ that *successfully attacks* x with respect to $\mathcal{R}$, there is some $z \in S$ that successfully attacks y with respect to $\mathcal{R}$. A subset S of AR is *conflict-free with respect to the audience* $\mathcal{R}$ if: for each $(x, y) \in S \times S$, either $(x, y) \notin \mathcal{A}$ or $\eta(y) \succ_{\mathcal{R}} \eta(x)$. A subset S of AR is *admissible* with respect to the audience $\mathcal{R}$ if: S is conflict free with respect to $\mathcal{R}$ and every $x \in S$ is acceptable to S with respect to $\mathcal{R}$.

Definition 3-4. A subset S is a *preferred extension* for the audience $\mathcal{R}$ if it is a maximal admissible set with respect to $\mathcal{R}$.

In this framework it is assumed that an agent Ag_i is completely autonomous and has access to its individual ontology:

Definition 3-5. An agent Ag_i is defined by a 4-tuple $\langle O_i, VAF_i, Pref_i, \varepsilon_i \rangle$ where O_i is the OWL ontology; $VAF_i = \langle AR_i, A_i,$
$\mathcal{V}, \eta_i \rangle$ is the Valued-based Argumentation Framework of the agent Ag_i; $Pref_i$ is the private pre-ordering of preferences over $\mathcal{V}$ and ε_i is the private threshold value.

A set of agents $A = \{Ag_1, \ldots, Ag_n\}$ form a multi-agent system (MAS). The preferences and threshold selected by an agent depend on its context and on the agent's ontology, and its structural features, such as the depth of the subclass hierarchy and branching factor, ratio of properties to concepts, etc. The agent can then determine its preferences and threshold based on the characteristics of its ontology: an agent committing to an ontology lacking in structure will select a preference for terminological mapping, whilst it will prefer an extensional mapping if it commits to an ontology rich in instances. In Laera's framework, arguments and counter-arguments are generated by an agent using these preferences and thresholds.

3.1 Upper Bound For Argumentation Based Ontology Alignment

3.1.1 Version 1

A = set of all possible alignments, whereby $a \in A | a = \{m_1, m_2, \ldots, m_n\}$. Where $m = (e, e', \eta, r)$. Therefore, we can define the set N as the set of entities in O, such that $e \in N$; the set M as the set of entities in O', such that $e' \in M$;

2. We refer the interested reader to [12] for a detailed explanation of these types of mismatches.

the set R as the set of relations, such that $r \in R$. Thus, the upper bound of A can be defined as[3]:

$$|N| \times |M| \times |R| \quad (1)$$

3.1.2 Version 2

A = set of all possible alignments, whereby $a \in A | a = \{m_1, m_2, \ldots, m_n\}$. Where $m = (e, e', \eta, r, g)$. Therefore, we can define the set N as the set of entities in O, such that $e \in N$; the set M as the set of entities in O', such that $e' \in M$; the set R as the set of relations, such that $r \in R$ and the set G as the set of all possible grounds, such that $g \in G$. Thus, the upper bound of A can be defined as[4]:

$$|N| \times |M| \times |R| \times |G| \quad (2)$$

4 Ontology Modularization

The *Meaning-based argumentation* framework presented in [12] and recalled in Sect. **??** provides a mechanism for collaboratively searching over the space of all possible agent correspondences to locate those that are acceptable to both agents. Taking into account the messages that the agents are exchanging, the search space space can be reduced by modularizing the ontology with respect to a *signature*, consisiting of the named concepts mentioned in the messages. Depending on the mentioned concepts and on the ontologies involved in their definitions, the number of correspondences which refer to the module obtained can be much smaller than the number of original candidate correspondences, thus restricting the search space the agents have to explore.

This paper proposes to adopt an *ontology modularization* [4], [5] process to select a subset of the concepts on which the agents negotiate. The hypothesis is that reducing the ontology to a module corresponds to a reduction in the number of correspondences that are used in the negotiation process, which then selects those that are acceptable to all the agents involved in a transaction. Furthermore, the paper analyses to what extent the reduction in the search space affects the search, and whether all the modularization techniques behave equally when used as a filtering mechanism over the the search space (see Sect. 7).

An ontology, $\mathcal{O}$, is defined as a pair, $\mathcal{O} = (Ax(\mathcal{O}), Sig(\mathcal{O}))$, where $Ax(\mathcal{O})$ is a set of axioms (intensional, extensional and assertional) describing the entities e (classes, properties, and instances) in the ontology $\mathcal{O}$ and $Sig(\mathcal{O})$ is the signature of $\mathcal{O}$, that is the set of entity names used by $\mathcal{O}$, i.e., its vocabulary[5]. The notion of *ontology module extraction* can thus be more formally defined as:

Definition 4-1. *Ontology module extraction* extracts a consistent[6] module M from an ontology $\mathcal{O}$ that covers a specified signature $Sig(M)$, such that $Sig(M) \subseteq Sig(\mathcal{O})$.

M is the relevant part of $\mathcal{O}$ that is said to cover the elements defined by $Sig(M)$, as such $M \subseteq \mathcal{O}$. M is an ontology itself and it is possible that further modules could be extracted from it.

Ontology module extraction approaches can be split into two distinct categories: logical approaches and traversal approaches. Logical approaches [4] focus on maintaining the logical properties of coverage and minimality when extracting modules. Traversal approaches [5], [14], [16] represent the extraction as a graph traversal, with the module being defined by the conditional traversal of a graph.

Logical approaches are based on the notion of conservative extension [13]: an ontology $\mathcal{O}_1 \cup \mathcal{O}_2$ is a conservative extension of one of its modules $\mathcal{O}_1$ for a signature Sig if and only if every logical consequence of $\mathcal{O}_1 \cup \mathcal{O}_2$ formulated using only symbols from Sig, is already a logical consequence of $\mathcal{O}_1$. In other words, if adding the ontology $\mathcal{O}_2$ to $\mathcal{O}_1$ does not change the ontology $\mathcal{O}_1$ for what concerns the concepts that are built *only* from the concept and role names in the signature Sig. Logical approaches define a module $\mathcal{O}_i$ within an ontology $\mathcal{O}$ as a subset of $\mathcal{O}$ such that $\mathcal{O}$ is a conservative extension of $\mathcal{O}_i$ w.r.t. the concept and role names that belong to $Sig(\mathcal{O}_i)$. Cuenca Grau *et al* [4] use conservative extensions to define the notion of *safety*: $\mathcal{O}_1$ is *safe* for $\mathcal{O}_2$ if $\mathcal{O}_1 \cup \mathcal{O}_2$ is a conservative extension of $\mathcal{O}_2$. Lutz *et al* [13] have shown that deciding if an ontology is a conservative extension of an ontology module, and thus its safety, is undecidable for OWL-DL. However, Cuenca Grau *et al.* [4] have proposed a number of sufficient conditions for safety, for example *locality*: if these conditions are satisfied by an ontology, then it is safe, but the converse does not necessarily hold. Testing for locality in expressive description logics (such as the one underlying OWL-DL) is decidable, but the ontology modules defined by means of locality are not guaranteed to be minimal, as those defined by conservative extensions. Jiménez-Ruiz *et al.* [11] defined two variants of locality, namely $\perp$ and $\top$, depending on whether an Ontology Engineer is modularising in order to generalise or specialise the reused concepts. Both types of locality imply safety, and they allow the definition of two extraction techniques, one for the *upper* module (corresponding to testing for $\perp$-locality), and one for the *lower* module (corresponding to testing for $\top$-locality).

Extraction methods based on graph traversal utilise a graph representation of the ontology, where an ontology $\mathcal{O}$ corresponds to the graph $\mathcal{O} = (N, V)$ where the set of nodes N is the set of concept names, and the vertices in V are the relations between the concepts, such as property restrictions and subsumption relationships[7]. Ontology modules are defined as a conditional traversal on the graph $\mathcal{O}$. Seidenberg and Rector [16] aim to include all the elements (concepts and restrictions) that participate, either directly or indirectly, in the definition of the already included entities. Assuming that the concept A is in the module then, all of A's superclasses and subclasses are included; but its sibling classes are not. The

3. η is not considered here as it is an infinite continuum
4. η is not considered here as it is an infinite continuum
5. This definition is agnostic with respect to the language used to represent the ontology, but the modularization techniques in this paper assume OWLas a language, and therefore Description Logic based semantics.
6. OWL is monotonic and hence guarantees consistency if the extraction is done on a consistent ontology.
7. Graph traversal extraction methods are possible since OWL ontologies map to RDF graphs (see http://www.w3.org/TR/owl-semantics/)

restrictions (intersection, union and equivalent) of the already included classes can now be added to the module. Finally, links across the hierarchy from the previously included classes are traversed; the target of these also have their superclasses included.

In contrast to Seidenberg *et al*, Doran *et al* [5] include all the subclasses of the input signature, but none of the superclasses. The aim is to include everything that is defined by the input signature in a tractable time (the approach has polynomial complexity), thus all relations between these subclasses are included. The only exception is that in the first step of the traversal, disjoint classes are not included.

The approach by Noy and Musen [14] proposes a module extraction technique based on *traversal views*, i.e. a set of directives that guide the traversal of the ontology graph, and in particular for defining the length of the paths that will be followed along different types of relationships. This approach allows domain experts to specify explicitly which subset of the ontology they are interested in, and therefore is user led.

WE NEED D'AQUIN HERE

4.1 Combining Ontology Modularization and Argumentation

1.	Ag_1 asks a query, *query*$(A \in Sig(O))$, to Ag_2.
2.	Ag_2 does not understand the query, $A \notin Sig(O')$, and informs Ag_1 they need to use an Ontology Alignment Service (OAS)
3.	Ag_1 produces, $om(O, Sig(A))$, an ontology module, M, to cover the concepts required for its task.
4.	Ag_1 and Ag_2 invoke the OAS. Ag_1 sends its ontology, O and the signature of M, $Sig(M)$.
5.	The OAS aligns the two ontologies and filters the correspondences according to M. Only those correspondences featuring an entity from M are returned to both agents.
6.	The agents begin the Meaning-Based Argumentation process, and iterate it, with each agent generating arguments and counter-arguments.
7.	The iteration terminates when the agents reach an agreement on a set of correspondences, and this set is returned to both agents.
8.	Ag_1 asks a query to Ag_2 but uses the correspondences so that Ag_2 understands, query$(A \in Sig(O) \wedge B \in Sig(O'))$ where A and B are aligned.
9.	Ag_2 answers the query making use of the resulting alignment.

TABLE 1
Steps involved in Ontology Modularization and Argumentation

Ontology modularization can be used as a pre-processing step to improve the efficiency of an argumentation framework, when used to search the space of all candidate ontology mappings. When two agents communicate, only the initiating agent (Ag_1) is aware of its task, and consequently, what concepts are relevant to this task. It can therefore select these relevant concepts within the signature of the desired ontology module. The signature of the resulting ontology module can then be used to filter the correspondences, and consequently the number of arguments necessary within the argumentation process. The steps in Table 1 describe this process, whilst Figure 1 depicts the process as a UML Sequence Diagram. It is assumed that two agents, Ag_1 and Ag_2 have ontologies O and O' respectively.

The set of ontology correspondences are filtered at Step 5 according to the following function:

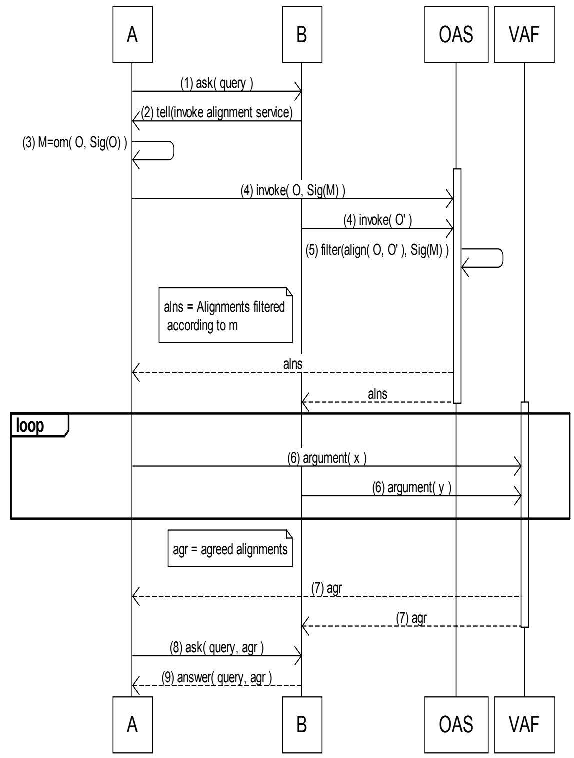

Fig. 1. UML Sequence Diagram of Ontology Modularization and Argumentation.

Definition 4-2. A *filtering function*, filter(), filters the set of candidate mappings prior to argumentation Z into a subset $Z' \subseteq Z$ such that $filter(Z, Sig(M)) : Z \rightarrow Z' \mid \forall m \in Z'$, $m = \langle e, e', n, R \rangle$ and $e \in Sig(M)$.

Steps 6 and 7 represent a black-box process, which is the argumentation process. Modularization is therefore used to filter the correspondences that are passed to this process. The combination of these two processes reduces the cost of reaching an agreement over the set of correspondences, by reducing the size of the set of correspondences, and hence the number of arguments.

5 Possibility For Information Loss

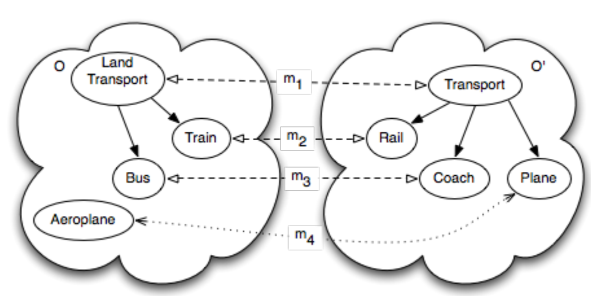

Fig. 2. Possible scenario where information loss could occur.

In the work presented in this paper it is assumed that an agent *knows all* the concepts that are relevant to its task. However, in practice it may be possible that this assumption does not hold. Consider the scenario shown in Figure 2. O belongs to Ag_1 and O' belongs to Ag_2, in the case where no modularization is used, the following alignments are used: $m_1 = \langle O : \mathtt{a}, O' : \mathtt{a}', 1, = \rangle$, $m_2 = \langle O : \mathtt{b}, O' : \mathtt{b}', 1, = \rangle$, $m_3 = \langle O : \mathtt{c}, O' : \mathtt{c}', 1, = \rangle$, $m_4 = \langle O : \mathtt{d}, O' : \mathtt{d}', 1, = \rangle$. If Ag_1 were to ask Ag_2 to give them all the instances of a then Ag_1 would be able to also make sense of the concept d, even if it doesn't know a property that connects them.

Now consider the case where modularization is used, Ag_1 extracts a module from O with the $Sig(\mathtt{a}, \mathtt{b}, \mathtt{c})$ and this produces the ontology module M that inlcudes a,b,c and the properties that link them. When the possible ontology alignments are filtered according to this M, the set of alignments that the agents would argue over is: $m_1 = \langle O : \mathtt{a}, O' : \mathtt{a}', 1, = \rangle$, $m_2 = \langle O : \mathtt{b}, O' : \mathtt{b}', 1, = \rangle$, $m_3 = \langle O : \mathtt{c}, O' : \mathtt{c}', 1, = \rangle$. The possible fourth alignment in this case is not considered, as the entity for d is not in M, however, when Ag_2 answers the query give me all the instances of a it can decide whether or not to return the information regarding d'. In the case that it decides not to then there is no problem, in this case the assumption is that Ag_1 knows all the concepts that are relavant to its task. In the other case where it does return the information regarding d' then Ag_1will not be able to make full use of it as there is no alignment between d and d'.

5.1 Preventing Information Loss

A way to deal with the possible loss of information is for Ag_2 to also carry out a modularization step. Once the alignments have been filtered Ag_2 uses the entities identified as a signature for modularization and filters the mappings according to its module. There are two possible ways for Ag_2 to carry out this process:

Sol. 1 Ag_1 filters the alignment according to its $Sig(M)$, which would produce the set of alignments A. Ag_2 then uses A as the input to its own $om()$. Ag_2 now filters the alignments according to its $Sig(M')$. This strategy is labelled $COMPLETE$ (shortened to C).

Sol. 2 Rather than using A Ag_2 uses a subset defined by the $Sig()$ by Ag_1. This strategy is labelled $SIGNATURE-ONLY$ (shortened to SO).

The third possible approach (labelled $NONE$, shortened in N) is to accept the possibility of information loss and therefore Ag_2 is not supposed to do any modularization; the three strategies are used in the evaluation and the results are illustrated in Sect. 7.

Either of the two solutions would solve the problem presented in the example above. However, since Ag_2 may identify new, possibly relevant, entities Ag_1 should now also include the previously missing entities to the signature of its ontology module. Evidently this recursive process could be expensive, and both agents could end up with ontology modules equal to the original ontology. One possible way to prevent this would be via conservative extensions [13] that guarantee inferential completeness, but due to their undecidability above $\mathcal{EL}^{++}$ the relaxation of the assumption made in this paper is tested further in Sect. 7, where the effect of the two possible solutions above are investigated.

6 An Illustrative Example

This simple example illustrates the ideas presented previously[8] and relates them to the steps identified in Sect. 4.1. Assume that we have two agents; Ag_1 wishes to ask a query of Ag_2(Step 1), Ag_1 wants to know the instances of Paper_Author. Ag_1 uses O_1 the EKAW ontology[9] and Ag_2 uses O_2 the OpenConf[10] ontology. Due to space constraints we only show a subset of these ontologies, in Figure 3. Until the agents have aligned their ontologies Ag_2 (Step 2) will be unable to fulfil the request of Ag_1.

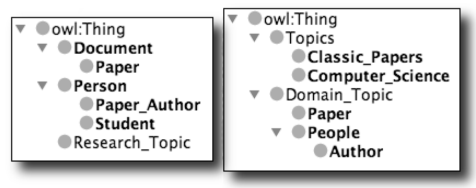

Fig. 3. Ontologies O_1 and O_2

Ag_1 knows the concepts that are relevant for its task and extracts an ontology module (Step 3), M, in this example the M will only include the concept Paper_Author. Now when the agents invoke the Ontology Alignment Service (OAS) Ag_1 will send its ontology O_1 and the signature of M ($Sig(M) = \{\mathtt{Paper\_Author}\}$)(Step 4). The OAS produces the following set of possible correspondences:

$m_1 = \langle O_1 : \mathtt{Paper\_Author}, O_2 : \mathtt{Author}, 0.45, = \rangle$
$m_2 = \langle O_1 : \mathtt{Paper\_Author}, O_2 : \mathtt{Paper}, 0.54, = \rangle$
$m_3 = \langle O_1 : \mathtt{Research\_Topic}, O_2 : \mathtt{Topics}, 0.44, = \rangle$
$m_4 = \langle O_1 : \mathtt{Research\_Topic}, O_2 : \mathtt{Domain\_Topic}, 0.44, = \rangle$
$m_5 = \langle O_1 : \mathtt{Person}, O_2 : \mathtt{People}, 0.9, = \rangle$

The OAS will now filter the alignments according to the Sig(M) using the function defined in Sect. 4.1 (Step 5). The result of this process is the following reduced set of ontology correspondences:

$m_1 = \langle O_1 : \mathtt{Paper\_Author}, O_2 : \mathtt{Author}, 0.45, = \rangle$
$m_2 = \langle O_1 : \mathtt{Paper\_Author}, O_2 : \mathtt{Paper}, 0.54, = \rangle$

This reduced set of ontology correspondences will now be used as input to the argumentation process(Step 6). The preference ordering that the agents possess affects how the argumentation phase advances. However, this preference should not affect the premise that the fewer alignments there are to argue over then the fewer arguments that are generated. If we assume now that Ag_1 prefers terminological to external structure (T $\succ$ ES), whilst Ag_2 prefers external structure to terminological (ES $\succ$ T). Thus, Ag_1 accepts both m_1 and m_2, whilst Ag_2 accepts m_1 and rejects m_2.

The arguments and counter arguments made during the argumentation phase are shown in Table **??**. This set of arguments allows the agents to build the argument graph, shown in Figure 4; whereby the nodes represent the arguments and the arcs represent the attacks relation, with the direction indicating the direction of attack.

8. This example builds on the one presented in [12]
9. http://nb.vse.cz/ svabo/oaei2006/data/ekaw.owl
10. http://nb.vse.cz/ svabo/oaei2006/data/OpenConf.owl

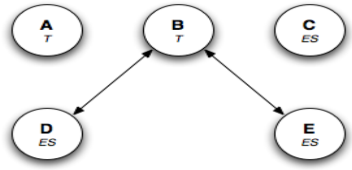

Fig. 4. Attack graph for the illustrative example.

The graph shows that both arguments A and C support m_1, but that arguments D and E argue against m_2 whilst B is in favour; thus D and E form a symmetric attach against C. Given the different preferences of the two agents, the preferred extensions are shown in Table 2. The argument that are accepted by both agents are A and C. Thus, the agreed alignment returned to both agents (Step 7) is m_1. If the two agents had argued over the alignments before modularization then m_1, m_3, m_4 and m_5 would have been the agreed alignments.

TABLE 2
Arguments for and against m_1 and m_2

Agent	Preferred Extensions
Ag_1	$\{A,C,B\}$,$\{A,C\}$
Ag_2	$\{A,C\}$,$\{A,C,D\}$,$\{A,C,E\}$,$\{A,C,D,E\}$

Now Ag_1 can ask $Ag_2$the query (Step 8) and Ag_2 is able to answer (Step 9) due to the agreed set of alignments.

7 Evaluation

The evaluation has two main objectives: (a) to quantify the impact of the use of modularization techniques on the number of mappings that the argumentation process receives as input; (b) to quantitatively evaluate the quality of the resulting alignments compared with the alignments obtained without modularization. These two objectives are explained in the following sections.

7.1 Ontologies and Tracks

Objective (a) requires a diversified set of ontologies, ideally covering different domains, and a diversified set of alignment techniques for the generation of alignments. This is required to overcome the bias deriving from the alignment technique being used, since a specific technique might produce extremely small or extremely large modules, thus skewing the reduction results, but the modules produced might not be useful for the agents task. On the other hand, it is not easy to find extensive sets of ontologies covering different domains and for which there are reliable or verifiable mappings available, so a tradeoff must be chosen.

The eleven ontologies used in the evaluation were taken from the *OAEI 2007 Conference Track* repository (with the exception of three ontologies whose memory requirements for reasoning were over 2 GB). The ontologies are listed in Table 3, complete with a brief characterization in terms of the number of classes and properties, and the level of DL expressivity used to represent them.

The alignment techniques available are those used by each system participating in the OAEI tracks; in order to simplify the experiment setup, the systems themselves are not involved in the evaluation; the Alignment API is used instead to access the submitted results. For the chosen track, five systems have submitted a sufficient number of alignments for an overall comparison, i.e., they align each ontology with the others; some systems also provide reverse alignments, i.e., the alignments are of the form $O_A \leftrightarrow O_B$ and $O_B \leftrightarrow O_A$.

The modularization techniques used in this evaluation are some of those described in Sect. 4.1; in particular, Cuenca Grau's *lower* and *upper* techniques, d'Aquin's, Doran's and Seidenberg's have been used; the implementations for these techniques are all available, and only minor modifications to the code have been implemented, to enable them to work in our experimental framework. For clarity, the techniques have been labelled $Techs = \{BASELINE$, $CUENCAGRAU^U$, $CUENCAGRAU^L$, $DAQUIN$, $DORAN$, $SEIDENBERG\}$ (shortened to $\{B$, CG^U, CG^L, DAQ, DOR, $SEID\}$); $BASELINE$ corresponds to the original alignment, i.e., the alignment as is produced by the alignment techniques, and is used as baseline for averaging the results.

7.2 Quality of Alignments

Objective (b) requires either a gold standard alignment as a reference to evaluate the resulting mappings, or a method to compare the reduced alignments with the original ones, i.e., a way to verify whether a reduced alignment is equivalent to the original one with respect to correctness and completeness of the results; for the latter evaluation, the following use case is being considered, based on query answering:

Agent A_1 and agent A_2 engage in communication; A_1 is asking queries of the kind $Q_I = \{x \mid x\ is\ a\ Y\}$, i.e., instance retrieval queries, to A_2; alternatively, queries of the kind $Q_{sup} = \{x \mid x\ is\ a\ superclass\ of\ Y\}$ and $Q_{sub} = \{x \mid x\ is\ a\ subclass\ of\ Y\}$, i.e., queries that explore the concepts' structure.

The original alignment M_0 enables A_2 to give the set of answers X_0; given a reduced alignment M_i, the corresponding set of answers X_i is computed and compared to X_0.

To do this, three $retention$ measures are defined:

Definition 7-1 (Instance Retention). Given an OWL ontology O, an OWL class C not defined in O and two alignments M_0 and M_i, with $M_i \subseteq M_0$, $instance\ retention$

$$IR : \{O, C, M_0, M_i\} \rightarrow [0, 1]$$

is the function defined as the number of instances of C described in $O \sqcup M_i$ divided by the number of instances of C described in $O \sqcup M_0$.

Similar definitions can be given for $Subclass\ Retention$ and $Superclass\ Retention$:

Definition 7-2 (Subclass Retention). Given an OWL ontology O, an OWL class C not defined in O and two alignments

M_0 and M_i, with $M_i \subseteq M_0$, $subclass\ retention$

$$SubR : \{O, C, M_0, M_i\} \rightarrow [0, 1]$$

is the function defined as the number of subclasses of C defined in $O \sqcup M_i$ divided by the number of subclasses of C defined in $O \sqcup M_0$.

Definition 7-3 (Superclass Retention). Given an OWL ontology O, an OWL class C not defined in O and two alignments M_0 and M_i, with $M_i \subseteq M_0$, $superclass\ retention$

$$SupR : \{O, C, M_0, M_i\} \rightarrow [0, 1]$$

is the function defined as the number of superclasses of C defined in $O \sqcup M_i$ divided by the number of superclasses of C defined in $O \sqcup M_0$.

These functions can be used to quantify what information is lost by using a smaller alignment M_i, obtained by using a modularization technique, in place of the original alignment M_0, taking into account the task at hand, which is represented by the OWL classes used as signature for the modularization process. The closest the result of a retention measure is to 1, the less information is being lost due to the reduction in the alignment size.

Retention functions work well under the hypothesis that for all possible OWL classes on which the measure is computed there are instances available (respectively, subclasses and superclasses). In the ontologies used in the evaluation, this is not always true, i.e., some concepts do not have instances and some classes do not have superclasses or subclasses defined in the ontology. Artificial instances can be added, but it is not meaningful to add artificial subclasses or superclasses in the cases in which they are missing; those cases are left out of the current evaluation.

It is possible that for some concepts there are no alignments that generate answers, i.e., there is no correspondence that generates an answer for C with respect to $O \sqcup M_0$, therefore the retention measures give $0/0$ indetermined form. These cases are not averaged in the evaluation, since they depend on the quality of the alignment itself, i.e., on the suitability of the original alignment for communication, and not on the modularization techniques or on the argumentation process; therefore, evaluating and discussing these results would be outside the scope of this paper.

A corner case is raised when the alignment M_i is empty, i.e., when the modularization techniques generate modules that do not refer to any concept mentioned in the mappings. In this case, the retention measures can only report 0 or $0/0$ indetermined form.

A value of 0 would mean that some answers that were available with respect to M_0 have been lost by reducing the alignment, while a $0/0$ would instead mean that there was no retrievable answer; in this second case, reducing the alignment to 0 has the advantage of skipping the argumentation and the query answering phases, since it is already known that no answers will be generated.

The number of cases in which this last hypothesis is true has been evaluated in the current experiments, and there is indeed evidence that this hypothesis is wrong only in 0.15% of the cases, i.e., trusting an empty alignment obtained from modularization to signify that no answers are available is wrong in one case over six hundred. More details are given in the following.

Ontology name	# of classes	# of properties	DL expressivity
cmt	31	64	ALCHIF(D)
Conference	61	69	ALCHIF(D)
confOf	40	41	SHIF(D)
crs_dr	16	22	ALCHIF(D)
edas	105	55	ALCHIF(D)
ekaw	75	38	SHIN
MICRO	33	31	ALCHOIF(D)
OpenConf	64	50	ALCHOI(D)
paperdyne	47	83	ALCHOIF(D)
PCS	25	43	ALCHIF(D)
sigkdd	51	33	ALCHI(D)

TABLE 3
Classes, properties and DL expressivity for the OAEI ontologies.

7.3 Evaluation setup

The experiments are divided into *runs*; each run is described as a tuple $< O_1, O_2, A, T, S, I >$, where:

- O_1 and O_2 are distinct ontologies from the same track (the order is important, so each track produces $n*(n-1)$ pairs);
- A is the alignment being produced by a specific alignment technique on the pair of ontologies, and therefore a set of mappings;
- T is a modularization technique, with S being the signature used for the extraction process;
- I is the technique being used to fix the information loss problem described in Sect. 5.1.

The modularization technique T is one of the elements in $Techs$; when $T = B$, i.e., the baseline, no modularization happens, and therefore the signature S is ignored, as well as I. For the other values of T, three new runs are generated for each named concept C in O_1, with $S = \{C\}$ and I being one of $\{NONE,\ COMPLETE,\ SIGNATURE-ONLY\}$ (shortened to $\{N,\ C,\ SO\}$).

Called $NC(O)$ the number of named concepts in ontology O and n the number of ontologies in a track, therefore, the number of runs with O as first ontology for a track is $6*(3*NC(O)+1)*(n-1)$.

Information being recorded includes:

- for each ontology, number of concepts, properties, anonymous concepts and DL expressivity (in Table 3);
- for each module with $|S| = 1$, number of concepts, properties, anonymous concepts and DL expressivity;
- for each pair of ontologies (O_1, O_2)
  - Modules extracted from O_1, and percentage reduction in concept and property number wrt the original ontology (a sketch of these values is given in Table **??**);
  - Number of mappings and arguments being generated without modularization (baseline technique B,

	CG^U	CG^L	DAQ	DOR	$SEID$	Total
ALC	32	32	2	31	145	242
ALC(D)	2	2	0	0	0	4
ALCF	1	1	25	0	0	27
ALCF(D)	17	17	0	0	170	204
ALCH(D)	10	10	0	0	0	20
ALCHI	18	18	0	4	0	40
ALCHIF	32	32	0	38	0	102
ALCHIF(D)	6	6	0	0	0	12
ALCHIN	7	7	0	0	0	14
ALCHOF(D)	0	0	0	0	45	45
ALCHOI	10	10	0	17	0	37
ALCHOIF	25	25	0	0	0	50
ALCN	0	0	0	0	73	73
ALCO	0	0	0	0	62	142
ALCOI	53	53	0	5	31	62
EL	196	198	349	363	0	1106
EL+	90	90	103	69	0	352
SHI	7	7	0	0	0	14
Classes	6.02	6.04	2.22	14.78	62.36	18.82
stdev	8.42	8.42	3.38	28.02	25.82	28.78
Object Properties	1.94	1.95	0	1.15	11.49	3.33
stdev	5.16	5.17	0	3.75	7.06	6.4
Datatype Properties	0.28	0.28	0	0	2.81	0.68
stdev	1.17	1.17	0	0	3.78	2.15
Anonymous Classes	4.98	5	0.43	3.08	33.52	9.81
stdev	11.13	11.14	1.42	10.48	25.96	19.01

TABLE 4
Modules statistics: DL expressivity and number of modules (upper section) and average and standard deviation for number of classes, object and datatype properties, and anonymous classes (lower section).

		Avg size with mod.			Avg retained size		
System	MT	C	SO	N	C	SO	N
Asmov	CG^L	1.09	1.49	0.93	9.23%	12.08%	8.05%
	CG^U	1.09	1.49	0.93	9.23%	12.08%	8.05%
B size	DAQ	1.77	2.19	1.49	14.16%	17.27%	11.87%
13.22	DOR	1.79	2.01	1.41	13.97%	15.33%	10.98%
	$SEID$	8.94	10.50	8.68	71.64%	81.22%	69.96%
Falcon	CG^L	1.45	2.14	1.07	11.38%	16.36%	8.40%
	CG^U	1.45	2.14	1.07	11.38%	16.36%	8.40%
B size	DAQ	2.93	3.33	2.59	22.26%	25.53%	19.69%
13.11	DOR	2.61	2.73	2.21	20.69%	21.42%	17.61%
	$SEID$	10.55	11.56	10.23	82.37%	89.50%	79.99%
Lily	CG^L	5.95	8.15	3.67	12.70%	17.13%	7.88%
	CG^U	5.95	8.15	3.67	12.70%	17.13%	7.88%
B size	DAQ	10.13	11.77	6.71	22.05%	25.55%	14.50%
46.95	DOR	9.40	8.85	5.81	18.95%	17.81%	11.69%
	$SEID$	35.83	37.38	33.45	76.32%	79.53%	71.55%
Ola	CG^L	0.25	0.25	0.25	0.70%	0.70%	0.70%
	CG^U	0.25	0.25	0.25	0.70%	0.70%	0.70%
B size	DAQ	0.15	0.15	0.15	0.41%	0.41%	0.41%
45.05	DOR	0.38	0.38	0.38	1.03%	1.03%	1.03%
	$SEID$	3.20	3.20	3.20	8.60%	8.60%	8.60%
Ontodna	CG^L	0.71	0.84	0.65	10.40%	11.94%	9.70%
	CG^U	0.71	0.84	0.65	10.40%	11.94%	9.70%
B size	DAQ	1.32	1.47	1.23	21.14%	23.19%	19.96%
5.92	DOR	0.68	0.72	0.61	10.88%	11.34%	9.94%
	$SEID$	4.12	4.75	4.06	70.30%	79.62%	69.44%

TABLE 7
Average alignment sizes with and without modularization

		Avg size with mod.			Avg retained size		
System	MT	C	SO	N	C	SO	N
Asmov	CG^L	2.36	3.09	1.99	19.00%	23.95%	16.15%
	CG^U	2.36	3.09	1.99	19.00%	23.95%	16.15%
B size	DAQ	4.23	5.26	3.47	33.89%	41.64%	27.49%
13.22	DOR	4.69	5.20	3.69	36.87%	40.32%	28.92%
	$SEID$	8.94	10.50	8.68	71.64%	81.22%	69.96%
Falcon	CG^L	2.93	4.07	2.21	23.59%	31.68%	17.92%
	CG^U	2.93	4.07	2.21	23.59%	31.68%	17.92%
B size	DAQ	6.30	7.23	5.36	47.96%	55.35%	40.80%
13.11	DOR	6.29	6.57	5.43	50.66%	52.45%	44.05%
	$SEID$	10.56	11.56	10.24	82.37%	89.50%	79.99%
Lily	CG^L	7.45	10.13	4.58	16.19%	21.62%	10.11%
	CG^U	7.45	10.13	4.58	16.19%	21.62%	10.11%
B size	DAQ	15.23	17.78	9.76	34.23%	39.89%	21.87%
46.95	DOR	13.93	13.17	8.97	29.40%	27.79%	19.06%
	$SEID$	35.84	37.38	33.45	76.32%	79.53%	71.55%
Ola	CG^L	4.85	4.85	4.85	13.55%	13.55%	13.55%
	CG^U	4.85	4.85	4.85	13.55%	13.55%	13.55%
B size	DAQ	1.87	1.87	1.87	5.44%	5.44%	5.44%
45.05	DOR	4.68	4.68	4.68	13.20%	13.20%	13.20%
	$SEID$	35.20	35.20	35.20	94.61%	94.61%	94.61%
Ontodna	CG^L	1.97	2.23	1.81	36.32%	39.30%	33.57%
	CG^U	1.97	2.23	1.81	36.32%	39.30%	33.57%
B size	DAQ	3.48	3.88	3.14	60.01%	65.71%	54.97%
5.92	DOR	2.53	2.68	2.27	44.20%	46.22%	40.90%
	$SEID$	4.29	4.93	4.22	73.00%	82.68%	72.11%

TABLE 8
Average alignment sizes with and without modularization (excluding alignments of size 0)

values in columns *Original size (B)* and *Original alignments* in Table 5 and Table 6);

- Number of mappings and arguments resulting from modularization with each value for T and I (Table 5 and Table 6);
- Number of mappings being accepted and rejected by the argumentation process for all cases above (Table 7 and Table 8);

Where relevant, tables report also the values for each run excluding the cases in which the reduced alignments had size 0, i.e. no correspondences were found to be relevant.

7.4 Results Discussion

The results presented in Table 13 and in Table 14 show that the overall impact of using a modularization technique for reducing the amount of candidate correspondences can vary from 57% to 95% (*candidate* column in both tables), depending on the modularization technique being chosen; the impact of the information loss solution is more contained: whichever technique is being used, the expected reduction in candidate alignments is close to 80% and the search space for the argumentation is reduced from 75% to 80%.

Paired with the results presented in Table 11, that evaluate the quality of the resulting alignments in terms of the retention measures defined in Sect. 7.2, the data outlines the following conclusion: there is a tradeoff between reduction of the argumentation search space and retention, where to a reduction of the search space close to 95% (for the CG^U and CG^L techniques) corresponds the highest loss on retention, averaged at 0.05%. Some examples, reported in Table 12, show that in a very restricted number of cases these techniques can lead to higher loss in retention, when coupled with lack of information loss solutions. On the other hand, lower impact on the number of candidate correspondences, such as in the $SEID$ technique, produces maximal retention values, thus guaranteeing that no information is being lost; however, the average reduction in the search space for this technique (see Table 10) is slightly lower

		Original	Original	Retained alignment size and # of arguments						Avg percent					
System	MT	size (B)	arguments	C M	C A	SO M	SO A	N M	N A	C M	C A	SO M	SO A	N M	N A
Asmov	CG^L	13.22	26.45	0.94	2.98	0.94	2.18	0.93	1.87	7.10%	11.27%	7.10%	8.26%	7.10%	7.10%
	CG^U			0.94	2.98	0.94	2.18	0.93	1.87	7.10%	11.27%	7.10%	8.26%	7.10%	7.10%
	DAQ			1.49	4.38	1.49	3.55	1.49	2.99	11.31%	16.56%	11.31%	13.44%	11.31%	11.31%
	DOR			1.41	4.01	1.41	3.59	1.41	2.83	10.70%	15.18%	10.70%	13.60%	10.70%	10.70%
	$SEID$			8.68	21.00	8.68	17.88	8.68	17.36	65.64%	79.38%	65.64%	67.59%	65.64%	65.64%
Falcon	CG^L	13.10	26.21	1.07	4.28	1.07	2.90	1.07	2.14	8.17%	16.35%	8.17%	11.09%	8.17%	8.17%
	CG^U			1.07	4.28	1.07	2.90	1.07	2.14	8.17%	16.35%	8.17%	11.09%	8.17%	8.17%
	DAQ			2.59	6.67	2.59	5.86	2.59	5.19	19.81%	25.46%	19.81%	22.37%	19.81%	19.81%
	DOR			2.21	5.47	2.21	5.23	2.21	4.42	16.88%	20.89%	16.88%	19.95%	16.88%	16.88%
	$SEID$			10.23	23.12	10.23	21.11	10.23	20.47	78.09%	88.21%	78.09%	80.55%	78.09%	78.09%
Lily	CG^L	46.94	93.89	3.67	16.30	3.67	11.90	3.67	7.35	7.83%	17.37%	7.83%	12.68%	7.83%	7.83%
	CG^U			3.67	16.30	3.67	11.90	3.67	7.35	7.83%	17.37%	7.83%	12.68%	7.83%	7.83%
	DAQ			6.71	23.55	6.71	20.26	6.71	13.43	14.31%	25.09%	14.31%	21.59%	14.31%	14.31%
	DOR			5.81	17.71	5.81	18.81	5.81	11.63	12.39%	18.87%	12.39%	20.03%	12.39%	12.39%
	$SEID$			33.45	74.76	33.45	71.67	33.45	66.90	71.26%	79.63%	71.26%	76.34%	71.26%	71.26%
Ola	CG^L	65.56	131.12	0.40	0.81	0.40	0.81	0.40	0.81	0.62%	0.62%	0.62%	0.62%	0.62%	0.62%
	CG^U			0.40	0.81	0.40	0.81	0.40	0.81	0.62%	0.62%	0.62%	0.62%	0.62%	0.62%
	DAQ			0.25	0.50	0.25	0.50	0.25	0.50	0.39%	0.39%	0.39%	0.39%	0.39%	0.39%
	DOR			0.70	1.40	0.70	1.40	0.70	1.40	1.07%	1.07%	1.07%	1.07%	1.07%	1.07%
	$SEID$			5.62	11.25	5.62	11.25	5.62	11.25	8.58%	8.58%	8.58%	8.58%	8.58%	8.58%
Ontodna	CG^L	5.91	11.83	0.65	1.69	0.65	1.42	0.65	1.31	11.13%	14.31%	11.13%	12.02%	11.13%	11.13%
	CG^U			0.65	1.69	0.65	1.42	0.65	1.31	11.13%	14.31%	11.13%	12.02%	11.13%	11.13%
	DAQ			1.23	2.95	1.23	2.65	1.23	2.47	20.89%	24.95%	20.89%	22.43%	20.89%	20.89%
	DOR			0.61	1.45	0.61	1.37	0.61	1.22	10.32%	12.31%	10.32%	11.64%	10.32%	10.32%
	$SEID$			4.06	9.50	4.06	8.25	4.06	8.12	68.70%	80.28%	68.70%	69.75%	68.70%	68.70%

TABLE 5
Average candidate alignment sizes with and without modularization

		Original	Original	Retained alignment size and # of arguments						Avg percent					
System	MT	size (B)	arguments	C M	C A	SO M	SO A	N M	N A	C M	C A	SO M	SO A	N M	N A
Asmov	CG^L	13.23	26.45	1.99	6.21	1.99	4.74	1.99	3.97	15.01%	23.47%	15.01%	17.93%	15.01%	15.01%
	CG^U			1.99	6.21	1.99	4.74	1.99	3.97	15.01%	23.47%	15.01%	17.93%	15.01%	15.01%
	DAQ			3.47	10.51	3.47	8.47	3.47	6.94	26.23%	39.74%	26.23%	32.01%	26.23%	26.23%
	DOR			3.69	10.40	3.69	9.37	3.69	7.38	27.91%	39.30%	27.91%	35.43%	27.91%	27.91%
	$SEID$			8.68	21.00	8.68	17.88	8.68	17.36	65.64%	79.38%	65.64%	67.59%	65.64%	65.64%
Falcon	CG^L	13.11	26.22	2.21	8.22	2.21	5.89	2.21	4.43	16.89%	31.34%	16.89%	22.47%	16.89%	16.89%
	CG^U			2.21	8.22	2.21	5.89	2.21	4.43	16.89%	31.34%	16.89%	22.47%	16.89%	16.89%
	DAQ			5.36	14.46	5.36	12.59	5.36	10.71	40.85%	55.14%	40.85%	48.02%	40.85%	40.85%
	DOR			5.43	13.15	5.43	12.58	5.43	10.86	41.43%	50.15%	41.43%	47.97%	41.43%	41.43%
	$SEID$			10.24	23.13	10.24	21.12	10.24	20.47	78.09%	88.21%	78.09%	80.55%	78.09%	78.09%
Lily	CG^L	46.94	93.89	4.58	20.58	4.58	15.11	4.58	9.15	9.75%	21.92%	9.75%	16.09%	9.75%	9.75%
	CG^U			4.58	20.58	4.58	15.11	4.58	9.15	9.75%	21.92%	9.75%	16.09%	9.75%	9.75%
	DAQ			9.76	35.56	9.76	30.47	9.76	19.52	20.79%	37.87%	20.79%	32.45%	20.79%	20.79%
	DOR			8.97	26.35	8.97	27.85	8.97	17.94	19.11%	28.06%	19.11%	29.66%	19.11%	19.11%
	$SEID$			33.45	74.76	33.45	71.67	33.45	66.91	71.26%	79.63%	71.26%	76.34%	71.26%	71.26%
Ola	CG^L	65.56	131.12	7.66	15.33	7.66	15.33	7.66	15.33	11.69%	11.69%	11.69%	11.69%	11.69%	11.69%
	CG^U			7.66	15.33	7.66	15.33	7.66	15.33	11.69%	11.69%	11.69%	11.69%	11.69%	11.69%
	DAQ			3.16	6.32	3.16	6.32	3.16	6.32	4.82%	4.82%	4.82%	4.82%	4.82%	4.82%
	DOR			8.62	17.24	8.62	17.24	8.62	17.24	13.15%	13.15%	13.15%	13.15%	13.15%	13.15%
	$SEID$			61.90	123.80	61.90	123.80	61.90	123.80	94.41%	94.41%	94.41%	94.41%	94.41%	94.41%
Ontodna	CG^L	5.91	11.83	1.81	4.48	1.81	3.97	1.81	3.61	30.54%	37.86%	30.54%	33.57%	30.54%	30.54%
	CG^U			1.81	4.48	1.81	3.97	1.81	3.61	30.54%	37.86%	30.54%	33.57%	30.54%	30.54%
	DAQ			3.14	7.75	3.14	6.95	3.14	6.28	53.08%	65.53%	53.08%	58.77%	53.08%	53.08%
	DOR			2.27	5.37	2.27	5.06	2.27	4.54	38.35%	45.37%	38.35%	42.75%	38.35%	38.35%
	$SEID$			4.22	9.87	4.22	8.57	4.22	8.44	71.34%	83.37%	71.34%	72.43%	71.34%	71.34%

TABLE 6
Average candidate alignment sizes with and without modularization (excluding alignments of size 0)

than 24%, and, as shown in the more detailed breakdown in Table 6 when $SEID$ is used over the results of the Ola system, it can be as small as 5 %.

These possibilities suggest a flexible architecture able to use more than one modularization technique, where a lower than expected number of results to a query can trigger the use of a more conservative modularization technique; this would enable the system to guarantee the lowest possible loss of information, while ensuring maximum reduction of the search space in the average case.

The quality of the resulting alignments in terms of the retention measures is quite high: only a handful of cases shows retention lower than 95%, and a great majority shows retention equal to 1, therefore granting that the choice of modularization technique and information loss solution can safely be done on the basis of the size of the resulting alignment.

Moreover, there is a large number of cases in which the use of modularization yields an empty alignment, as shown in Table **??**. Experimental evaluation shows that in the largest majority of these cases (99.85%) an empty alignment is correlated

Average original alignment size (B): 24.84						
		DAQ	DOR	CG^U	CG^L	$SEID$
Accepted alignment size	C	3.26	2.97	1.89	1.89	12.53
	SO	3.78	2.94	2.57	2.57	13.47
	N	2.43	2.08	1.32	1.32	11.92
Accepted alignment size (%)	C	16.00%	13.10%	8.88%	8.88%	61.85%
	SO	18.39%	13.39%	11.64%	11.64%	67.70%
	N	13.28%	10.25%	6.94%	6.94%	59.91%
Average candidate alignment size: 28.95						
Average # of arguments: 57.90						
		DAQ	DOR	CG^U	CG^L	$SEID$
Avg candidates with mod.	C	2.46	2.15	1.35	1.35	12.41
	SO	2.46	2.15	1.35	1.35	12.41
	N	2.46	2.15	1.35	1.35	12.41
Avg # args with mod.	C	7.62	6.01	5.22	5.22	27.93
	SO	6.57	6.08	3.85	3.85	26.04
	N	4.92	4.30	2.70	2.70	24.83
Avg candidates with mod. (%)	C	13.34%	10.27%	6.97%	6.97%	58.45%
	SO	13.34%	10.27%	6.97%	6.97%	58.45%
	N	13.34%	10.27%	6.97%	6.97%	58.45%
Avg # args with mod. (%)	C	18.49%	13.66%	11.98%	11.98%	67.22%
	SO	16.04%	13.26%	8.94%	8.94%	60.56%
	N	13.34%	10.27%	6.97%	6.97%	58.45%

TABLE 9
Average accepted alignment sizes (averaged by modularization technique)

Average original alignment size (B): 24.84						
		DAQ	DOR	CG^U	CG^L	$SEID$
Accepted alignment size	C	6.22	6.42	3.91	3.91	18.96
	SO	7.20	6.46	4.87	4.87	19.92
	N	4.72	5.01	3.09	3.09	18.36
Accepted alignment size (%)	C	36.31%	34.87%	21.73%	21.73%	79.59%
	SO	41.61%	36.00%	26.02%	26.02%	85.51%
	N	30.11%	29.23%	18.26%	18.26%	77.64%
Average candidate alignment size: 28.95						
Average # of arguments: 57.90						
		DAQ	DOR	CG^U	CG^L	$SEID$
Avg candidates with mod.	C	4.98	5.80	3.65	3.65	23.70
	SO	14.92	14.50	10.96	10.96	50.51
	N	4.98	5.80	3.65	3.65	23.70
Avg # args with mod.	C	12.96	14.42	9.01	9.01	48.61
	SO	4.98	5.80	3.65	3.65	23.70
	N	9.95	11.59	7.30	7.30	47.40
Avg candidates with mod. (%)	C	29.16%	27.99%	16.78%	16.78%	76.15%
	SO	29.16%	27.99%	16.78%	16.78%	76.15%
	N	29.16%	27.99%	16.78%	16.78%	76.15%
Avg # args with mod. (%)	C	40.62%	35.21%	25.26%	25.26%	85.00%
	SO	35.22%	33.79%	20.35%	20.35%	78.26%
	N	29.16%	27.99%	16.78%	16.78%	76.15%

TABLE 10
Average accepted alignment sizes (averaged by modularization technique, excluding alignments of size 0)

with no answers available for the concepts in the signature, even with the complete original alignment. This confirms that there are cases in which the argumentation process can be skipped altogether without hampering the reliability of the system in terms of retention; it is in fact always higher than 99% on average.

Modules: table and discussion

Detailed presentation of the tables

?? missed points in the discussion?

8 Conclusions

NEEDS REWRITING

Agents need to reconcile ontological differences, especially within the context of open and dynamic environments where no *a priori* assumptions about the nature of the ontology can be made. Negotiation frameworks (such as the *Meaning-based argumentation*), allow agents to negotiate over different ontology correspondences, and identify those alignments that are mutually acceptable. However, this collaborative search is computationally costly, as the complexity of the decision problems reach $\Pi_2^{(p)}$-complete. In this paper we have proposed the use of *Ontology Modularization* as a mechanism to reduce the size of the search space for finding acceptable alignments. The use of ontology modularization as a filter-based pre-processing stage was evaluated empirically, by considering three approaches (CG^L, CG^U and D) over eleven ontologies used in the OEAI initiative. The results show that the use of modularization can significantly reduce the average number of correspondences presented to the argumentation framework, and hence the size of the search space – in some cases by up to 97%, across a number of different ontology pairs. In addition, three patterns emerged: i) where no reduction in size occurred (in 4.84% of cases on average); ii) where the number of correspondences was reduced (55.14%); and iii) where modules of size zero were found (40.02%). We found that this latter case corresponded to failure scenarios; i.e. where the subsequent transaction could fail due to insufficient alignment

Technique	IR	stdev	$SubR$	stdev	$SupR$	stdev
DAQ	99.87%	0.61%	100%	0%	100%	0%
DOR	99.97%	0.26%	100%	0%	100%	0%
CG^U	99.59%	2.85%	99.73%	2.79%	99.73%	2.79%
CG^L	99.59%	2.85%	99.73%	2.79%	99.73%	2.79%
$SEID$	100%	0%	100%	0%	100%	0%
Overall average	99.80%	1.83%	99.89%	1.77%	99.89%	1.77%

TABLE 11
Instance, Subclass and Superclass Retention values

O_1	O_2	System	I	MT	IR	$SubR$	$SupR$
OpenConf	paperdyne	Ontodna	N	CG^L CG^U	50.00%	50.00%	50.00%
OpenConf	Conference	Ontodna	N	CG^L CG^U	66.67%	66.67%	66.67%
OpenConf	cmt	Lily	N	CG^L CG^U	71.43%	71.43%	71.43%
OpenConf	crs_dr	Lily	N	CG^L CG^U	78.57%	78.57%	78.57%
OpenConf	paperdyne	Lily	N	CG^L CG^U	82.00%	82.50%	82.50%
OpenConf	edas	Lily	N	CG^L CG^U	82.05%	82.05%	82.05%
OpenConf	confOf	Lily	N	CG^L CG^U	87.50%	87.50%	87.50%
OpenConf	paperdyne	Falcon	N	CG^L CG^U	87.50%	87.50%	87.50%

TABLE 12
A snapshot of the lower retention values

	accepted / %	candidate / %	arguments / %
DAQ	3.16 / 12.73%	2.46 / 8.50%	6.37 / 11.00%
DOR	2.67 / 10.75%	2.15 / 7.43%	5.47 / 9.44%
CG^U	1.93 / 7.77%	1.35 / 4.66%	3.92 / 6.77%
CG^L	1.93 / 7.77%	1.35 / 4.66%	3.92 / 6.77%
$SEID$	12.65 / 50.89%	12.41 / 42.87%	26.26 / 45.36%
C	4.51 / 21.74%	3.95 / 19.20%	10.40 / 24.67%
SO	5.07 / 24.55%	3.95 / 19.20%	9.28 / 21.55%
N	3.82 / 19.47%	3.95 / 19.20%	7.89 / 19.20%

TABLE 13
Average over all runs for each modularization technique (upper half) and for each information loss solution (lower half)

	accepted / %	candidate / %	arguments / %
DAQ	6.05 / 24.34%	4.98 / 17.19%	12.61 / 21.78%
DOR	5.96 / 24.00%	5.80 / 20.02%	13.50 / 23.32%
CG^U	3.96 / 15.92%	3.65 / 12.60%	9.09 / 15.70%
CG^L	3.96 / 15.92%	3.65 / 12.60%	9.09 / 15.70%
$SEID$	19.08 / 76.78%	23.70 / 81.85%	48.84 / 84.34%
C	7.89 / 38.84%	8.35 / 33.37%	20.37 / 42.27%
SO	8.66 / 43.03%	8.35 / 33.37%	18.80 / 37.59%
N	6.85 / 34.70%	8.35 / 33.37%	16.71 / 33.37%

TABLE 14
Average over all runs for each modularization technique (upper half) and for each information loss solution (lower half). Alignments of size zero are not included in the average.

MT	System	Size 0	Size $\neq 0$	Total	Size 0 %
DAQ	Asmov	1446	737	2183	66.24%
	Falcon	1725	904	2629	65.61%
	Lily	1034	1581	2615	39.54%
	Ola	4924	336	5260	93.61%
	Ontodna	3838	1276	5114	75.05%
	Average				68.01%
DOR	Asmov	1374	809	2183	62.94%
	Falcon	1624	1005	2629	61.77%
	Lily	977	1638	2615	37.36%
	Ola	4924	336	5260	93.61%
	Ontodna	3902	1212	5114	76.30%
	Average				66.40%
CG^U	Asmov	1235	948	2183	56.57%
	Falcon	1370	1259	2629	52.11%
	Lily	713	1902	2615	27.27%
	Ola	5041	219	5260	95.84%
	Ontodna	3548	1566	5114	69.38%
	Average				60.23%
CG^L	Asmov	1235	948	2183	56.57%
	Falcon	1370	1259	2629	52.11%
	Lily	713	1902	2615	27.27%
	Ola	5041	219	5260	95.84%
	Ontodna	3548	1566	5114	69.38%
	Average				60.23%
$SEID$	Asmov	0	2183	2183	0.00%
	Falcon	0	2629	2629	0.00%
	Lily	0	2615	2615	0.00%
	Ola	4880	380	5260	92.78%
	Ontodna	208	4906	5114	4.07%
	Average				19.37%
Over all techniques					
	Asmov	5290	5625	10915	48.47%
	Falcon	6089	7056	13145	46.32%
	Lily	3437	9638	13075	26.29%
	Ola	24810	1490	26300	94.33%
	Ontodna	15044	10526	25570	58.83%
	Overall average				54.85%

TABLE 15
Percentage of empty alignments by modularization technique and alignment system

Concept	AFLO	O_2	I.	SbC.	SpC.
$I = C$, Modularization Techniques = $\{CG^U\ CG^L\}$					
http://confOf#Scholar	x	cmt, crs_dr, edas, paperdyne	1	1	2
$I = N$, Modularization Techniques = $\{CG^U\ CG^L\}$					
http://cmt#ExternalReviewer	x x	edas, paperdyne, PCS	1	1	2
http://confOf#Administrator	x x	cmt, edas, paperdyne, PCS	1	1	3
http://confOf#Assistant	x	edas	1	1	2
http://confOf#Camera_Ready_event	x x	cmt, edas, ekaw, paperdyne, PCS	1	1	2
http://confOf#Event	x x x x	cmt, crs_dr, edas, ekaw, paperdyne	9	9	1
http://confOf#Member	x x	cmt, edas, ekaw, OpenConf	3	3	3
http://confOf#Participant	x x x	crs_dr, edas, ekaw	3	3	2
http://confOf#Person	x x x	cmt, crs_dr, edas, ekaw, MICRO, paperdyne, PCS, sigkdd	10	10	1
http://confOf#Poster	x x x	cmt, Conference, edas, ekaw, paperdyne	1	1	3
http://confOf#Registration_of_participants_event	x	cmt, edas	5	5	2
http://confOf#Reviewing_event	x x x	crs_dr, edas, ekaw, paperdyne, PCS, sigkdd	1	1	2
http://confOf#Reviewing_results_event	x	edas, paperdyne	10	10	2
http://confOf#Short_paper	x	cmt, edas, paperdyne	1	1	3
http://confOf#Student	x x	edas, ekaw, MICRO	1	1	3
http://confOf#Topic	x x x x	cmt, Conference, edas, MICRO, OpenConf, paperdyne	12	10	1
http://edas#CallForPapers	x	Conference	1	1	2
http://ekaw#Accepted_Paper	x x	cmt, Conference, edas, OpenConf, paperdyne, PCS, sigkdd	1	1	3
http://ekaw#Conference	x x x	cmt, Conference, confOf, crs_dr, edas, MICRO, OpenConf, paperdyne, PCS, sigkdd	2	2	2
http://openconf#Program_Committee	x x	sigkdd	1	1	2
http://paperdyne#Program_Comitee_Chair	x x x	cmt, edas, sigkdd	1	1	3
http://paperdyne#Reviewer	x x	cmt, edas, MICRO	1	1	3

TABLE 16
Alignments of size 0 that have 0 retention. A is for Asmov, F for Falcon, L for Lily, O for OntoDNA, I for instances, SbC for subclasses, SpC for superclasses.

between the ontologies. Thus, we demonstrate that ontology modularization not only reduces the cost of negotiating over correspondences and establishing communication, but that it can be effectively used to predict cases where negotiation will fail to identify relevant correspondences to support meaningful queries.

Acknowledgment

This work is supported by the EPSRC grant: EP/D064287/1

References

[1] T. Bench-Capon, "Value based argumentation frameworks," in *Proceedings of Non Monotonic Reasoning*, 2002, pp. 444–453.

[2] ——, "Persuasion in practical argument using value-based argumentation frameworks," *Journal of Logic and Computation*, vol. 13, no. 3, pp. 429–448, 2003.

[3] T. Berners-Lee, J. Hendler, and O. Lassila, "The semantic web," *Scientific American*, vol. 284, no. 5, 2001.

[4] B. Cuenca Grau, I. Horrocks, Y. Kazakov, and U. Sattler, "Modular reuse of ontologies: Theory and practice," *J. of Artificial Intelligence Research (JAIR)*, vol. 31, pp. 273–318, 2008. [Online]. Available: http://www.jair.org/media/2375/live-2375-3703-jair.pdf

[5] P. Doran, V. Tamma, and L. Iannone, "Ontology module extraction for ontology reuse: an ontology engineering perspective," in *CIKM*. ACM, 2007, pp. 61–70.

[6] C. T. dos Santos, P. Quaresma, and R. Vieira, "Conjunctive queries for ontology based agent communication in mas," in *7th International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2008), Estoril, Portugal, May 12-16, 2008, Volume 2*, 2008, pp. 829–836.

[7] P. Dung, "On the Acceptability of Arguments and its Fundamental Role in Nonmonotonic Reasoning, Logic Programming and n-person Games," in *Artificial Intelligence*, vol. 77, no. 2, 1995, pp. 321–358.

[8] P. Dunne and T. Bench-Capon, "Complexity in value-based argument systems," in *JELIA*, ser. Lecture Notes in Computer Science, vol. 3229. Springer, 2004, pp. 360–371.

[9] J. Euzenat and P. Shvaiko, *Ontology Matching*. Springer-Verlag, 2007.

[10] J. Euzenat and P. Valtchev, "Similarity-based ontology alignment in OWL-Lite," in *European Conference on Artificial Intelligence ECAI-04*, 2004. [Online]. Available: http://www.iro.umontreal.ca/\~{}owlola/pdf/align-ECAI04-FSub.pdf

[11] E. Jiménez-Ruiz, B. Cuenca Grau, U. Sattler, T. Schneider, and R. B. Llavori, "Safe and economic re-use of ontologies: A logic-based methodology and tool support," in *The Semantic Web: Research and Applications, 5th European Semantic Web Conference, (ESWC 2008)*, 2008, pp. 185–199.

[12] L. Laera, I. Blacoe, V. Tamma, T. Payne, J. Euzenat, and T. Bench-Capon, "Argumentation over ontology correspondences in mas," in *AAMAS*, 2007, p. 228.

[13] C. Lutz, D. Walther, and F. Wolter, "Conservative extensions in expressive description logics," in *IJCAI*, 2007, pp. 453–458.

[14] N. F. Noy and M. Musen, "Specifying ontology views by traversal." in *Int. Semantic Web Conference*, 2004.

[15] P. Patel-Schneider, P. Hayes, and I. Horrocks, "OWL Web Ontology Language Semantics and Abstract Syntax," 2004. [Online]. Available: http://www.w3.org/TR/owl-semantics/

[16] J. Seidenberg and A. Rector, "Web ontology segmentation: analysis, classification and use," in *WWW '06: Proceedings of the 15th Int. Conference on World Wide Web*, 2006.

[17] J. van Diggelen, R.-J. Beun, F. Dignum, R. van Eijk, and J.-J. Meyer, "Ontology negotiation in heterogeneous multi-agent systems: The anemone system," *Applied Ontology*, vol. 2, no. 3-4, pp. 267–303, 2007.