

Dynamic Selection of Ontological Alignments: A Space Reduction Mechanism*

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

Department of Computer Science, University of Liverpool, UK

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' own ontologies may be determined dynamically through argumentation-based negotiation (such as *Meaning-based Argumentation*). However, the complexity of this process is high, approaching $\Pi_2^{(p)}$ -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 concepts are negotiated. The suitability of different modularization approaches as filtering mechanisms for reducing the negotiation search space is investigated, and a framework that integrates modularization with Meaning-based Argumentation is proposed. We empirically demonstrate that some modularization approaches not only reduce the number of alignments required to reach consensus, but also predict those cases where a service provider is unable to satisfy a request, without the need for negotiation.

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 enable 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 [Berners-Lee *et al.*, 2001].

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 scale 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 set of ontology correspondences (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 (where *alignments* are sets of correspondences) [Euzenat and Shvaiko, 2007] 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 correspondences can be generated offline and stored for later use within publicly available repositories, such as the *Ontology Alignment Server* (OAS) [Euzenat and Valtchev, 2004].

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 [van Diggelen *et al.*, 2007; Laera *et al.*, 2007; dos Santos *et al.*, 2008]. 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 Modulariza-*

*Work funded by EPSRC project EP/D064287/1

tion 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 different techniques that have been 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 the number of negotiated alignments can be reduced by an average of 53.1% (i.e. through eliminating unnecessary alignments for a given transaction). Whilst this reduction is 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: a review of negotiation techniques for resolving ontological heterogeneity between transacting agents is given in Section 2, before presenting our framework which integrates modularization within an argumentation-based negotiation approach in Section 3. The evaluation of applying ontology modularization to argumentation over ontology correspondences is detailed in Section 4, and the paper concludes in Section 5.

2 Negotiating Correspondences

A number of solutions have been proposed that attempt to resolve ontological mismatches within open environments [van Diggelen *et al.*, 2007; Laera *et al.*, 2007; dos Santos *et al.*, 2008]. van Diggelen *et al.* [2007] dynamically generate a *minimal* shared ontology, where minimality is evaluated against the ability of the different components in the environment to communicate with no information loss. The agents can explain concepts to each other via the communication mechanism; either by defining the concept in terms already understood or by invoking an extensional learning mechanism. However, the ontological model used here is limited and non-standard, as its expressivity supports only simple taxonomic structures, with no properties and few restrictions other than disjointness and partial overlap, and thus does not correspond to any of the OWL flavours¹. As such, its applicability to the augmentation of existing real-world, published, OWL ontologies on the web is limited.

The increased availability of mechanisms for ontology mapping and alignment [Euzenat and Shvaiko, 2007] has facilitated the potential construction of a plethora of different correspondence sets between two ontologies (*alignments*), depending on the approach used. As the choice of set can be highly dependent on the current task or available knowledge, it is difficult to establish which set would be mutually acceptable to two transacting agents. For example, an agent may

prefer terminological correspondences over semantic correspondences if the ontology it uses is mainly taxonomic, or vice versa if the ontology is semantically rich. Thus, the selection of mutually acceptable correspondences is essentially a collaborative search problem through the space of possible ontology correspondences (which were pre-generated, and retrieved from an alignment repository).

Various approaches that facilitate this collaborative search have been proposed. A simple method might consist of a brute force approach that selects only those mappings whose level of confidence is above a certain threshold specified for each agent. A more sophisticated approach is to exploit the use of *argumentation* as a negotiation mechanism to locate mappings that are mutually acceptable by both agents. Laera *et al.* [2007] 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)* [Bench-Capon, 2003], 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 as a common ontology language. Dos Santos *et al.* [2008] propose 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 correspondences can, however, become prohibitive when complex negotiation mechanisms such as argumentation are involved, and reach $\Pi_2^{(p)}$ -complete [Dunne and Bench-Capon, 2004]. 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 to identify only those alignments that are relevant to some transaction can greatly reduce the time required to find the relevant correspondences.

3 Ontology Modularization

The *Meaning-based argumentation* framework presented in [Laera *et al.*, 2007] provides a mechanism for collaboratively searching over the space of all possible agent correspondences to locate those that are mutually acceptable. As this search can be computationally complex (reaching $\Pi_2^{(p)}$ when deciding if an argument exists in every preferred extension), the space could be reduced by modularizing the ontology with respect to the *signature* of the concepts in the message (which in turn would reduce the number of candidate correspondences).

One possible way to reduce the search space is by limiting the concepts on which agents negotiate, and this paper proposes to adopt an *ontology modularization* [Doran *et al.*, 2007; Cuenca Grau *et al.*, 2008] process to select a subset of the concepts on which the agents negotiate. The hypothesis is that reducing the search space corresponds to a reduction in the number of correspondences that are used in the negotiation process that 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

¹The authors mention a reformulation of their model using Description Logics (the logical theory underpinning the standard ontology language OWL [Patel-Schneider *et al.*, 2004]), but provide no formal proof of its soundness [van Diggelen *et al.*, 2007].

the search, and whether all the modularization techniques behave equally when used as a filtering mechanism.

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 vocabulary². The notion of *ontology module extraction* can thus be more formally defined as:

Definition 3-1. *Ontology module extraction* extracts a consistent³ 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 $Sig(M)$ over a language $\mathcal{L}$, such as $M \subseteq \mathcal{O}$. M is an ontology itself and it is possible that by using other techniques, further modules could be extracted from it.

Ontology module extraction approaches provide a signature $Sig(M)$ and extract a subset of the ontology that contains the signature. They can be split into two distinct categories: traversal approaches and logical approaches. Traversal approaches [Doran *et al.*, 2007; Seidenberg and Rector, 2006; Noy and Musen, 2004] represent the extraction as a graph traversal, with the module being defined by the conditional traversal of a graph. Logical approaches [Cuenca Grau *et al.*, 2008] focus on maintaining the logical properties of coverage and minimality when extracting modules.

Logical approaches are based on the notion of conservative extension [Lutz *et al.*, 2007]: 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.* [Cuenca Grau *et al.*, 2008] 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.* [2007] 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.* [2008] 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.* [2008] defined two variants of locality, namely $\perp$ and $\top$, depending on whether an ontology engineer is modularizing in order to

reuse the module, or in order to generalise over 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 $\mathcal{O} = (V, E)$ where the set of vertices V is the set of concept names, and the edges in E are the relations between the concepts, such as property restrictions and subsumption relationships⁴. Ontology modules are defined as a conditional traversal on the graph $\mathcal{O}$. Seidenberg and Rector [2006] 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 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 and Rector, Doran *et al.* [2007] 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 [2004] 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.

3.1 Combining 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 correspondences. 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 $\mathcal{O}$ and $\mathcal{O}'$ respectively.

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

²This definition is agnostic with respect to the language used to represent the ontology, but the modularization techniques in this paper assume a description logic representation.

³OWL is monotonic and hence guarantees consistency if the extraction is done on a consistent ontology.

⁴Graph traversal extraction methods are possible since OWL ontologies map to RDF graphs (see <http://www.w3.org/TR/owl-semantics/>)

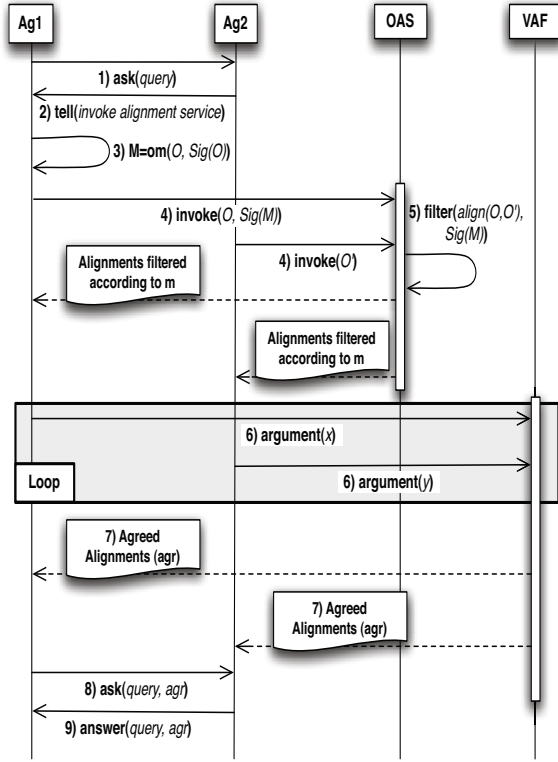


Figure 1: UML Sequence Diagram of Ontology Modularization and Argumentation.

Definition 3-2. A *filtering function*, $filter()$, filters the set of candidate correspondences 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.

1. Ag_1 asks a query, $query(A \in Sig(\mathcal{O}))$, to Ag_2 .
2. Ag_2 does not understand the query, $A \notin Sig(\mathcal{O}')$, and informs Ag_1 they need to use an Ontology Alignment Service (OAS)
3. Ag_1 produces, $om(\mathcal{O}, Sig(A))$, 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, $\mathcal{O}$ 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, $query(A \in Sig(\mathcal{O}) \wedge B \in Sig(\mathcal{O}'))$ 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

4 Evaluation

The aim of the evaluation is to contrast different modularization techniques for use as a search space reduction mechanism prior to argumentation. By identifying modules based on the signature of a query, those ontological definitions not relevant to the signature can be removed; thus eliminating the possibility of needlessly negotiating over irrelevant correspondences. Our hypothesis is that a reduction in the size of search space corresponds to a reduction in the number of correspondences that are considered by the argumentation process. To evaluate this, we analyze 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 search space. Three ontology modularization techniques have been evaluated: the upper and lower variants of Cuenca Grau *et al.* [2008] and Doran *et al.* [2007]. The technique proposed by Seidenberg and Rector [2006] was initially considered, but the results were found to be inconclusive, and thus not included within this analysis. Likewise, Noy and Musen [2004]’s technique was not considered as it is user-led.

4.1 Evaluation Setup

The eleven ontologies used in the evaluation were taken from the *OAEI 2007 Conference Track* repository (with the exception of three ontologies⁵), as they represent a large number of real world ontologies covering the same domain. This allows for the discovery of more pairwise alignments when compared to ontologies in other tracks⁶. The ontologies are listed in Table 2, complete with a brief characterization in terms of the number of classes and properties, and the level of DL expressivity used to represent them.

Ontology Name	# Cl.	# Prop.	DL expressivity	% Reduction in size	
				D	$CG^L (CG^U)$
Cmt	31	59	$\mathcal{ALCIF}(\mathcal{D})$	61.00	93.24
ConfTool	40	36	$SIF(\mathcal{D})$	76.00	91.50
Crs	16	17	$SHIN$	71.09	93.75
Edas	105	50	$\mathcal{ALCIF}(\mathcal{D})$	73.80	96.25
Ekaw	75	33	$SHIN(\mathcal{D})$	67.89	96.41
Sofsem	61	64	$\mathcal{ALCHIF}(\mathcal{D})$	86.97	91.29
Micro	33	26	$\mathcal{ALCIOF}(\mathcal{D})$	90.73	85.03
Pcs	25	38	$\mathcal{ELUIF}(\mathcal{D})$	78.08	92.00
OpenConf	64	45	$\mathcal{ALCIO}(\mathcal{D})$	64.28	64.94
Paperdyne	47	78	$\mathcal{ALCHIOF}(\mathcal{D})$	0.00	67.50
Sigkdd	51	28	$\mathcal{ELI}(\mathcal{D})$	90.08	97.27

Table 2: Classes, properties, expressivity, and average % reduction in module size for the modularization techniques by Doran *et al.* (D), and Cuenca Grau *et al.*’s upper (CG^U) and lower (CG^L) variants over the different OAEI ontologies.

Each experiment consists of identifying a module for a named class in one (source) ontology, and then identifying the correspondences for the elements in that module to a destination ontology. This is repeated for each of the ontologies (excluding the source ontology). As the correspondence between a named class A and its pairing A' in another ontology may be named depending on the pair order, both pairs are

⁵These ontologies have memory requirements of $> 1.5GB$.

⁶<http://oaei.ontologymatching.org/2007/conference/>

considered, i.e. (A, A') and (A', A) . This process is repeated for each of the named classes in the ontologies, resulting in $2 \times (11^2 - 11) = 220$ tests for each named concept.

To explore the effectiveness of modularization with different alignment techniques, two sets of alignments were used; *Alignment-API* and *Falcon-AO*. The *Alignment-API* set was generated using Euzenat’s Alignment API⁷, which generates different correspondences based on textual similarities, and similarities based on the structure of the semantic models. This approach represents a combination of various alignment techniques. The second set of alignments was taken from the Falcon-AO system⁸, which was found to produce the best alignments in the *OAEI 2007 Conference Track* competition [Euzenat *et al.*, 2007]. This set is included to provide an approximately optimal (or *gold standard*) set of alignments between the ontologies.

The argumentation procedure was executed for modules generated for each named class from each ontology, for each of the three modularization techniques. The number of correspondences that were argued and subsequently either accepted and rejected were recorded for each test. The result of the argumentation process between the ontologies when no modularization occurred was used as a baseline result for each pair.

4.2 Results

The average reduction in the number of classes due to the different modularization processes for each ontology is given in Table 2. Whilst there were minor differences in the modules generated by CG^U and CG^L , the average module sizes (and subsequent results) for both methods were found to be the same for all the ontologies tested. This could be explained by the fact that whilst both methods are similar, they only differ in behavior for certain boundary cases. When compared to Doran *et al.*’s approach, however, for all but one ontology (*Micro*), both methods produced smaller modules. The Paperdyne ontology was not affected by Doran *et al.*’s approach as it has a very shallow hierarchy with respect to the number of concepts, and has numerous properties which result in a highly interconnected ontology that is not amenable to modularization techniques based on graph traversal.

The results indicate that ontology modularization has a considerable impact on the number of correspondences that are argued. Figure 2 shows a scatter plot representing all the test cases, where each point is plotted against the total number of correspondences argued without using any modularization (x -axis), and the number of correspondences argued when using Doran *et al.*’s technique. Three broad categories emerge from this plot: i) those cases where modularization yields a significant reduction in the number of correspondences argued (which appear in the lower sector of the plot); ii) those cases where no reduction occurs (i.e. points which lie on the $x = y$ axis); and iii) those cases where no correspondences are identified, and thus no argumentation occurs (i.e. points that lie on the $y = 0$). This third category was unexpected, and corresponds to those scenarios whereby no suitable cor-

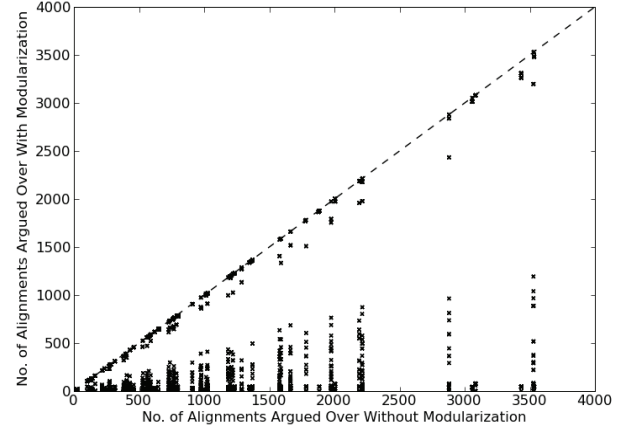


Figure 2: A scatter plot showing the total number of correspondences argued over without modularization and with Doran *et al.*’s technique.

respondences exist for the alignment of the original signature, and thus any argumentation would be redundant. Similar behavior was also observed for both Cuenca Grau *et al.*’s variants (CG^U and CG^L). Table 3 shows the percentage breakdown of these three categories between the two alignment sets: *Alignment-API* and *Falcon-AO*. These results suggest that for the *Falcon-AO* alignments (i.e., the set of correspondences that are more accurate and reliable), there is a greater number of cases identified where argumentation is redundant. Conversely the number of cases where modularization has no effect is greater for the *Alignment-API* alignments.

	Alignment-API		Falcon-AO	
	D	$CG^L (CG^U)$	D	$CG^L (CG^U)$
Reduction in args	52.1%	66.5%	42.0%	51.9%
No Reduction	22.2%	3.0%	0.7%	0.1%
No resulting args	25.7%	30.6%	57.3%	48.0%

Table 3: Percentage breakdown of three different point types

For both alignment sets, both Cuenca Grau *et al.*’s variants (CG^U and CG^L) behave similarly and have the greatest percentage of cases where there is an effect on the number of points used. However, D identifies the greatest percentage of cases for both alignment sets where there is possibly no need for argumentation. As shown in Table 2, both Cuenca Grau *et al.*’s variants produced smaller modules than that when Doran *et al.*’s approach is used, and also result in a smaller set of agreed correspondences. This result is not surprising, and could be explained by the fact that Cuenca Grau *et al.*’s variants try to include the minimal set of axioms that safely define the signature of the query. However, this could lead to child concepts of those in the original query signature being omitted, which contrasts with Doran *et al.*’s approach, which retains the child concepts of the original query. Whilst they all ensure that the query is satisfied, module minimality may not always be desirable. For example, when considering query-answering scenarios for information gathering tasks,

⁷<http://alignapi.gforge.inria.fr/>

⁸<http://iws.seu.edu.cn/projects/matching/>

then it may be desirable to include the results of more specific queries, that would occur from the inclusion of child concepts than just those of the query. However, this hypothesis is difficult to prove as there are no knowledge bases (containing instances) associated with the ontologies used by this study, and synthetically generating instances could potentially bias any associated results (as discussed in [Doran *et al.*, 2009]).

5 Conclusions

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 6.5% of cases on average); ii) where the number of correspondences was reduced (53.1%); and iii) where alignments of size zero were found (40.4%). We found that this latter case corresponded to failure scenarios; i.e. where the subsequent transaction could fail due to insufficient alignment 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.

References

- [Bench-Capon, 2003] T.J.M. Bench-Capon. Persuasion in practical argument using value-based argumentation frameworks. *Journal of Logic and Computation*, 13(3):429–448, 2003.
- [Berners-Lee *et al.*, 2001] T. Berners-Lee, J. Hendler, and O. Lassila. The semantic web. *Scientific American*, 284(5), 2001.
- [Cuenca Grau *et al.*, 2008] B. Cuenca Grau, I. Horrocks, Y. Kazakov, and U. Sattler. Modular reuse of ontologies: Theory and practice. *J. of Artificial Intelligence Research (JAIR)*, 31:273–318, 2008.
- [Doran *et al.*, 2007] P. Doran, V. Tamma, and L. Iannone. Ontology module extraction for ontology reuse: an ontology engineering perspective. In *CIKM*, pages 61–70. ACM, 2007.
- [Doran *et al.*, 2009] Paul Doran, Valentina Tamma, Terry R. Payne, and Ignazio Palmisano. Evaluating a space reduction mechanism for the dynamic selection of ontological alignments. Ulcs-09-010, University of Liverpool, 2009.
- [dos Santos *et al.*, 2008] C. Trojahn dos Santos, P. Quaresma, and R. Vieira. Conjunctive queries for ontology based agent communication in mas. In *7th Int. Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 2008)*, Estoril, Portugal, May 12-16, 2008, Volume 2, pages 829–836, 2008.
- [Dunne and Bench-Capon, 2004] P.E. Dunne and T.J.M. Bench-Capon. Complexity in value-based argument systems. In *JELIA*, volume 3229 of *Lecture Notes in Computer Science*, pages 360–371. Springer, 2004.
- [Euzenat and Shvaiko, 2007] J. Euzenat and P. Shvaiko. *Ontology Matching*. Springer-Verlag, 2007.
- [Euzenat and Valtchev, 2004] J. Euzenat and P. Valtchev. Similarity-based ontology alignment in OWL-Lite. In *European Conference on Artificial Intelligence ECAI-04*, 2004.
- [Euzenat *et al.*, 2007] J. Euzenat, A. Isaac, C. Meilicke, P. Shvaiko, H. Stuckenschmidt, O. Šváb, V. Svátek, W.R. van Hage, and M. Yatskevich. Results of the ontology alignment evaluation initiative 2007. In *Proceedings of the Ontology Matching Workshop at ISWC’07+ASWC’07*, 2007.
- [Jiménez-Ruiz *et al.*, 2008] E. Jiménez-Ruiz, B. Cuenca Grau, U. Sattler, T. Schneider, and R. Berlanga 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)*, pages 185–199, 2008.
- [Laera *et al.*, 2007] L. Laera, I. Blacoe, V. Tamma, T.R. Payne, J. Euzenat, and T.J.M. Bench-Capon. Argumentation over ontology correspondences in mas. In *AAMAS*, page 228, 2007.
- [Lutz *et al.*, 2007] C. Lutz, D. Walther, and F. Wolter. Conservative extensions in expressive description logics. In *IJCAI*, pages 453–458, 2007.
- [Noy and Musen, 2004] N. Fridman Noy and M.A. Musen. Specifying ontology views by traversal. In *Int. Semantic Web Conference*, 2004.
- [Patel-Schneider *et al.*, 2004] P.F. Patel-Schneider, P. Hayes, and I. Horrocks. OWL Web Ontology Language Semantics and Abstract Syntax, 2004.
- [Seidenberg and Rector, 2006] 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.
- [van Diggelen *et al.*, 2007] J. van Diggelen, R-J. Beun, F. Dignum, R.M. van Eijk, and J-J. Meyer. Ontology negotiation in heterogeneous multi-agent systems: The anemone system. *Applied Ontology*, 2(3-4):267–303, 2007.