Constructing $E - \mathcal{SHIQ}$ Distributed Knowledge Bases via Ontology Modularization: The mONTul method

Georgios SANTIPANTAKIS [a] George VOUROS [b]
[a] *University of the Aegean, Greece*
[b] *University of Piraeus, Greece*

Abstract. This article presents a reconfigurable method for the modularization of $\mathcal{SHIQ}$ ontologies, towards the construction of distributed $E - \mathcal{SHIQ}$ knowledge bases. The aim is to compute decompositions for correct, complete and efficient distributed reasoning. The proposed method combines graph-based modularization techniques with locality-based rules using a generic constraint problem solving framework. The paper presents experimental results concerning the modularization task, w.r.t. specific requirements for efficient distributed reasoning.

Keywords. ontology modularization, locality, constraint satisfaction problem, correct and complete reasoning

Introduction

Modularization denotes either the extraction of modules from large ontologies (e.g. for reusability purposes) or the partitioning of ontologies into modules (e.g. for facilitating evolution or reasoning tasks). In both cases, modules should not represent arbitrary chunks of knowledge. Each module must represent aspects from a specific sub-domain of the original ontology. Considering the *whole decomposition* of an ontology, all and only those entailments of the original ontology must be entailed by reasoning over *the collection of modules* extracted. It follows that the modular ontology must be in a decidable fragment of a representation language.

The aim in this paper is to propose a method for decomposing any ontology in the $\mathcal{SHIQ}$ fragment of Description Logics (DL) into an arbitrary number of modules that form a distributed $E-\mathcal{SHIQ}$ knowledge base, enabling correct and complete reasoning. Each module must preserve to a great extent the meaning of the terms in its signature. The $E - \mathcal{SHIQ}$ representation framework allows modules to be semantically associated, thus reducing replication of axioms between them. It follows that a distributed knowledge base is a network of associated modules and the meaning of terms in the decomposition is preserved by considering the coupling of modules. The decomposition can facilitate efficient reasoning using the sound and complete $E - \mathcal{SHIQ}$ distributed reasoner.

Targeting efficient distributed reasoning tasks, as shown by the experiments performed using the $E - \mathcal{SHIQ}$ reasoner [9], additional properties that must be

preserved by the networks of associated modules include: (a) Axioms must be distributed between modules in an as much as possible even way, (b) networks of modules must have a small diameter, and (c) there must not be many (and large) cycles of associated modules.

Existing modularization methods [6] are based on logical foundations, or may apply methods manipulating graphs that resemble the structure of the original ontology. While logic-based methods preserve certain logical properties (e.g. [1],[2], [3], [4], [5]), graph-based approaches apply heuristics to partition (e.g. [7], [8]) an ontology into modules. These methods usually utilize network metrics and empirical "distance measures" to define the module boundaries. The proposed modularization method combines graph-based modularization techniques with locality-based rules using a generic constraint problem solving framework. Ontology specifications are used for building a graph of dependencies among the ontology terms. These dependencies correspond to specific constraints concerning the assignment of terms in modules. The satisfaction of these constraints secure the construction of a well-formed $E-\mathcal{SHIQ}$ distributed knowledge base and specify conditions for locality-based modularization.

The specific contributions in this paper are the following: (a) It proposes the mONTul modularization method that combines rules for constructing well-formed $E-\mathcal{SHIQ}$ knowledge bases with rules for locality-based modularization in a constraint satisfaction problem. (b) The mONTul method proposed is reconfigurable and generic in several aspects: (i) The set of constraints may change w.r.t. the expressiveness of the original ontology and the modularization requirements, (ii) constraints may be considered to be soft or hard, in any arbitrary combination, (iii) mONTul can be used for extracting modules, although the focus here is on computing partitionings; and finally, (iv) the method itself has been developed in a modular architecture, allowing components to be replaced by state-of-the-art methods. (c) It revises the notion of locality by also considering possible associations between terms of different modules in a distributed $E-\mathcal{SHIQ}$ knowledge base.

In the next sections, Section 1 briefly presents the background knowledge and Section 2 revises the notion of locality for $E-\mathcal{SHIQ}$. Section 3 presents the mONTul method, Section 4 the experimental results and Section 5 concludes the paper.

1. Background Knowledge

This section presents background knowledge on the $E-\mathcal{SHIQ}$ representation framework [9] and on locality-based modularization [1]. To provide concrete examples, we consider the ontology $\mathcal{O}$ with the axioms:

$$Conference \equiv MedicalConference \sqcup OtherConference \quad (1)$$
$$Conference \sqsubseteq Event \quad (2)$$
$$Event \sqsubseteq HumanActivity \quad (3)$$
$$PediatricConference \sqsubseteq MedicalConference \quad (4)$$
$$Article \sqsubseteq PublishedMaterial \sqcap \exists PresentedAt.Event \quad (5)$$
$$MedicalArticle \sqsubseteq Article \sqcap \forall PresentedAt.MedicalConference \quad (6)$$

The E-$\mathcal{SHIQ}$ representation framework [9] belongs to the family of modular representation frameworks for Description Logics. It provides constructors

for associating ontology units (modules) that are within the $\mathcal{SHIQ}$ fragment of Description Logics, preserving decidability.

Given a non-empty set of indices I and a collection of modules indexed by I, let N_{C_i}, N_{R_i} and N_{O_i} be the sets of concept, role and individual names for unit $i \in I$, respectively. For some $R \in N_{R_i}$, $Inv(R)$ denotes the inverse role of R and $(N_{R_i} \cup \{Inv(R) | R \in N_{R_i}\})$ is the set of $\mathcal{SHIQ}$ i-roles, i.e. the roles of the i-th ontology $\mathcal{M}_i$. An i-role axiom is either a role inclusion axiom or a transitivity axiom. Let $\mathcal{R}_i$ be the set of i-role axioms.

Let $\mathcal{E}_{ij}$ be the set of ij-link relations relating individuals in $\mathcal{M}_i$ and $\mathcal{M}_j$, $i \neq j \in I$. The sets of link relations are not pairwise disjoint, but are disjoint with respect to the set of concept names. An *ij-relation box* $\mathcal{R}_{ij}$ includes a finite set of $ij-link\ relation$ inclusion axioms in case $i \neq j$, and transitivity axioms of the form $Trans(E,(i,j))$, where E is in $(\mathcal{E}_{ij} \cap N_{R_i})$, i.e. it is an ij-link relation and an i-role. In case $i = j$, then $\mathcal{R}_{ij}$=$\mathcal{R}_i$ (with an abuse of notation) includes a finite set of $i-role$ inclusion axioms. Subsequently we use the term *property* to denote both roles and link-relations.

Briefly, the sets of *i-concepts* are inductively defined by the constructors within the $\mathcal{SHIQ}$ fragment of Description Logics, where i-roles can be replaced by ij-link relations, relating instances of the i-concept to instances of a j-concept, where $i \neq j \in I$. Let $i : C$ and $i : D$ possibly complex concepts and $i : C \sqsubseteq i : D$ (or $i : C \sqsubseteq D$) a *general concept inclusion* (GCI) axiom in $\mathcal{M}_i$. A finite set of GCI's in $\mathcal{M}_i$ is a TBox for i and it is denoted by $\mathcal{T}_i$.

Concept correspondences may be concept *onto* concept, or concept *into* concept: Let $C \in N_{C_i}$, $D \in N_{C_j}$ with $i \neq j \in I$. A concept *onto* (*into*) concept correspondence from $\mathcal{M}_i$ to $\mathcal{M}_j$ that subjectively holds for $\mathcal{M}_j$, is of the form $i : C \stackrel{\sqsupseteq}{\rightarrow} j : D$ (corresp. $i : C \stackrel{\sqsubseteq}{\rightarrow} j : D$).

Definition (Distributed Knowledge Base). A *distributed knowledge base* $\Sigma = \langle \mathbf{T}, \mathbf{R}, \mathbf{C} \rangle$ is composed of the distributed TBox $\mathbf{T}$, the distributed RBox $\mathbf{R}$, and a tuple of sets of correspondences $\mathbf{C} = (\mathbf{C}_{ij})_{i \neq j \in I}$ between modules. A *distributed TBox* is a tuple of TBoxes $\mathbf{T}$= $(\mathcal{T}_i)_{i \in I}$, where each $\mathcal{T}_i$ is a finite set of i-concept inclusion axioms. A *distributed RBox* is a tuple of ij-property boxes $\mathbf{R} = (\mathcal{R}_{ij})_{i,j \in I}$, where each $\mathcal{R}_{ij}$ is a finite set of property inclusion axioms and transitivity axioms.

A *distributed ABox* (DAB) includes a tuple of ABox'es $\mathcal{A}_i$ for each ontology unit $\mathcal{M}_i$, and sets $\mathcal{A}_{ij}, i \neq j$ with individual correspondences of the form $j{:}a \stackrel{=}{\mapsto} i{:}b$, and property assertions of the form $(a,b) : E_{ij}$, where E_{ij} is an ij-link relation in $\mathcal{E}_{ij}, i \neq j$. Thus, individual correspondences are specified from the subjective point of view of $\mathcal{M}_i$ and, together with assertions concerning linked individuals, these are made locally available to $\mathcal{M}_i$.

Example: Let us for instance consider two units $\mathcal{M}_1$, $\mathcal{M}_2$ for $I = \{1,2\}$, computed from ontology $\mathcal{O}$. Then, the $E-\mathcal{SHIQ}$ distributed knowledge base is $\Sigma = \langle \mathbf{T}, \mathbf{R}, \mathbf{C} \rangle$, and is composed by the distributed TBox $\mathbf{T}$, the distributed RBox $\mathbf{R}$, and a tuple of sets of correspondences $\mathbf{C} = (\mathbf{C}_{ij})_{i \neq j \in I}$ between ontology units. Specifically,

-$\mathbf{T}$= $(\mathcal{T}_i)_{i \in I}$, where
$\mathcal{T}_1 = \{1,2,3,4\}$, (indicating the axioms included) and
$\mathcal{T}_2 = \{5,6\}$.

- $\mathcal{R} = ((R_i)_{i \in I}, (R_{ij})_{i \neq j \in I})$, where $R_i = R_{ij} = \emptyset$, $i,j \in I$,

- $\mathbf{C}=(\mathbf{C}_{ij})_{i\neq j\in I}$, where
$\mathbf{C}_{21}=\{2:MedicalConference \stackrel{\equiv}{\rightarrow} 1:MedicalConference, 2:Event \stackrel{\equiv}{\rightarrow} 1:Event\}$
$\mathbf{C}_{12}=\{1:MedicalConference \stackrel{\equiv}{\rightarrow} 2:MedicalConference, 1:Event \stackrel{\equiv}{\rightarrow} 2:Event\}$
- $DAB=((\mathcal{A}_i)_{i\in I},(\mathcal{A}_{ij})_{i\neq j\in I})$, where $\mathcal{A}_i=\emptyset$ and $\mathcal{A}_{ij}=\emptyset$, for any $i,j\in I$.

For the sake of brevity, we use equivalence subjective correspondences: These are actually specified by means of onto and into subjective correspondences. Please notice that both modules hold subjective knowledge on concepts correspondences, and these are symmetric. Finally, it must be noticed that the property $PresentedAt$ is a 2-role, only: $E-\mathcal{SHIQ}$ does not support correspondences between roles.

A distributed knowledge base forms a network of associated (via correspondences and link relations) modules. Subsequently we use the terms network (of modules), decomposition and distributed knowledge base interchangeably. Associations are directional and may form cycles in the network of modules (as also shown in the example above).

Definition (Domain relations). Domain relations $r_{ij}, i\neq j\in I$ represent equalities between individuals, from the subjective point of view of j. A *domain relation* $r_{ij}, i\neq j$ from Δ_i to Δ_j is a subset of $\Delta_i\times\Delta_j$, s.t. in case $d'\in r_{ij}(d_1)$ and $d'\in r_{ij}(d_2)$, then according to the subjective view of j, $d_1=d_2$ (denoted by $d_1=_j d_2$). Also, given a subset D of $\Delta^{\mathcal{I}_i}$, $r_{ij}(D)$ denotes $\cup_{d\in D} r_{ij}(d)$.

Given that domain relations represent equalities, in case $d_1\in r_{ij}(d)$ and $d_2\in r_{ij}(d)$, then $d_1=_j d_2$. Therefore, $E-\mathcal{SHIQ}$ domain relations are globally one-to-one relations.

Definition (Distributed Interpretation). Given the index I and $i,j\in I$, a *distributed interpretation* $\mathfrak{I}$ of a distributed knowledge base Σ is the tuple formed by the interpretations $\mathcal{I}_{ij}=\langle\Delta_i,\Delta_j,\cdot^{\mathcal{I}_{ij}}\rangle$, $i,j\in I$, and a set of domain relations r_{ij}, in case $i\neq j\in I$. Formally, $\mathfrak{I}=\langle(\mathcal{I}_{ij})_{i,j\in I},(r_{ij})_{i\neq j\in I}\rangle$.

A local interpretation $\mathcal{I}_i$ satisfies an i-concept C w.r.t. a distributed knowledge base Σ, i.e. $\mathcal{I}_i\vDash i:C$ iff $C^{\mathcal{I}_i}\neq\emptyset$. $\mathcal{I}_i$ satisfies an axiom $C\sqsubseteq D$ between i-concepts (i.e. $\mathcal{I}_i\vDash i:C\sqsubseteq D$) if $C^{\mathcal{I}_i}\subseteq D^{\mathcal{I}_i}$. Also, $\mathcal{I}_{ij}$ satisfies an *ij-property inclusion axiom* $R\sqsubseteq S$ ($\mathcal{I}_{ij}\vDash R\sqsubseteq S$) if $R^{\mathcal{I}_{ij}}\subseteq S^{\mathcal{I}_{ij}}$. A transitivity axiom $Trans(E;(i,j))$ is satisfied by $\mathfrak{I}$ iff $E^{\mathcal{I}_i}\cup E^{\mathcal{I}_{ij}}$ is transitive.

Definition (Distributed entailment and satisfiability). $\Sigma\vDash_d X\sqsubseteq Y$ if for every $\mathfrak{I}$, $\mathfrak{I}\vDash_d\Sigma$ implies $\mathfrak{I}\vDash_d X\sqsubseteq Y$, where X and Y are either i-concepts or ij-properties, $i,j\in I$. Σ is satisfiable if there exists a $\mathfrak{I}$ s.t. $\mathfrak{I}\vDash_d\Sigma$. A concept $i{:}C$ is satisfiable with respect to Σ if there is a $\mathfrak{I}$ s.t. $\mathfrak{I}\vDash_d\Sigma$ and $C^{\mathcal{I}_i}\neq\emptyset$.

The $E-\mathcal{SHIQ}$ distributed reasoner implements a sound and complete tableau algorithm [9] for combining local reasoning chunks corresponding to the individual modules in a peer-to-peer fashion, inherently supporting the propagation of subsumptions between reasoning peers. The key mechanism for distributed reasoning is the projection of nodes from the completion graph of one peer (with a module) to another peer (with an associated module). This allows reasoning peers to combine their local (subjective) knowledge with the knowledge that other peers hold, in order to *jointly* compute entailments. Through projections, peers propagate subsumptions through their connections to distant peers. Each projec-

tion request from a peer to another can trigger further projections from the later, resulting to "avalanches" of triggered projections across paths in the network of associated modules. Due to this, as also shown in [9], the distributed reasoner is affected by (a) the distribution of axioms in modules, (b) the diameter of the network of modules, and (c) the number of cycles in the network of modules.

Locality-based modularization. The locality-based modularization extracts a module for a given signature s.t. it can compute all and only the entailment of the original ontology involving the terms in the signature. Formally, a module $\mathcal{M}$ for an ontology $\mathcal{O}$ in the language L, w.r.t. a signature $\mathcal{S}$ is an ontology $\mathcal{M} \subseteq \mathcal{O}$ s.t. $\mathcal{M}$ and $\mathcal{O}$ entail the same axioms over $\mathcal{S}$ in L. As shown in [5], this notion can be formalized using the notion of *model-based conservative extensions*. In this case, every model of a module $\mathcal{M}$ of $\mathcal{O}$ can be extended to a model of $\mathcal{O}$ without changing the interpretation domain or the interpretation of symbols in $\mathcal{S}$. Thus, given an $\mathcal{O}$ and $\mathcal{S} \subseteq Sig(\mathcal{O})$, a module $\mathcal{M} \subseteq \mathcal{O}$ is defined to be a *module* of $\mathcal{O}$ for $\mathcal{S}$ if $\mathcal{O}$ is a model conservative extension of $\mathcal{M}$ for $\mathcal{S}$.

Since the problem of checking whether any "part" $\mathcal{M}$ is a module of $\mathcal{O}$ for a signature $\mathcal{S}$ is undecidable for fairly lightweight fragments of OWL [4], [5], we need to compute approximations. According to [1] and [2], a sufficient condition for a conservative model is locality:

Definition ($\emptyset$-locality). Let $\mathcal{S}$ be a signature. An interpretation is $\emptyset$-local for $\mathcal{S}$ if for every class A and property R not in $\mathcal{S}$, we have $A^{\mathcal{I}} = R^{\mathcal{I}} = \emptyset$. An axiom α is $\emptyset$-local for $\mathcal{S}$ if $\mathcal{I} \vDash \alpha$ for each $\mathcal{I}$ that is $\emptyset$-local for $\mathcal{S}$. An ontology $\mathcal{O}$ is $\emptyset$-local for $\mathcal{S}$ if every axiom in $\mathcal{O}$ is $\emptyset$-local for $\mathcal{S}$.

Example. We can easily check that axioms (5) and (6) in $\mathcal{O}\backslash\mathcal{M}_1$ are $\emptyset$-local for Sig($\mathcal{M}_1$)= {*Conference, MedicalConference, OtherConference, Event, HumanActivity, PediatricConference*}: Any interpretation $\mathcal{I}$ that interprets all symbols in Sig($\mathcal{O}$) \ Sig($\mathcal{M}_1$) as the empty set, is $\emptyset$-local for $\mathcal{S}$=Sig($\mathcal{M}_1$). However, $\mathcal{M}_1$ is not $\emptyset$-local for Sig($\mathcal{M}_2$) if we consider that Sig($\mathcal{M}_2$) includes the terms $Event$ and/or $MedicalConference$: Axiom (3) for instance can not be made $\emptyset$-local for any interpretation that interprets $HumanActivity$ as the empty set, since $Event \in$ Sig($\mathcal{M}_2$). This is the case for the axiom (1), as well, due to the concept $MedicalConference$.

Since checking $\emptyset$-locality is proved costly as well, we can use the following syntactic conditions for checking $\bot$-locality for a specific signature $\mathcal{S}$. $\bot$-locality implies $\emptyset$-locality [1], [2].

Given a signature $\mathcal{S} \subseteq Sig(\mathcal{O})$ for a $\mathcal{SHIQ}$ ontology $\mathcal{O}$, the following grammar recursively defines the *positive $\bot$-concepts*, denoted as $C^+_{\mathcal{S}}$, for $\mathcal{S}$:

$$C^+_{\mathcal{S}} ::= A^+ \mid \neg C^- \mid C \sqcap C^+ \mid \exists R^+.C \mid \exists R.C^+ \mid \geq nR^+.C \mid \geq nR.C^+ \quad (7)$$

where A^+ is a concept name and R^+ a role in $Sig(\mathcal{O})\backslash\mathcal{S}$, $C \in Sig(\mathcal{O})$ and $R \in Sig(\mathcal{O})$. It must be noticed that given that $\mathcal{S} \subseteq Sig(\mathcal{O})$, a concept or role in $Sig(\mathcal{O})$ may not be in $C^+_{\mathcal{S}}$ or in $C^-_{\mathcal{S}}$. The *negative $\bot$-concepts* $C^-_{\mathcal{S}}$ for $\mathcal{S}$ are as follows:

$$C^-_{\mathcal{S}} ::= \neg C^+ \mid C^-_1 \sqcap C^-_2 \mid C^- \sqcup C \mid \forall R^+.C \mid \forall R.C^- \mid \leq nR^+.C \mid \leq nR.C^- \quad (8)$$

The other constructs of $\mathcal{SHIQ}$ can be expressed using the above constructors, so they can be used in local concepts as well.

A role inclusion axiom $R^+ \sqsubseteq R$ or a transitivity axiom $Trans(R^+)$ is local w.r.t. $\mathcal{S}$. A GCI is local w.r.t. $\mathcal{S}$ if it is either of the form $C^+_{\mathcal{S}} \sqsubseteq C$ or of the form $C \sqsubseteq C^-_{\mathcal{S}}$.

A module $\mathcal{M}$ of an ontology is $\bot-$local for a signature $\mathcal{S}$, iff all its axioms are local for $\mathcal{S} \cup Sig(\mathcal{M})$.

Definition ($\bot$-module). Let $\mathcal{O}$ be an ontology and let $\mathcal{S}$ be a signature, $\mathcal{S} \subseteq Sig(\mathcal{O})$. $\mathcal{M} \subseteq \mathcal{O}$ is a $\bot$-module for $\mathcal{O}$ for $\mathcal{S}$, if $\mathcal{O} \backslash \mathcal{M}$ is $\bot$-local for $\mathcal{S} \cup Sig(\mathcal{M})$.

Example. We can easily check that $\mathcal{M}_2 = \mathcal{O} \backslash \mathcal{M}_1$ with axioms 5 and 6 is $\bot$-local for $\mathcal{S} \cup Sig(\mathcal{M}_1) = \{$*Conference, MedicalConference, OtherConference, Event, HumanActivity, PediatricConference*$\}$.

2. $E - \mathcal{SHIQ}$ locality-based modules

Given a distributed $E - \mathcal{SHIQ}$ knowledge base indexed by I, we have to specify the rules for deciding when an individual $E - \mathcal{SHIQ}$ module $\mathcal{M}_i$, $i \in I$, is a $\bot-$module for its signature. To do this, we need first to refine the rules for $\bot$-concepts specified in formulae (7) and (8), so as (a) to assure that i-roles are specified for i-concepts only, $i \in I$, and (b) to incorporate the cases where i-concepts are constructed by means of restrictions on link relations.

Therefore, denoting a role in an $E-\mathcal{SHIQ}$ knowledge base with R and a link relation with E, then for an $E - \mathcal{SHIQ}$ module $\mathcal{M}_i$, $i \in I$, the rules for positive and negative $\bot-$concepts for $\mathcal{S} = Sig(\mathcal{M}_i)$ are:

$$C^+_{\mathcal{S}} ::= A^+ \mid \neg C^- \mid C \sqcap C^+ \mid \exists R^+.C^+ \mid \exists E^+.C^- \mid \geq nR^+.C^+ \mid \geq nE^+.C^- \quad (9)$$

$$C^-_{\mathcal{S}} ::= \neg C^+ \mid C^-_1 \sqcap C^-_2 \mid C^- \sqcup C \mid \forall R^-.C^- \mid \forall E^+.C^- \mid \leq nR^-.C^- \mid \leq nE^+.C^- \quad (10)$$

Example. Considering our example with $\mathcal{M}_1$ including axioms $\{1, 2, 3, 4\}$ and $\mathcal{M}_2$ the axioms $\{5, 6\}$, then for $Sig(\mathcal{M}_1) = \{$*Conference, MedicalConference,OtherConference, Conference,Event, HumanActivity, PediatricConference*$\}$, the concept $\exists PresentedAt.Event$ is $\bot-$local for $Sig(\mathcal{M}_1)$, if $PresentedAt$ is a 21-link relation, according to the fifth rule for positive $\bot-$concepts.

Forming $PresentedAt$ as a 21-link relation makes the $\mathcal{M}_2$-concept $\forall PresentedAt.MedicalConference$ a negative $\bot-$concept for $Sig(\mathcal{M}_1)$. Nevertheless, in this case the module $\mathcal{M}_2$ is $\bot-$local for $Sig(\mathcal{M}_1)$, given that $Article$ is a positive $\bot-$concept for $Sig(\mathcal{M}_1)$.

The above example shows that concepts of the form $\forall R.C$ may present difficulties for maintaining the $\bot-$locality of $E - \mathcal{SHIQ}$, depending on the context of their appearance.

To completely define $\bot-$locality for $E - \mathcal{SHIQ}$, we must also consider the subjective correspondences between concept names. Considering subjective *onto* concept correspondences between the module $\mathcal{M}_i$ and any other module $\mathcal{M}_j$, $i \neq j \in I$, in a distributed $E - \mathcal{SHIQ}$ knowledge base, then similarly to GCIs for the module $\mathcal{M}_i$, the following rule holds for making an *onto* correspondence $\bot-$local for $Sig(\mathcal{M}_j)$: $\quad j : C \stackrel{\sqsupseteq}{\Rightarrow} i : C^+_{\mathcal{S}}$

At this point we have to clarify that correspondences that $\mathcal{M}_i$ subjectively holds do not affect the locality of $\mathcal{M}_j$.

In the general case, *into* correspondences of $\mathcal{M}_i$, can not be $\bot-$local for $Sig(\mathcal{M}_j)$, since they are of the form: $\quad j : C^-_{\mathcal{S}} \stackrel{\sqsubseteq}{\Rightarrow} i : C^+_{\mathcal{S}}$

Nevertheless, such a subjective correspondence can be $\bot-$local for $\mathcal{S} = Sig(\mathcal{M}_j)\backslash(Sig(\mathcal{M}_j) \cap Sig(\mathcal{M}_i))$ when the same concept name appears in both sides of the correspondence and this concept name belongs in $Sig(\mathcal{M}_i)$. Such a correspondence is of the form $j : C^+_{\mathcal{S}} \stackrel{\sqsubseteq}{\Rightarrow} i : C^+_{\mathcal{S}}$, where $\mathcal{S} = Sig(\mathcal{M}_j)\backslash Sig(\mathcal{M}_i)$.

It must be noticed that the above hold for equivalence correspondences as well, since these are conjunctions of *onto and into* subjective correspondences.

Example: Considering again our running example, let us consider that axioms (2) and (3) in $\mathcal{M}_1$ are formed as correspondences. Thus, the knowledge base becomes as follows: $\mathcal{M}_1$ includes the axioms $\{1, 4\}$, $\mathcal{M}_2$ the axioms $\{5, 6\}$, $\mathcal{S} = Sig(\mathcal{M}_1) = \{$*Conference, MedicalConference,OtherConference,HumanActivity, PediatricConference*$\}$ and $Sig(\mathcal{M}_2) = Sig(\mathcal{O}\backslash\mathcal{M}_1)$. There are also two correspondences between modules: *1:Conference*$^-_{\mathcal{S}} \stackrel{\sqsubseteq}{\Rightarrow}$ *2:Event*$^+_{\mathcal{S}}$ and *1:HumanActivity*$^-_{\mathcal{S}} \stackrel{\sqsupseteq}{\Rightarrow}$ *2:Event*$^+_{\mathcal{S}}$. Focusing on the correspondences, we can observe that the first one is not $\bot-$*local* for $\mathcal{M}_2$ (both correspondences are from the subjective point of view of $\mathcal{M}_2$), while the second correspondence is $\bot-$ *local* for $\mathcal{M}_2$. Therefore, $\mathcal{M}_2$ is not local for $Sig(\mathcal{M}_1)$.

Given that only *into* correspondences can bound the size of interpretations for concepts in module $\mathcal{M}_i$, and in order to maintain the notion of locality in these cases, we revisit the definition of $\bot$-module for $E-\mathcal{SHIQ}$, considering the *restricted* signature of such a module, denoted by $\mathcal{S}^*$:

Definition ($\bot$ − E-module). Let $\mathcal{O}$ be an ontology and let $\mathcal{S}$ be a signature, $\mathcal{S} \subseteq Sig(\mathcal{O})$. $\mathcal{M} \subseteq \mathcal{O}$ is a $\bot - E$-module for $\mathcal{O}$ for $\mathcal{S}$, if $\mathcal{O}\backslash\mathcal{M}$ is $\bot$-local for the *restricted* signature $\mathcal{S}^* = \mathcal{S} \cup Sig^*(\mathcal{M})$, where $Sig^*(\mathcal{M}) = Sig(\mathcal{M})$- $\{C|$ *C is a concept name and there exists a correspondence of the form* $C \stackrel{\sqsubseteq}{\Rightarrow} C$ *from the subjective point of view of* $\mathcal{O}\backslash\mathcal{M}\}$.

Subsequently, the module $\mathcal{O}\backslash\mathcal{M}$ is called a $\bot - E$-local module for $\mathcal{S}^* = \mathcal{S} \cup Sig^*(\mathcal{M})$. In case $Sig^*(\mathcal{M}) = Sig(\mathcal{M})$, then $\mathcal{O}\backslash\mathcal{M}$ is $\bot-$local. Thus, every $\bot-$local ($\bot-$module) is a $\bot-E$-local module (resp. $\bot-E-$module), but not vice-versa. It must be noticed that the existence of onto correspondences do not harm the completeness and correctness of the reasoning tasks, given that reasoning tasks combine the (subjective) knowledge of *all* associated modules.

The $E-\mathcal{SHIQ}$ constructors offer many options for associating different modules, either via link-relations, or via concept correspondences, or via combinations of these: These are also alternative options for the modularization method. Nevertheless, not all combinations are valid for $E - \mathcal{SHIQ}$. For instance, lack of role-to-role correspondences in $E - \mathcal{SHIQ}$, pose the restrictions that a role hierarchy is set in a single module.

3. The mONTul method

The intuition behind the proposed modularization method is to keep highly-dependent ontology terms in the same module, subject to satisfying the constraints for $\bot - E-$local modules w.r.t. their signature. Please recall that the major aim is not to provide locality-preserving modularizations per se, but to compute decompositions that are complete and correct for reasoning tasks w.r.t. locality-preserving constraints, towards enhancing the efficiency of distributed

reasoning tasks. Thus, locality constraints drive the decomposition process, reducing the several options that the method may consider for constructing distributed $E-\mathcal{SHIQ}$ knowledge bases, preserving the coherency of the subject matters that each module includes. The major steps of the method are as follows:

a) Construction of a dependency graph for specifying the dependencies between concepts and roles, according to the original theory,

b) Clustering concepts and roles into groups so as to satisfy the specified constraints (as much as possible) and to decide the signature of modules; and finally,

c) The construction of modules and of their associations.

Finally, the network of associated modules is further "tuned" so as to eliminate cycles and reduce the diameter of the network by merging small modules. The paragraphs that follow describe each major step of mONTul.

Dependency Graph. Given an ontology $\mathcal{O}$ in the $\mathcal{SHIQ}$ fragment of Description Logics, the *dependency graph* for that ontology is a directed graph $G = \langle \mathbf{E}, \mathbf{V} \rangle$, where $\mathbf{V}$ is a set of nodes corresponding to concepts (concept names and complex concepts) and roles, and a set of unidirectional *dependency edges* $\mathbf{E}$ connecting the nodes. Each node for a concept C in the graph is associated with a state variable S_C. The state S_R of a node corresponding to an ontology role R, comprises two variables $S_{R_{from}}$ and $S_{R_{to}}$, indicating the module to which R belongs and the module linked via R, respectively. All state variables range to a finite subset of natural numbers $\mathbf{D}$ and their values specify whether dependent ontology terms must appear in the signature of the same module or not.

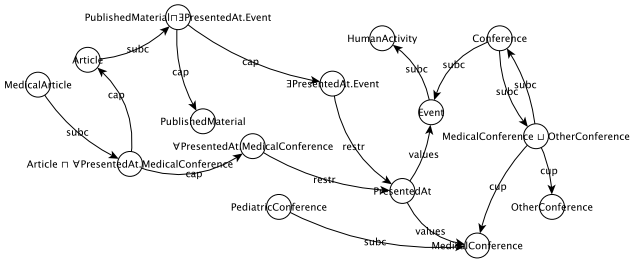

Figure 1. The dependency graph for the example ontology

The types of edges in the dependency graph are according to the constructors available in the $\mathcal{SHIQ}$ fragment of Description Logics:

[E1:] Two nodes v_C and v_D representing concepts C and D can be connected:
(a) With an edge of type $subc - dep$ if $C \sqsubseteq D$,
(b) with an edge of type $cap - dep$, if C is of the form $\sqcap_i D_i$, $i = 1, 2...$ and there is a D_k, s.t. $D = D_k$ (the equality specifies equality between the terms lexicalizing the concepts),
(c) with an edge of type $cup - dep$, if C is of the form $\sqcup_i D_i$, $i = 1, 2...$ and there is a D_k, s.t. $D = D_k$,
(d) with an edge of type $compl$, in case C and D are concept names and C is of the form $(\neg D)$, or

[E2:] A concept C is related to a role R via an edge of type $\mathrm{Q}-dep$, if C is of the form $\mathrm{Q}R.D$, where $\mathrm{Q} \in \{\forall, \exists, \leq n, \geq n\}$.

[E3:] A role R is related to a concept C via an edge $values - dep$, if there is a restriction of the form $\mathrm{Q}R.C$, where $\mathrm{Q} \in \{\forall, \exists, \leq n, \geq n\}$.

[E4:] A node v_R corresponding to a role R can be related to a node v_S corresponding to a role S as follows:
(a) with a $subr - dependency$ in case $R \sqsubseteq S$, or
(b) with an $inv - dependency$ in case R is the inverse of S.
(c) Transitivity of roles do not impose further dependencies.

Any $C \equiv D$ and $disjoint(C, D)$ axiom can be expressed using subsumption relations. Figure 1 presents the dependency graph for the example ontology.

Edges in the dependency graph are associated with constraints that specify how the states of adjacent nodes can be assigned values from **D**. These constraints can be distinguished into generic constraints (GC) and locality preserving constraints (LC). While the former suffice for the construction of a $E-\mathcal{SHIQ}$ distributed knowledge base, the later encode the rules in formulae (9) and are necessary for computing $\bot-E-$local modules $\mathcal{M}$ for the signature $Sig(\mathcal{O})\backslash Sig(\mathcal{M})$. Specifically, the constraints are the following:

[GC1:] If there is a $Q-dep$ between a node v_C and a node v_R, then it must hold that $S_C = S_{R_{from}}$.
[GC2:] If there is a $values-dep$ between a node v_R and a node v_C, then it must hold that $S_C = S_{R_{to}}$
[GC3:] If there is a $subc-dep$ between nodes v_C and v_D, then it must hold that $S_C = S_D$.
[GC4:] If there is a $subr-dep$ or $inv-dep$ between nodes v_R and v_S, then it must hold that $S_{S_{from}} = S_{R_{from}}$ and $S_{S_{to}} = S_{R_{to}}$
[LC1:] If there is a $cap-dep$ between nodes v_C and v_D, and D is a complex concept, then $S_C \neq S_D$. If D is a concept name, then it must hold that $S_C = S_D$.
[LC2:] If there is a $cup-dep$ between nodes v_C and v_D, then it must hold that $S_C = S_D$.
[LC3:] If there is an edge *Q-dep* between C and R, where $Q \in \{\exists, \geq n\}$ and there is no *Q-dep* between C' and R where $Q \in \{\forall, \leq n\}$, then it must hold that $S_{R_{from}} = S_{R_{to}}$.

It must be noticed that GC3 can be omitted. In this case we can specify *onto* correspondences between concepts in different modules resulting to modules $\mathcal{M}$ that are $\bot-E-$local for $Sig(\mathcal{O})\backslash Sig(\mathcal{M})$. Regarding the locality preserving constraints, these aim to construct positive $\bot-$concepts for the module $\mathcal{M}$ according to formulae (9), for the signature $Sig(\mathcal{O})\backslash Sig(\mathcal{M})$.

Regarding specifications of the form $QR.C$, where $Q \in \{\forall, \leq n\}$, checking the context where they appear is necessary. Given that we primarily aim at efficient reasoning, in this work we choose to sacrifice locality in cases where this is affected by this type of specification for the efficiency and simplicity of the modularization method. Therefore, to deal with specifications of the form $QR.C$, where $Q \in \{\forall, \leq n\}$ the method considers the following constraint:

[LC4:] If there is an edge $Q-dep$ between C and R, where $Q \in \{\forall, \leq n, \}$, then it may hold that $S_{R_{from}} \neq S_{R_{to}}$, even if there is a *Q-dep* between C' and R, where $Q \in \{\exists, \geq n\}$.

This constraint (in contrast to the others) aims to make the concept $QR.C$ with $Q \in \{\forall, \leq n\}$ negative $\bot-$ concepts for the signature $Sig(\mathcal{O})\backslash Sig(\mathcal{M})$.

Now, given all the constraints specified, we need to point out that not all constraints can be satisfied if they occur in a specific problem instance. Therefore we can distinguish between soft and hard constraints: Given our decomposition purpose, GC constraints (except GC3) are considered hard in any case. Configurations with GC constraints being soft are beyond the scope of this work.

Constraint Satisfaction. During the construction of the dependency graph, the axioms of the ontology are consulted and parsed to detect constituent concepts and roles. For each new concept or role, the method creates a new dependency graph node and it assigns a value to its state variable from **D**, so as to satisfy the constraints associated to it. This task ends with (probably many) violated hard constraints, which are resolved using a CSP solver, with initial values being those already assigned to the state variables during the construction of the dependency graph. Given the assignments computed by the CSP solver, a hill-climbing algorithm minimizes the number of the remaining violated constraints (if any).

Module Construction. Once the state values of the nodes are decided, dependency graph nodes are clustered into groups. Based on that, the method can decide on the signature of each module. Each group contains those concept names

and properties whose state variables (S_C for concepts and $S_{R_{from}}$ for properties) are equal, and the corresponding nodes are connected via a path in the dependency graph.

At this stage, the method considers additional desiderata concerning the reasoner's performance, i.e. the distribution of axioms, the diameter of the network and the size of cycles of associated modules. For each axiom α in the ontology, the module $\mathcal{M}$ that will host this axiom is the one that maximizes the function: $U(\mathcal{M}, \alpha) = \frac{|Sig(\mathcal{M})|+|Sig(\mathcal{M})\cap Sig(\alpha)|}{|T(\mathcal{M})|+1+|Sig(\alpha)|-|Sig(\mathcal{M})\cap Sig(\alpha)|}$, where $Sig(\mathcal{M})$ is the signature of $\mathcal{M}$, $T(\mathcal{M})$ is the set of axioms already in $\mathcal{M}$, and $Sig(\alpha)$ is the set of terms in the axiom α. The intuition is to find the module $\mathcal{M}$ that includes most of the terms in $Sig(\alpha)$, subject to the restriction that the number of axioms and correspondences in $\mathcal{M}$ (counted in the denominator) will be kept low. Once the module that will host the axiom is decided, the process will construct any correspondences and link-relations to connect terms already in other modules. This happens without affecting the locality of modules, given that no locality constraint has been violated.

Finally, the Node Merging (NM) tuning method reduces the number of modules in the network (which in the general case also reduces the diameter of the network), improving the even distribution of axioms in the modules. Candidate modules to be merged are those (a) with a common neighbor module, and (b) with terms corresponding to dependency graph nodes with the same state. Modules are merged, if the new network will have lower coefficient of variation of the distribution of axioms in the modules. The algorithm iterates until there are no merging actions that can be performed. Merging actions do not violate locality, since, given two $\bot-E-$local modules $\mathcal{M}_i, \mathcal{M}_j$, for $Sig^*(\mathcal{O}\backslash\mathcal{M}_i)$ and $Sig^*(\mathcal{O}\backslash\mathcal{M}_j)$ respectively, $\mathcal{M}_i \cup \mathcal{M}_j$ is also a $\bot - E-$local module for $Sig^*(\mathcal{O}\backslash(\mathcal{M}_i \cup \mathcal{M}_j))$.

It can be proved that the modularization method computes an $E - \mathcal{SHIQ}$ distributed knowledge base Σ, satisfying correctness and completeness of reasoning: by considering each unresolved constraint, and the actions taken to fix the conflict, by associating modules. It follows that the proposed method is a reduction of any $\mathcal{SHIQ}$ ontology $\mathcal{O}$ onto an equivalent $E - \mathcal{SHIQ}$ Σ, and given that the $E - \mathcal{SHIQ}$ reasoner is sound and complete [9], the modularization method satisfies correctness and completeness of reasoning.

4. Experimental Results

To experiment with the proposed modularization method, we have gathered ontologies of various size and expressiveness by crawling web ontology repositories[1]. From these ontologies, we present the results concerning 18 ontologies. The corpus is extended with six versions of the Semantic Information System (SIS) registry [11] which are of considerable size and expressiveness.

The 24 ontologies are categorized by their size: small (between 100-499 axioms), medium (between 500-4999 axioms), large (between 5000-9999 axioms) and SIS ontologies (10000 and more axioms). The expressiveness of the language used in all categories varies from $\mathcal{ALC}$ to $\mathcal{SHIQ}$. All SIS ontologies are within the $\mathcal{ALCHIQ}$ fragment of Description Logics.

[1] (June 25th, 2013), http://bioportal.bioontology.org/, http://www.cs.ox.ac.uk/isg/ontologies/ and http://owl.cs.manchester.ac.uk/owlcorpus

In a previous work [10] we have evaluated montul evaluated against the locality based module extraction algorithm reported in [3], and the results have shown that mONTul computes smaller and less modules. In this paper, the experimental cases concern specific configurations for each ontology. Each configuration depends on the sets of constraints used (and their distinction as hard or soft), the size of the state variables domain **D** varying in the set {2, 10, 20, 40}, and the use of the network tuning method. We distinguish between the configuration of type **(A)**, where constraints GC1,GC2,GC4,LC1,LC2,LC3 are hard and the NM tuning method is used; of type **(B)** where GC1,GC2,GC4 are hard, LC1,LC2,LC3,LC4 are soft, and the NM tuning methods is used, and finally; type **(C)** which is the same as (B) but without the tuning method. Each configuration is denoted by a template of the form "CaseType"_"number of states". E.g. "A_20" denotes type A with |**D**|=20.

Figures 2 and 3 show the results of different mONTul configurations w.r.t. our desiderata for (a) small diameter of the network of modules, (b) small number of cycles of associated modules, (c) balanced distribution of axioms in the modules. Due to space restrictions, Figure 2 presents results only for |**D**| = 2, while comparative results for larger **D** are shown only for the A configuration in Figure 3: These are representative for the results reported from configurations B and C.

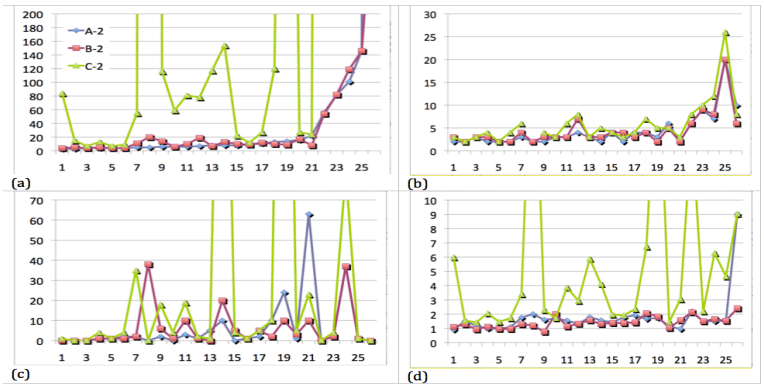

Figure 2. Comparative results for A/B/C_2 configurations and for all networks: (a) Number of modules, (b) Network diameter, (c) Number of cycles, (d) Distribution of axioms.

Ontologies are ordered in the x axis by the (ascending) number of modules produced by the A_2 configuration. Succinctly, configurations A and B compute decompositions with no significant differences in all cases, although we may notice that configurations A present a larger number of cycles in some experimental cases, independently from the size of **D**. Configurations of type C, compared to A and B, compute decompositions with larger number of modules/cycles, and uneven distribution of axioms in modules (figures indicate the coefficient of variation of the distribution). This shows the necessity for tuning the decompositions constructed. Furthermore, as Figure 3 shows, while the size of **D** increases, the results get worst w.r.t. our requirements: This is due to the fact that larger **D** provides more degrees of freedom for the decomposition, thus more modules, greater diameter and more cycles in the network of associated modules.

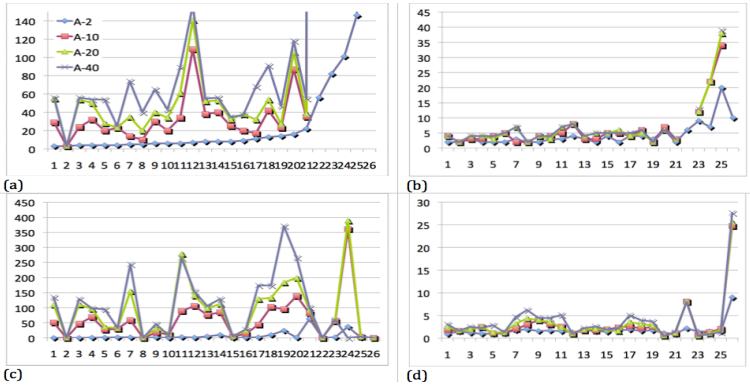

Figure 3. Comparative results for (a) number of modules, (b) network diameter, (c) number of cycles, (d) distribution of axioms for all networks and A_|**D**| configurations.

5. Concluding Remarks

Given an ontology within the $\mathcal{SHIQ}$ fragment of Description Logics, this paper proposes the mONTul method for partitioning this ontology into an arbitrary number of modules that (a) form a distributed $E-\mathcal{SHIQ}$ knowledge base, such that (b) it can be used for correct and complete reasoning, and (c) each module preserves to a great extent the meaning of the terms in its signature. Future work concerns further investigation on the use of different sets of (hard and soft) constraints and methods for tuning the network of associated modules.

References

[1] Bernardo Cuenca Grau, Ian Horrocks, Yevgeny Kazakov and Ulrike Sattler: A Logical Framework for Modularity of Ontologies, *In proc. of IJCAI 2007*, 298–303.

[2] Bernardo Cuenca Grau, Ian Horrocks, Yevgeny Kazakov and Ulrike Sattler: Just the right amount: extracting modules from ontologies, *Proc. of WWW '07*, (2007), 717–726.

[3] Bernardo Cuenca Grau, Ian Horrocks, Yevgeny Kazakov and Ulrike Sattler: Modular reuse of ontologies: Theory and practice. *JAIR*, **31:1**, (2008) 273–318

[4] Boris Konev, Carsten Lutz, Walther Dirk and Frank Wolter: Logical Difference and Module Extraction with CEX and MEX, *Description Logics*, CEUR Workshop Proc., **353**, 2008.

[5] Carsten Lutz, Dirk Walther and Frank Wolter: Conservative Extensions in Expressive Description Logics, *In Proc. of IJCAI 2007*, 453-458.

[6] Jyotishman Pathak, Thomas M. Johnson, and Christopher G. Chute: Survey of modular ontology techniques and their applications in the biomedical domain, *Integr. Comput.-Aided Eng.*, IOS Press, (16:3), (2009), 225–242.

[7] Julian Seidenberg and Alan Rector: Web ontology segmentation: analysis, classification and use, In Proc. of the 15th Int. Conf. on World Wide Web, ACM Press (2006), 13–22.

[8] Heiner Stuckenschmidt and Anne Schlicht: Structure-Based Partitioning of Large Ontologies, *Modular Ontologies*, LNCS, **5445**, (2009), 187–210.

[9] George A. Vouros and Georgios M. Santipantakis: Combining Ontologies with Correspondences and Link Relations: The E-SHIQ Representation Framework, *arXiv*, **1310.2493**, (2013), cs.AI/1310.2493.

[10] Georgios M. Santipantakis, George A. Vouros: Modularizing Ontologies for the Construction of $E-\mathcal{SHIQ}$ Distributed Knowledge Bases, *SETN 2014*, 192–206.

[11] George A. Vouros, et al: A semantic information system for services and traded resources in Grid e-markets, *FGCS*, **26:7**, (2010), 916–933.