Reasoning with a Network of Aligned Ontologies*

Antoine Zimmermann and Chan Le Duc

INRIA Rhône-Alpes - LIG

Abstract. In the context of the Semantic Web or semantic peer to peer systems, many ontologies may exist and be developed independently. Ontology alignments help integrating, mediating or reasoning with a system of networked ontologies. Though different formalisms have already been defined to reason with such systems, they do not consider ontology alignments as first class objects designed by third party ontology matching systems. Correspondences between ontologies are often asserted from an external point of view encompassing both ontologies. We study consistency checking in a network of aligned ontologies represented in Integrated Distributed Description Logics (IDDL). This formalism treats local knowledge (ontologies) and global knowledge (inter-ontology semantic relations, *i.e.*, alignments) separately by distinguishing local interpretations and global interpretation so that local systems do not need to directly connect to each other. We consequently devise a correct and complete algorithm which, although being far from tractacle, has interesting properties: it is independent from the local logics expressing ontologies by encapsulating local reasoners. This shows that consistency of a IDDL system is decidable whenever consistency of the local logics is decidable. Moreover, the expressiveness of local logics does not need to be known as long as local reasoners can handle at least $\mathcal{ALC}$.

1 Introduction

Reasoning on a network of multiple ontologies can be achieved by integration of several knowledge bases or by using non standard distributed logic formalisms. With the first, knowledge must be translated into a common logic, and reasoning is fully centralized. The second option, which has been chosen for Distributed Description Logics (DDL) [3], $\mathcal{E}$-connections [7], Package-based Description Logics (P-DL) [2] or [8] consists in defining new formalisms which allow reasoning with multiple domains in a distributed way. The non-standard semantics of these formalisms reduces conflicts between ontologies, but they do not adequately formalize the quite common case of ontologies related with ontology alignments produced by third party ontology matchers. Indeed, these formalisms assert cross-ontology correspondences (bridge rules, links or imports) from one ontology's point of view, while often, such correspondences are expressed from a point of view that encompasses both aligned ontologies. Consequently, correspondences, being tied to one "context", are not transitive, and therefore, alignments cannot be composed in these languages.

* This work has been partly supported by the European integrated project NeOn (IST-2004-507482).

Our proposed formalism, Integrated Distributed Description Logics (IDDL), addresses this situation and offer sound alignment composition. The principle behind it was presented in [12] under the name *integrated distributed semantics*, and particularized for Description Logics in [10]. This article aims at providing a distributed reasoning procedure for IDDL, which has the following interesting characteristics:

– the distributed process takes advantage of existing DL reasoners (*e.g.*, Pellet, Racer, FacT++, *etc.*);
– local ontologies are encapsulated in the local reasoning system, so it is not necessary to access the content of the ontologies in order to determine the consistency of the overall system;
– the expressiveness of local ontologies is not limited as long as it is a decidable description logic.

The noticeable drawbacks are the following:

– cross-ontology correspondences are limited to concept subsumption or disjointness, and role subsumption (so, role disjointness is not supported); individual correspondences are treated via nominal concepts;
– the algorithm is highly intractable. However this paper is above all concerned by the decidability of IDDL.

The presentation is organized as follows. We start with a presentation of the formalism itself. Then, we describe the reasoning process in the case of concept correspondences alone, in order to lighten the complexity of the notations which will serve to prove the correctness of the algorithm. The following section updates the notations and theorem in the more general case of possible cross-ontology role subsumption. Finally, a discussion on planned implementation and further work is given as well as concluding remarks. Additionally, an Appendix provides a sketch of the proof of the correctness of IDDL decision procedure.

2 Integrated Distributed Description Logics

IDDL is a formalism which inherits from both the field of Description Logics and from the analysis of the forms of distributed semantics in [12].

In a preliminary section, we provide definitions if the syntax and semantics of classical description logics. Thereafter, we provide the definition of correspondence, alignment and distributed system, for which we define a semantics.

2.1 DL: syntax and semantics

IDDL ontologies have the same syntax and semantics as in standard DLs. More precisely, a DL ontology is composed of concepts, roles and individuals, as well as axioms built out of these elements. A concept is either a primitive concept A, or, given concepts C, D, role R, individuals $a_1, \dots, a_k$, and natural number n, $\bot$, $\top$, $C \sqcup D$, $C \sqcap D$, $\exists R.C$, $\forall R.C$, $\leq nR.C$, $\geq nR.C$, $\neg C$ or $\{a_1, \dots, a_k\}$. A role is either a primitive role P, or, given roles R and S, $R \sqcup S$, $R \sqcap S$, $\neg R$, R^-, $R \circ S$ and R^+.

Interpretations are pairs $\langle \Delta^I, \cdot^I \rangle$, where Δ^I is a non-empty set (the domain of interpretation) and $\cdot^I$ is the function of interpretation such that for all primitive concepts A, $A^I \subseteq \Delta^I$, for all primitive roles P, $P^I \subseteq \Delta^I \times \Delta^I$, and for all individuals a, $a^I \in \Delta^I$. Interpretations of complex concepts and roles are inductively defined by $\bot^I = \emptyset$, $\top^I = \Delta^I$, $(C \sqcup D)^I = C^I \cup D^I$, $(C \sqcap D)^I = C^I \cap D^I$, $(\exists R.C)^I = \{x | \exists y. y \in C^I \wedge \langle x, y\rangle \in R^I\}$, $(\forall R.C)^I = \{x | \forall y. \langle x, y\rangle \in R^I \Rightarrow y \in C^I\}$, $(\leq nR.C)^I = \{x | \sharp\{y \in C^I | \langle x, y\rangle \in R^I\} \leq n\}$, $(\geq nR.C)^I = \{x | \sharp\{y \in C^I | \langle x, y\rangle \in R^I\} \geq n\}$, $(\neg C)^I = \Delta^I \setminus C^I$, $\{a_1, \ldots, a_k\} = \{a_1^I, \ldots, a_k^I\}$, $(R \sqcup S)^I = R^I \cup S^I$, $(R \sqcap S)^I = R^I \cap S^I$, $(\neg R)^I = (\Delta^I \times \Delta^I) \setminus R^I$, $(R^-)^I = \{\langle x, y\rangle | \langle y, x\rangle \in R^I\}$, $(R \circ S)^I = \{\langle x, y\rangle | \exists z. \langle x, z\rangle \in R^I \wedge \langle z, y\rangle \in S^I\}$ and $(R^+)^I$ is the reflexive-transitive closure of R^I.

Axioms are either subsumptions $C \sqsubseteq D$, sub-role axioms $R \sqsubseteq S$, instance assertions $C(a)$, role assertions $R(a, b)$ and individual identities $a = b$, where C and D are concepts, R and S are roles, and a and b are individuals. An interpretation I *satisfies* axiom $C \sqsubseteq D$ iff $C^I \subseteq D^I$; it satisfies $R \sqsubseteq S$ iff $R^I \subseteq S^I$; it satisfies $C(a)$ iff $a^I \in C^I$; it satisfies $R(a, b)$ iff $\langle a^I, b^I\rangle \in R^I$; and it satisfies $a = b$ iff $a^I = b^I$. When I satisfies an axiom α, it is denoted by $I \models \alpha$.

An ontology O is composed of a set of terms (primitive concepts/roles and individuals) called the signature of O and denoted by $\mathrm{Sig}(O)$, and a set of axioms denoted by $\mathrm{Ax}(O)$. An interpretation I is a model of an ontology O iff for all $\alpha \in \mathrm{Ax}(O)$, $I \models \alpha$. In this case, we write $I \models O$. The set of all models of an ontology O is denoted by $\mathrm{Mod}(O)$. A semantic consequence of an ontology O is a formula α such that for all $I \in \mathrm{Mod}(O)$, $I \models \alpha$.

2.2 Distributed Systems

A distributed system (DS) is composed of a set of ontologies, connected by ontology alignments. An ontology alignment describes semantic relations between ontologies.

Syntax: An ontology alignment is a set of *correspondences*. A correspondence can be seen as an axiom that asserts a relation between concepts, roles or individuals of two distinct ontologies. They are homologous to bridge rules in DDL. We use a notation similar to DDL in order to identify in which ontology a concept, role or individual is defined. If a concept/role/individual E belongs to ontology i, then we write it $i{:}E$. The 6 possible types of correspondences between ontologies i and j are:

Definition 1 (Correspondence). *A* correspondence *between two ontologies i and j is one of the following formulas:*

- $i{:}C \xleftrightarrow{\sqsubseteq} j{:}D$ *is a cross-ontology concept subsumption;*
- $i{:}R \xleftrightarrow{\sqsubseteq} j{:}S$ *is a cross-ontology role subsumption;*
- $i{:}C \xleftrightarrow{\perp} j{:}D$ *is a cross-ontology concept disjointness;*
- $i{:}R \xleftrightarrow{\perp} j{:}S$ *is a cross-ontology role disjointness;*
- $i{:}a \xleftrightarrow{\in} j{:}C$ *is a cross-ontology membership;*
- $i{:}a \xleftrightarrow{=} j{:}b$ *is a cross-ontology identity.*

Notice that it is possible that $i = j$. Ontology alignments and DL ontologies form the components of a Distributed System in IDDL.

Definition 2 (Distributed system). *A* distributed system *or DS is a pair* $\langle \mathbf{O}, \mathbf{A} \rangle$ *such that* $\mathbf{O}$ *is a set of ontologies, and* $\mathbf{A} = (A_{ij})_{i,j \in \mathbf{O}}$ *is a family of alignments relating ontologies of* $\mathbf{O}$.[1]

Semantics Distributed systems semantics depends on local semantics, but does not interfere with it. A standard DL ontology can be straightforwardly used in IDDL system. Informally, interpreting an IDDL system consists in assigning a standard DL interpretation to each local ontology, then correlating the domains of interpretation thanks to what we call an *equalizing function*.

Definition 3 (Equalizing function). *Given a family of local interpretations* $\mathbf{I}$*, an equalizing function* $\boldsymbol{\varepsilon}$ *is a family of functions indexed by* $\mathbf{I}$ *such that for all* $I_i \in \mathbf{I}$, $\varepsilon_i : \Delta^{I_i} \rightarrow \Delta_\varepsilon$ *where* Δ_ε *is called the global domain of interpretation of* $\boldsymbol{\varepsilon}$.

A distributed interpretation assigns a standard DL interpretation to each ontology in the system, as well as an equalizing function that correlates local knowledge into a global domain of interpretation.

Definition 4 (Distributed interpretation). *Let* $S = \langle \mathbf{O}, \mathbf{A} \rangle$ *be a DS. A* distributed interpretation *of* S *is a pair* $\langle \mathbf{I}, \boldsymbol{\varepsilon} \rangle$ *where* $\mathbf{I}$ *is a family of interpretations indexed by* $\mathbf{O}$, $\boldsymbol{\varepsilon}$ *is an equalizing function for* $\mathbf{I}$*, such that for all* $i \in \mathbf{O}$, I_i *interprets* i *and* $\varepsilon_i : \Delta^{I_i} \rightarrow \Delta_\varepsilon$.

While local satisfiability is the same as standard DL, correspondence satisfaction involves the equalizing function.

Definition 5 (Satisfaction of a correspondence). *Let* S *be a DS, and* i, j *two ontologies of* S*. Let* $\mathcal{I} = \langle \mathbf{I}, \boldsymbol{\varepsilon} \rangle$ *be a distributed interpretation. We define satisfaction of a correspondence* c *(denoted by* $\mathcal{I} \models_d c$*) as follows:*

$$\begin{array}{lll}
\mathcal{I} \models_d i{:}C \xleftrightarrow{\sqsubseteq} j{:}D & \textit{iff} & \varepsilon_i(C^{I_i}) \subseteq \varepsilon_j(D^{I_j}) \\
\mathcal{I} \models_d i{:}R \xleftrightarrow{\sqsubseteq} j{:}S & \textit{iff} & \varepsilon_i(R^{I_i}) \subseteq \varepsilon_j(S^{I_j}) \\
\mathcal{I} \models_d i{:}C \xleftrightarrow{\perp} j{:}D & \textit{iff} & \varepsilon_i(C^{I_i}) \cap \varepsilon_j(D^{I_j}) = \emptyset \\
\mathcal{I} \models_d i{:}R \xleftrightarrow{\perp} j{:}S & \textit{iff} & \varepsilon_i(R^{I_i}) \cap \varepsilon_j(S^{I_j}) = \emptyset \\
\mathcal{I} \models_d i{:}a \xleftrightarrow{\in} j{:}C & \textit{iff} & \varepsilon_i(a^{I_i}) \in \varepsilon_j(C^{I_j}) \\
\mathcal{I} \models_d i{:}a \xleftrightarrow{=} j{:}b & \textit{iff} & \varepsilon_i(a^{I_i}) = \varepsilon_j(b^{I_j})
\end{array}$$

Additionally, for all local formulas $i{:}\phi$, $\mathcal{I} \models_d i{:}\phi$ iff $I_i \models \phi$ (*i.e.*, local satisfaction is equivalent to global satisfaction of local formulas). A distributed interpretation $\mathcal{I}$ satisfies an alignment A iff it satisfies all correspondences of A (denoted by $\mathcal{I} \models_d A$) and

[1] We systematicaly use bold face to denote a mathematical family of elements. So, $\mathbf{O}$ denotes $(O_i)_{i \in I}$ where I is a set of indices.

it satisfies an ontology O_i iff it satisfies all axioms of O_i (denoted by $\mathcal{I} \models_d O_i$). When all ontologies and all alignments are satisfied, the DS is satisfied by the distributed interpretation. In which case we call this interpretation a *model* of the system.

Definition 6 (Model of a DS). *Let $S = \langle \mathbf{O}, \mathbf{A} \rangle$ be a DS. A distributed interpretation $\mathcal{I}$ is a* model *of S (denoted by $\mathcal{I} \models_d S$), iff:*

- *for all $O_i \in \mathbf{O}$, $\mathcal{I} \models_d O_i$;*
- *for all $A_{ij} \in \mathbf{A}$, $\mathcal{I} \models_d A_{ij}$.*

The set of all models of a DS is denoted by $\mathrm{Mod}(S)$. A formula α is a consequence of a DS ($S \models_d \alpha$) iff $\forall \mathcal{M} \in \mathrm{Mod}(S), \mathcal{M} \models_d \alpha$. This model-theoretic semantics offers special challenges to the reasoning infrastructure, that we discuss in next section.

3 Reasoning in IDDL with concept correspondences

In this section, we investigate a reasoning procedure for checking whether or not $S = \langle \mathbf{O}, \mathbf{A} \rangle$ is consistent, in the case when only concepts are put in correspondences. Role correspondences are considered in the next section.

We can reduce the problem of entailment $S \models_d \alpha$ to deciding (in)consistency of a DS when α is either a local GCI ($i{:}C \sqsubseteq D$), a concept correspondence ($i{:}C \xleftrightarrow{\sqsubseteq} j{:}D$ or $i{:}C \xleftrightarrow{\bot} j{:}D$) or a local ABox assertion ($i : C(a)$). Local entailment reduction is straightforward. However, correspondence entailment like $S \models_d i{:}C \xleftrightarrow{\sqsubseteq} j{:}D$ is equivalent to the inconsistency of $S \cup \{i{:}\{a\} \xleftrightarrow{\sqsubseteq} i{:}C\} \cup \{i{:}\{a\} \xleftrightarrow{\bot} j{:}D\}$, where a is a new individual name added to ontology O_i.

When ontologies are correlated with alignments, new deductions may occur. Indeed, cross-ontology knowledge interacts with local knowledge. Moreover, knowledge from one ontology may influence knowledge from another ontology. Besides, local knowledge would also induce cross-ontology knowledge (*i.e.*, alignments). And finally, deductions can be made with and about the alignments alone.

In fact, the difficulty of reasoning in IDDL resides in determining what knowledge propagates from local domains to global domain, or from global to local domains. For instance, if there is a correspondence which asserts disjointness of two concepts from a local ontology then the semantics of the system imposes disjointness of these two concepts in the local ontology. We will show that, in the restricted case when only concept correspondences are allowed, it suffices to propagate only unsatisfiability and non-emptiness of concepts.

Example 1. Let S be the DS composed of $O_1 = \{D_1 \equiv B_1 \sqcap C_1\}$, $O_2 = \{B_2 \sqsubseteq \top, C_2 \sqsubseteq \top\}$ and alignment $A_{12} = \{1{:}B_1 \xleftrightarrow{\sqsubseteq} 2{:}B_2, 1{:}C_1 \xleftrightarrow{\bot} 2{:}B_2, 1{:}D_1 \xleftrightarrow{\sqsupseteq} 2{:}C_2\}$.

We see that $S \models 1{:}B_1 \xleftrightarrow{\bot} 1{:}C_1$. If $B_1 \sqsubseteq \neg C_1$ is added to O_1 (as a consequence of knowledge propagation from the alignments to the ontology) then D_1 becomes unsatisfiable in O_1. From the correspondence $1 : D_1 \xleftrightarrow{\sqsupseteq} 2 : C_2$, it follows that C_2 is unsatisfiable in O_2 as well.

Ex. 1 shows that reasoning on IDDL systems is not trivial and the existing algorithms for reasoning on DL-based ontologies (*e.g.*, tableau algorithms) cannot be directly used.

The principle behind the algorithm is based on the fact that correspondences are similar to axioms, and alignments resemble ontologies. In fact, an alignment represents an ontology which would be interpreted in the global domain of interpretation (see Def. 4). In this algorithm, the alignments will be translated into an ontology (the global ontology). However, this is not enough to check global consistency because local knowledge influences global reasoning. So, the idea consists in extending the global ontology together with the local ontologies by adding specific axioms which represent knowledge propagated through the distributed system.

As a matter of fact, if correspondences are restricted to cross-ontology concept subsumption or disjointness, only concept unsatisfiability and concept non-emptiness[2] can be propogated. Indeed, if a concept is locally interpreted as empty, then its image via the equalizing function is empty too. Conversely, a non-empty set has a non-empty image through ε.

Unfortunately, it is not possible to propagate knowledge by analysing ontologies one by one. A subtle combination of several ontologies **and** alignments can impose unsatisfiability of a locally satisfiable concept (see Ex. 1).

In order to be certain that all concept unsatisfiability and non-emptiness are propagated, our algorithm exhaustively tests each combination of concept unsatisfiability and non-emptiness by explicitely adding these facts, and propagating them accordingly.

In the sequel, we introduce the construction of extended ontologies from $\mathbf{A}$ and $\mathbf{O}$ and we show that the consistency of an IDDL system S is equivalent to the existence of such extended ontologies such that they are consistent.

3.1 Configurations and extended ontologies

This section provides the formal definitions which will finally lead to the construction of the extended ontologies mentioned above. A configuration determines whether certain well-chosen concepts in a vocabulary are unsatisfiable or non-empty. In our specific case, the vocabulary in question is defined by the correspondences. It will be proven that it is sufficient to consider concepts appearing in correspondences when dealing with knowledge propagation in IDDL.

More precisely, concepts occurring as the left or right side of correspondences in alignments constitute the vocabulary of an alignment ontology, namely *global vocabulary*. It consists, in turn, of *local vocabularies* which are originated from local ontologies. The following definitions introduce formally the construction of these elements.

Definition 7 (Local vocabulary). *Let $S = \langle \mathbf{O}, \mathbf{A} \rangle$ be a DS. We denote by $\mathscr{C}_i$ the set that includes the top concept $\top$ and all (primitive or complex) concepts that appear in the left side of correspondences in A_{ij} or in the right side of correspondences in A_{ji}.*

[2] In this paper, "concept non-emptiness" means that an interpretation satisfies the system only if the concept is interpreted as a non-empty set.

Definition 8 (Global vocabulary). *Let $S = \langle \mathbf{O}, \mathbf{A} \rangle$ be a DS. The set of global concept names of S is $\mathscr{C} = \bigcup_{i \in \mathbf{O}} \{i{:}C \mid C \in \mathscr{C}_i\} \cup \{\top\}$. When $w \subseteq \mathscr{C}_i$, we denote by $\widehat{w}$ the set $\{i{:}C \mid C \in w\}$ of (global) concept names. When $W \subseteq \mathscr{C}$, we denote by $W|_i$ the set $\{C \in \mathscr{C}_i \mid i{:}C \in W\}$.*

Example 2. Considering the system of Ex. 1, the local vocabulary $\mathscr{C}_1$ is $\{B_1, C_1, D_1, \top_1\}$ and $\mathscr{C}_2$ is $\{B_2, C_2, \top_2\}$, while the global vocabulary $\mathscr{C}$ is $\{1{:}B_1, 1{:}C_1, 1{:}D_1, 1{:}\top_1, 2{:}B_2, 2{:}C_2, 2{:}\top_2, \top\}$.

As mentioned before Ex. 1 we need to determine the unsatisfiability or non-emptiness of certain concepts but not only the concepts of the vocabularies. It is necessary to know also the unsatisfiability or non-emptiness of all *atomic decompositions* [9] on concepts in a vocabulary. The reason is that, for instance, the non-emptiness of two concepts $C, D \in \mathscr{C}$ does not mean the non-emptiness of $C \sqcap D$ which should be propagated to local ontologies. Concepts defined in Def. 9 express just all atomic decompositions on concepts in a set T.

Definition 9. *Let T be a set of concepts (primitive or complex) including $\top$. For each non empty subset $W \subseteq T$, we define the concept $C_W^T := (\bigsqcap_{X \in W} X \sqcap \bigsqcap_{X' \in T \setminus W} \neg X')$.*

From Def. 9, it follows that all concepts C_W^T are disjoint and their union is equivalent to $\top$. As a consequence, an interpretation of vocabulary T associates to the set of concepts C_W^T a partition of the interpretation domain.

Relying on concepts C_W^T we can define an equivalence relation over the set of interpretations of T as follows: two interpretations belong to an equivalence class if for each subset $W \subseteq T$, they both interpret concept C_W^T as empty, or both interpret it as non empty. The notion of *configuration* defined below represents such equivalence classes. For more convenience, a configuration only indicates the subset of atomic decompositions which will be considered as non-empty, while the others are considered unsatisfiable.

Consequently, a configuration is just a choice of a subset of all atomic decompositions.

Definition 10 (Global configuration). *Let S be a DS with a set of global concept names $\mathscr{C}$. A* global configuration *of S is a subset Ω of $2^{\mathscr{C}}$.*

Configurations are essential because, as we will show, consistency of a DS can be equated to finding a relevant configuration instead of considering the possibly infinite set of all equalizing functions. However, to achieve this, we must translate the configuration into axioms and assertions which express non-emptiness and unsatisfiability, respectively.

We have prepared necessary elements for constructing the so-called *alignment ontology*. This ontology "axiomatizes" the alignments, which represent inter-ontology knowledge. Apart from axioms expressing correspondences in alignments, an alignment ontology includes additional axioms or assertions representing the global configuration.

Definition 11 (Alignment ontology). *Let $S = \langle \mathbf{O}, \mathbf{A} \rangle$ be a DS. Let Ω be a global configuration of S. The* alignment ontology *w.r.t. Ω is an ontology $\widehat{\mathbf{A}}_\Omega$ defined as follows:*

1. *for each $i, j \in \mathbf{O}$, if $i : C \xleftrightarrow{\sqsubseteq} j : D$ (resp. $i : C \xleftrightarrow{\perp} j : D$) is a concept correspondence in $\mathbf{A}$ then $i{:}C \sqsubseteq j{:}D$ (resp. $i{:}C \sqsubseteq \neg j{:}D$) is an axiom of $\widehat{\mathbf{A}}_\Omega$;*
2. *for each $W \in \Omega$, $C_W \equiv \{a_W\}$ is an axiom of $\widehat{\mathbf{A}}_\Omega$ where a_W is a new individual name;*
3. *for each $W \notin \Omega$, $C_W \sqsubseteq \perp$ is an axiom of $\widehat{\mathbf{A}}_\Omega$.*

Axiomatization of the alignments renders explicit the constraints imposed by correspondences. The additional axioms or assertions constrain interpretations to belong to the equivalence class represented by configuration Ω.

Example 3. Reconsidering Ex. 1, if we pick, for instance, the configuration $\Omega = 2^{\mathscr{C}} \setminus \{\emptyset\}$ to build an alignment ontology $\widehat{\mathbf{A}}_\Omega$ according to Def. 11, then $\widehat{\mathbf{A}}_\Omega$ is inconsistent because $W = \{1{:}B_1, 1{:}C_1\} \in \Omega$, $\widehat{\mathbf{A}}_\Omega \models (1{:}B_1 \sqcap 1{:}C_1)(a_W)$ and $\widehat{\mathbf{A}}_\Omega \models 1{:}B_1 \sqsubseteq \neg 1{:}C_1$.

The construction of local configurations is very similar to that of global configuration except that compatibility of local configurations with a given global configuration must be taken into account. This compatibility results from the semantics of IDDL system, which imposes that if the image of a set under an equalizing function is not empty then that set must be not empty.

Definition 12 (Local configuration). *Let $S = \langle \mathbf{O}, \mathbf{A} \rangle$ be a DS. Let Ω be a global configuration of S. For each $O_i \in \mathbf{O}$, we define a* local configuration *of O_i w.r.t. Ω as a subset Ω_i of $2^{\mathscr{C}_i}$. Moreover, if $w \in \Omega_i$ then there must exist $W \in \Omega$ such that $\widehat{w} \subseteq W$.*

As discussed at the beginning of this section, knowledge propagation from alignments to local ontologies is crucial to the construction of extended ontologies which preserve the consistency of an IDDL system. A global configuration Ω and a local configuration Ω_i which is compatible with Ω provide necessary elements to define such extended ontologies. The following definition describes how to propagate knowledge from alignments to local ontologies through the determined configurations.

Definition 13 (Extended ontologies). *Let $S = \langle \mathbf{O}, \mathbf{A} \rangle$ be a DS. Let Ω be a global configuration of S. For each $O_i \in \mathbf{O}$, let Ω_i be a local configuration w.r.t. Ω. The* extended ontology *$\widehat{O}_{\Omega_i}$ w.r.t. Ω_i and Ω is defined as follows:*

1. *$O_i \subseteq \widehat{O}_{\Omega_i}$;*
2. *for each $w \in \Omega_i$, $C_w^{\mathscr{C}_i}(b_w)$ is an axiom of $\widehat{O}_{\Omega_i}$, where b_w is a new individual name;*
3. *for each $w \notin \Omega_i$, $C_w^{\mathscr{C}_i} \sqsubseteq \perp$ is an axiom of $\widehat{O}_{\Omega_i}$;*
4. *for each $W \in \Omega$ and for each $X \in W|_i$, we define a new concept C_W^X for ontology $\widehat{O}_{\Omega_i}$ such that:*
   (a) *$C_W^X \sqsubseteq X \sqcap \displaystyle\bigsqcap_{X' \in \mathscr{C}_i \setminus W|_i} \neg X'$ is an axiom of $\widehat{O}_{\Omega_i}$;*
   (b) *$C_W^X(b_W^X)$ is an axiom of $\widehat{O}_{\Omega_i}$ with b_W^X a new individual name in $\widehat{O}_{\Omega_i}$;*
   (c) *$C_W^X \sqsubseteq C_W^\top$ is an axiom of $\widehat{O}_{\Omega_i}$;*

5. *for each* $W, W' \subseteq \mathscr{C}$ *such that* $W \neq W'$, $C_W^\top \sqsubseteq \neg C_{W'}^\top$ *is an axiom of* $\widehat{O}_{\Omega_i}$.

Notice that the propagation of knowledge through a global configuration is not straightforward. The non-emptiness expressed by the assertion $C_W \equiv \{a_W\}$ indicates that each concept of W coming from the local vocabulary $\mathscr{C}_i$ must be individually non empty, but not necessarily *conjonctly* non-empty. Consequently, the decomposition of the concept C_W for the propagation as described in the item 4a in Def. 13 is necessary.

The following theorem establishes the most important result in the present section. It asserts that an IDDL system can be translated into an alignment ontology and extended ontologies that preserve the semantics of the IDDL system.

Theorem 1 (DS Consistency). *Let* $S = \langle \mathbf{O}, \mathbf{A} \rangle$ *be a DS.* S *is consistent iff there exist a global configuration* Ω *of* S *and a local configuration* Ω_i *for each* $O_i \in \mathbf{O}$ *w.r.t.* Ω *such that the alignment ontology* $\widehat{\mathbf{A}}_\Omega$ *and the extended local ontologies* $\{\widehat{O}_{\Omega_i}\}$ *as defined in Def. 11 and Def. 13 are consistent.*

A proof of this theorem is given in a technical report [11].

4 Reasoning with Cross-ontology Role Subsumption

In this section, we devise a new reasoning procedure which now takes into account cross-ontology role subsumption. The principle behind this improved reasoning task is the same as before, except that configurations must be extended to take into account the roles involved in correspondences. Since most of the definitions necessary for this part are the same or similar as the ones for the previous part, we simply update existing definitions or add new definitions when necessary.

First, a new notion of role vocabulary must be defined, locally or globally.

Definition 14 (Local role vocabulary). *Let* $S = \langle \mathbf{O}, \mathbf{A} \rangle$ *be a DS. We denote by* $\mathscr{R}_i$ *the set that includes primitive or complex roles that appear in the left side of correspondences in* A_{ij} *or in the right side of correspondences in* A_{ji} *together with their inverse roles (i.e.,* $R \in \mathscr{R}_i \longleftrightarrow R^- \in \mathscr{R}_i$*).*

Definition 15 (Global role vocabulary). *Let* $S = \langle \mathbf{O}, \mathbf{A} \rangle$ *be a DS. The set of global role names of* S *is* $\mathscr{R} = \bigcup_{i \in \mathbf{O}} \{i{:}R \mid R \in \mathscr{R}_i\}$.

Now we must define a new kind of configuration which has to be considered in addition to the already defined global and local configurations. However, the treatment of roles is quite different from the treatment of concepts only, because there are interactions between roles and concepts. Therefore, we need to keep track of role satisfiability in addition to concept satisfiability.

We do that by considering a given (concept) configuration Ω which represents a partition of the domain of interpretation. Then, according to this configuration, we define the role configuration as a family of relations over Ω indexed by the set of roles. In other terms, we determine in a role configuration whether there exists a relation R between two sets in the partition Ω.

Definition 16 (Role configuration). *Let S be a DS with a set of global role names $\mathscr{R}$. Let Ω be a global configuration of S. A* role configuration *of S w.r.t. Ω is a subset Φ_Ω of $\Omega \times \Omega \times \mathscr{R}$.*

The introduction of role configuration leads to additional constraints on the alignment ontology that we summarize in this addendum to Def. 11.

Definition 17 (Alignment ontology (revised)). *Let $S = \langle \mathbf{O}, \mathbf{A} \rangle$ be a DS. Let Ω be a global configuration of S, and let Φ_Ω be a role configuration w.r.t. Ω. The* alignment ontology *w.r.t. Ω and Φ_Ω is an ontology $\widehat{\mathbf{A}}_\Omega$ defined as follows:*

1. 2. and 3. See Def. 11;
4. for each $i, j \in \mathbf{O}$, if $i{:}R \xleftrightarrow{\sqsubseteq} j{:}S$ is a role correspondence in $\mathbf{A}$ then $i{:}R \sqsubseteq j{:}S$ is a sub-role axiom of $\widehat{\mathbf{A}}_\Omega$;
5. for each $\langle W, W', R \rangle \in \Phi_\Omega$, $C_W \sqsubseteq \exists R.C_{W'}$ is an axiom of $\widehat{\mathbf{A}}_\Omega$;
6. for each $\langle W, W', R \rangle \notin \Phi_\Omega$, $C_W \sqsubseteq \forall R.\neg C_{W'}$ is an axiom of $\widehat{\mathbf{A}}_\Omega$;

The axioms introduced by items 5 and 6 in Def. 17 express the semantics of a role configuration: they impose a R connection or no R connection between two atomic decompositions on concepts in $\mathscr{C}$.

Similarly to local configurations, local role configurations have to satisfy constraints which reflect the propagation of knowledge from the alignment ontology to extended ontologies.

Definition 18 (Local role configuration). *Let S be a DS with a set of global role names $\mathscr{R}$. Let Ω be a global configuration of S, let Φ_Ω be a role configuration w.r.t. Ω, and Ω_i a local configuration of O_i w.r.t. Ω. A* local role configuration *of O_i w.r.t. Ω, Ω_i and Φ_Ω is a subset Φ_{Ω_i} of $\Omega_i \times \Omega_i \times \mathscr{R}_i$ such that $\langle w, w', R \rangle \in \Phi_{\Omega_i}$ implies that there exists $W, W' \in \Omega$ such that $w \subseteq W|_i$, $w' \subseteq W'|_i$ and $\langle W, W', i{:}R \rangle \in \Phi_\Omega$.*

The extended ontologies are now further extended with axioms which involve roles.

Definition 19 (Extended ontologies (revised)). *Let $S = \langle \mathbf{O}, \mathbf{A} \rangle$ be a DS. Let Ω be a global configuration and Φ_Ω a role configuration of S. For each $O_i \in \mathbf{O}$, let Ω_i be a local configuration w.r.t. Ω and Φ_{Ω_i} be a local role configuration w.r.t. Ω, Ω_i and Φ_Ω. The* extended ontology *$\widehat{O}_{\Omega_i}$ w.r.t. Ω_i, Ω, Φ_Ω and Φ_{Ω_i} is defined as follows:*

1. 2. 3. 4. and 5. See Def. 13;
6. for each $\langle w, w', R \rangle \in \Phi_{\Omega_i}$, $(C_w^{\mathscr{C}_i} \sqcap \exists R.C_{w'}^{\mathscr{C}_i})(b_{w,w'}^R)$ is an axiom of $\widehat{O}_{\Omega_i}$, where $b_{w,w'}^R$ is a new individual name;
7. for each $\langle w, w', R \rangle \notin \Phi_{\Omega_i}$, $C_w^{\mathscr{C}_i} \sqsubseteq \forall R.\neg C_{w'}^{\mathscr{C}_i}$ is an axiom of $\widehat{O}_{\Omega_i}$;
8. for each $W, W' \subseteq \Omega$, and each $R \in \mathscr{R}_i$, we define a new concept name $C_{W,W'}^R$ for ontology $\widehat{O}_{\Omega_i}$ such that:
   (a) $C_{W,W'}^R \sqsubseteq C_W^\top$ is an axiom of $\widehat{O}_{\Omega_i}$;
   (b) if $\langle W, W', i:R \rangle \in \Phi_\Omega$ then $C_{W,W'}^R \sqsubseteq \exists R.C_{W',W}^{R^-}$ and $C_{W,W'}^R(\beta_{W,W'}^R)$ are axioms of $\widehat{O}_{\Omega_i}$ with $\beta_{W,W'}^R$ a new individual name;

(c) *else,* $C^R_{W,W'} \sqsubseteq \forall R.\neg C^{R^-}_{W',W}$ *is an axiom of* $\widehat{O}_{\Omega_i}$*;*

9. *for each* $R \in \mathscr{R}_i$*,* $\exists R.\top \sqsubseteq \bigsqcup_{W,W' \in \Omega} C^R_{W,W'}$ *is an axiom of* $\widehat{O}_{\Omega_i}$*.*

In the previous definition, item 6 means that a triple $\langle w, w', R\rangle$ in the local role configuration determines the existence of a relation R between some member of $C^{\mathscr{C}_i}_w$ and some member of $C^{\mathscr{C}_i}_{w'}$. Conversely, item 7 means that whenever a triple $\langle w, w', R\rangle$ is not in the local role configuration, then concepts $C^{\mathscr{C}_i}_w$ and $C^{\mathscr{C}_i}_{w'}$ are not related through R. Item 8 adds a concept $C^R_{W,W'}$ which represents the set of elements of local concept $C^\top_W$ which have their counterparts in global concept C_W and are in relation through R with elements which have their own counterparts in $C_{W'}$. Finally, item 9 asserts that any element involved in a relation R must belong to one of the newly introduced sets $C^R_{W,W'}$ for some W and W'. This last item is important to ensure that the role structure is correctly propagated.

Theorem 2 (DS Consistency). *Let* $S = \langle \mathbf{O}, \mathbf{A}\rangle$ *be a DS.* S *is consistent iff there exist a global configuration* Ω *of* S*, a role configuration* Φ_Ω *w.r.t.* Ω*, local configurations* Ω_i *for all* $O_i \in \mathbf{O}$ *w.r.t.* Ω *and local role configurations* Φ_{Ω_i} *w.r.t.* Ω*,* Ω_i *and* Φ_Ω*, such that the alignment ontology* $\widehat{\mathbf{A}}_\Omega$ *and the extended local ontologies* $\{\widehat{O}_{\Omega_i}\}$ *as defined in Def. 17 and Def. 19 are consistent.*

5 Algorithms and Improvements

In this section, we try to devise an explicit algorithm for checking the consistency of a distributed system in IDDL. We first present a naive algorithm, as a direct application of Theo. 1. Then, we propose a simple optimization for this particular problem. Finally, we show how the very same principle can be used in a less expressive setting to ensure a tractable consistency checking procedure.

5.1 Naive algorithm

Theo. 1 provides enough information for building a naive but correct and complete algorithm, which corresponds to an exhaustive traversal of all possible configurations (see Algo. 1).

Property 1. Given c the number of global concepts in $\mathscr{C}$ and N the number of ontologies in the system, the number of calls for consistency checking of extended ontologies in Alg. 1 is bounded by $N2^{(2^{c+1})}$. Moreover, the size of the extended ontologies to be checked is in the order of $O(2^c)$.

Proof. There are as many global configurations as there are subsets of $2^{\mathscr{C}}$, *i.e.*, $2^{(2^c)}$. For each global configuration, all local configurations have to be tested, for all ontologies. The number of local configuration, for a given ontology in the system, is bounded by $2^{(2^c)}$, and there are N ontologies to be checked. So the total number of consistency checking is bounded by $N2^{(2^c)} \cdot 2^{(2^c)} = N2^{(2^{c+1})}$.

input : $S = \langle \mathbf{O}, \mathbf{A} \rangle$ with $\mathbf{A} = \{A_{ij} \mid i, j \in \mathbf{O}\}$
output: IsConsistent(S)

foreach *global configuration* $\Omega \subseteq 2^{\mathscr{C}}$ **do**
  if `Consistent(`$\widehat{\mathbf{A}}_{\Omega}$`)` **then**
    foreach *family of local configurations* $(\Omega^i)_{i \in \mathbf{O}}$ **do**
      if `Consistent(`$\widehat{O}_{\Omega^i}$`)` *for all* $i \in \mathbf{O}$ **then**
        return `true`;

return `false`;

Algorithm 1: Consistency($\langle \mathbf{O}, \mathbf{A} \rangle$)

Prop. 1 shows that the complexity of Alg. 1 can be determined from that of consistency checking of extended ontologies. Whatever the local algorithm complexity, the complexity of this global consistency checking algorithm is at least in 2EXPTIME. In the case when the local algorithm is itself in 2EXPTIME (which can happen with some tableau algorithm over very expressive description logics), the global consistency checking algorithm is in 3EXPTIME.

This high intractibility must be balanced with the good properties that it guarantees. First, the algorithm proves that our distributed formalism is decidable whenever local logics are decidable. Second, the actual local logics need not be known, as well as the local decision procedure. Therefore, ontologies can be encapsulated in an interface which only communicate the consistency of its internal ontology extended with well defined axioms. Moreover, the expressiveness of the axioms added to the extended ontologies are restricted to $\mathcal{ALC}$, which is simple enough to cover many existing DL reasoners. The goal of the next section is to explore possible optimization for this particular problem.

5.2 Optimizing the algorithm

This section proposes several simple optimizations which significantly decrease complexity. Unfortunately, they do not change the class of complexity, but help approaching tractability in "favorable situations". The goal is to reduce as much as possible the number of configurations that must be considered. To simplify this discussion, we only focus on the global concept configurations. We consider three complementary methods for improving the algorithm.

Using correspondences: This first method takes advantage of the correspondences to systematically decrease the possible configurations. Indeed, it may be noticed that if $i:C \overset{\sqsubseteq}{\leftrightarrow} j:D \in A_{ij}$, then the non-emptiness of C implies the non-emptiness of D. Therefore, it is not necessary to inspect configurations containing $W \subseteq \mathscr{C}$ such that $C \in W$ and $D \notin W$. Additionally, if $i:C \overset{\perp}{\leftrightarrow} j:D \in A_{ij}$, then it is not necessary to inspect configurations containing $W \subseteq \mathscr{C}$ such that $C, D \in W$ since $i:C \sqcap D$ is necessarily empty. This can decrease the search space a lot. For instance, if there are n

concepts $C_1, \ldots, C_n$ such that $i : C_k \overset{\sqsubseteq}{\leftrightarrow} j : C_{k+1}$, then the non-emptiness of any C_k implies the non-emptiness of all C_l for all $l \geq k$. So, in the best possible situation, the number of configurations to be tested would be linear in the size of the alignements. Nonetheless, in the worst situation, the number of configurations is still exponential. Indeed, if there are n concepts $D_1, \ldots, D_n$ and a concept C such that $i : C \overset{\sqsubseteq}{\leftrightarrow} j : D_k$ for all k, then the non-emptiness of C implies the non-emptiness of all D_k, but when C is empty, the concepts D_k can be independently empty or non-empty. However interesting compared to the brute force approach, this optimization still places the algorithm in the double exponential class. We admit that it is still too high, but this worst case complexity can be avoided in many cases using the two following optimizations.

Using backtracking techniques: Thanks to Theo. 1, the problem of checking consistency has been reformulated into finding a configuration. We can notice that an appropriate configuration can be determined step by step with a decision tree, by deciding whether a given subset $W \subseteq \mathscr{C}$ is in the configuration or not. In fact, each node of the tree asks whether C_W is empty or not. In case C_W can neither be empty nor non-empty, the algorithm must backtrack and try another decision for the previous subset considered. Fig. 1 shows a part of a possible decision tree.

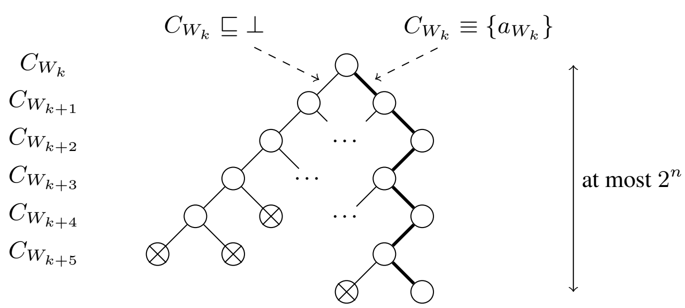

Fig. 1. At each node, the left branch indicates that the concept C_W is asserted as an empty concept ($C_W \sqsubseteq \bot$), while the right branch indicates a non empty concept ($C_W \equiv \{a_W\}$ for a new a_W). The thick path indicates a possible configuration for the distributed system.

Other backtracking techniques like backjumping may be used. With such a method, the number of calls to local reasoners may be reduced to 2^n in favorable cases, not to mention additional reductions due to the previous optimization.

Additional optimization: In the course of reasoning at a certain level of the decision tree, if it can be proved that a concept C is empty ($C \sqsubseteq \bot$), then it can also be asserted that all conjunctions of C with any other concepts are also empty. More precisely, for all $W \subseteq \mathscr{C}$ such that $C \in W$, the system implies that $C_W \sqsubseteq \bot$, so all configurations containing W can be eliminated. This further decreases the search space. Consequently, we conjecture that there are practical cases where reasoning with our algorithm can be carried out, in spite of the very high worst case complexity. Nonetheless, since these optimizations are still insufficient to treat hard cases practically, we study another possible improvement that can be done when the alignments are less expressive.

5.3 Reducing the expressivity of alignments

The highly intractable complexity of the algorithm for checking consistency of an IDDL system is originated from the fact that it may propagate to local ontologies a double exponential number of configurations with exponential size. These configurations represent the structure of models for alignment ontologies which are expressed in a sublogic of $\mathcal{ALC}$. We can simplify the structure of models by removing cross-ontology concept and role disjointness from the alignment language. Tableau-based algorithms, for example in [1], can generate a singleton model for a consistent ontology involving only subsumption axioms between primitive concepts or roles, *i.e.*, each configuration represents now the structure of a singleton model. Therefore, all concepts are interpreted either as the singleton or the empty set. Consequently, it is sufficient to represent models by a set of non-empty primitive concepts.

If $\mathscr{C}$ denotes the global vocabulary of an IDDL system (Def. 8) then a global configuration is now defined as $\Omega \subseteq \mathscr{C}$ and the algorithm needs to call to local reasoners at most $2^{\mathscr{C}}$ times with these polynomial configurations. This is true because, in absence of disjointness, testing whether $\bigsqcap_{X \in W} X \sqcap \bigsqcap_{X' \in T \setminus W} \neg X'$ is empty or not can be reduced to testing whether each primitive concept X are empty or equal to an identified singleton.

6 Conclusion and Future Work

We have proposed a reasoning procedure for IDDL which determines the consistency of a distributed system of DL ontologies and alignments. On the one hand, it only requires the minimal support of $\mathcal{ALC}$ reasoning locally, while there is no upper bound expressivity. On the other hand, alignments are currently limited to cross-ontology concept subsumption or disjointness and role subsumption. This restriction on the expressiveness of alignment language is not really severe since alignments produced by almost all ontology matching algorithms [5] are expressible within this restricted alignment language. Furthermore, the majority of these algorithms (OLA, AROMA, Falcon-AO, *etc*. [5]) yields only cross-ontology concept or role subsumption. This meets exactly the reduction of expressiveness of alignment language presented in Sect.5.2.

Several research directions are considered to continue this work. We plan to further optimize the algorithm. This will lead to a distributed implementation taking advantage of various reasoners encapsulating ontologies of unknown complexity. In particular, this would fit quite well in our modular framework presented in [6]. There are also potential optimization when local expressivity is limited to a logic known in advance. Another direction would involve peer reasoning, which means defining the inferences produced by a local reasoner taking advantage of global knowledge in a network of aligned ontologies.

Finally, our goal is to revise the consistency checking procedure by taking into account role disjointness. Eventually, we hope to extend the expressivity of the alignment language, by adding specific constructors in line with the expressive ontology mapping language proposed in [4].

References

1. Franz Baader. Tableau Algorithms for Description Logics. In R. Dyckhoff, editor, *Automated Reasoning with Analytic Tableaux and Related Methods, International Conference, TABLEAUX 2000, St Andrews, Scotland, UK, July 3-7, 2000, Proceedings*, volume 1847 of *LNCS*, pages 1–18. Springer, 2000.
2. Jie Bao, Doina Caragea, and Vasant Honavar. On the Semantics of Linking and Importing in Modular Ontologies. In I. F. Cruz, S. Decker, D. Allemang, C. Preist, D. Schwabe, P. Mika, M. Uschold, and L. Aroyo, editors, *The Semantic Web - ISWC 2006, 5th International Semantic Web Conference, ISWC 2006, Athens, GA, USA, November 5-9, 2006, Proceedings*, volume 4273 of *LNCS*, pages 72–86. Springer, 2006.
3. Alex Borgida and Luciano Serafini. Distributed Description Logics: Assimilating Information from Peer Sources. *Journal on Data Semantics I*, 2008:153–184, 2003.
4. Jérôme Euzenat, François Scharffe, and Antoine Zimmermann. Expressive alignment language and implementation. Deliverable 2.2.10, Knowledge Web NoE, 2007. `ftp://ftp.inrialpes.fr/pub/exmo/reports/kweb-2210.pdf`.
5. Jérôme Euzenat and Pavel Shvaiko. *Ontology Matching*. Springer, Heidelberg (DE), 2007.
6. Jérôme Euzenat, Antoine Zimmermann, and Fred Freitas. Alignment-based modules for encapsulating ontologies. In B. Cuenca-Grau, V. G. Honavar, A. Schlicht, and F. Wolter, editors, *Second International Workshop on Modular Ontologies (WOMO 2007)*, pages 32–45, 2007.
7. Oliver Kutz, Carsten Lutz, Frank Wolter, and Michael Zakharyaschev. $\mathcal{E}$-connections of abstract description systems. *Artificial Intelligence*, 156(1):1–73, 2004.
8. Maurizio Lenzerini. Data integration: a theoretical perspective. In L. Popa, editor, *Proceedings of the Twenty-first ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, June 3-5, Madison, Wisconsin, USA*, pages 233–246. ACM, 2002.
9. Hans Jürgen Ohlbach and Jana Koehler.
10. Antoine Zimmermann. Integrated Distributed Description Logics. In D. Calvanese, E. Franconi, V. Haarslev, D. Lembo, B. Motik, S. Tessaris, and A.-Y. Turhan, editors, *Proceedings of the 20th International Workshop on Description Logics DL'07*, pages 507–514. Bolzano University Press, 2007.
11. Antoine Zimmermann and Chan Le Duc. Reasoning on a Network of Aligned Ontologies. Technical report, INRIA, March 2008. `http://hal.inria.fr/inria-00267808/fr/`.
12. Antoine Zimmermann and Jérôme Euzenat. Three Semantics for Distributed Systems and their Relations with Alignment Composition. In I. F. Cruz, S. Decker, D. Allemang, C. Preist, D. Schwabe, P. Mika, M. Uschold, and L. Aroyo, editors, *The Semantic Web - ISWC 2006, 5th International Semantic Web Conference, ISWC 2006, Athens, GA, USA, November 5-9, 2006, Proceedings*, volume 4273 of *LNCS*, pages 16–29. Springer, 2006.