An Efficient Method for Computing Alignment Diagnoses

Christian Meilicke and Heiner Stuckenschmidt

Computer Science Institute, University of Mannheim,
B6,26 68159 Mannheim, Germany

Abstract. Formal, logic-based semantics have long been neglected in ontology matching. As a result, almost all matching systems produce incoherent alignments of ontologies. In this paper we propose a new method for repairing such incoherent alignments that extends previous work on this subject. We describe our approach within the theory of diagnosis and introduce the notion of a local optimal diagnosis. We argue that computing a local optimal diagnosis is a reasonable choice for resolving alignment incoherence and suggest an efficient algorithm. This algorithm partially exploits incomplete reasoning techniques to increase runtime performance. Nevertheless, the completeness and optimality of the solution is still preserved. Finally, we test our approach in an experimental study and discuss results with respect to runtime and diagnostic quality.[1]

1 Introduction

It has widely been acknowledged that logical semantics and reasoning are the basis of intelligent applications on the semantic web. This is underlined by the design of standard languages, like the Web Ontology Language (OWL), which have a clearly defined logical semantics. Contrary to this, in the area of ontology matching the use of logical semantics as a guiding principle has long been neglected. Existing matching systems are primarily based on lexical and heuristic methods [2] that often result in alignments that contain logical contradictions. At first glimpse some systems seem to be an exception, for example ASMOV and S-Match. ASMOV [5] has become a successful participant of the OAEI over the last years. One of its constituents is a semantic verification component used to filter out conflicting correspondence. In particular, a comprehensive set of pattern is applied to detect certain kind of conflicts. However, ASMOV lacks a well defined alignment semantics and notions as correctness or completeness are thus not applicable. The S-Match system [4], on the contrary, employs sound and complete reasoning procedures. Nevertheless, the underlying semantics is restricted to propositional logic due to the fact that ontologies are interpreted as tree-like structures. S-Match can thus not guarantee to generate a coherent alignment between expressive OWL-ontologies. We have already argued that the problem of generating coherent alignments can best be solved by applying principles of diagnostic reasoning [11]. In this paper, we extend previous work on this topic in different directions.

[1] An extended version of this paper is available as technical report at `http://webrum.uni-mannheim.de/math/lski/matching/lod/`.

– We define the general notion of a reductionistic alignment semantics and introduce a natural interpretation as concrete specification. Contrary to previous work, we support different alignment semantics within our framework.
– As extension of our previous work we do not only cover concept correspondences but additionally support correspondences between properties.
– We describe the problem of repairing incoherent alignments in terms of Reiters theory of diagnosis [14] and introduce the notion of a local optimal diagnosis.
– We present an algorithm for constructing a local optimal diagnosis - based on the algorithm described in [12] - and show how this algorithm can be enhanced by partially exploiting efficient but incomplete reasoning methods.
– We report on several experiments concerned with both the diagnostic quality as well as the runtime of both algorithms.

In Section 2 we define our terminology and introduce some definitions centered around the the notion of alignment incoherence. In Section 3 we argue that repairing an incoherent alignment can be understood as diagnosis task. In particular, we introduce the notion of a local optimal diagnosis. In Section 4 we briefly introduce different reasoning techniques and algorithms exploiting these reasoning techniques in order to compute a local optimal diagnosis. These algorithms are applied on different datasets in Section 5 where we also discuss the results and compare them against other approaches. In Section 6 we end with a short summary and some concluding remarks.

2 Preliminaries

The task of aligning two ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$ (sets of axioms) can be understood as detecting links between elements of $\mathcal{O}_1$ and $\mathcal{O}_2$. These links are referred to as correspondences and express a semantic relation. According to Euzenat and Shvaiko [2] we define a correspondence as follows and introduce an alignment as set of correspondences.

Definition 1 (Correspondence and Alignment). *Given ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$, let Q be a function that defines sets of matchable elements $Q(\mathcal{O}_1)$ and $Q(\mathcal{O}_2)$. A correspondence between $\mathcal{O}_1$ and $\mathcal{O}_2$ is a 4-tuple $\langle e, e', r, n\rangle$ such that $e \in Q(\mathcal{O}_1)$ and $e' \in Q(\mathcal{O}_2)$, r is a semantic relation, and $n \in [0, 1]$ is a confidence value. An alignment $\mathcal{A}$ between $\mathcal{O}_1$ and $\mathcal{O}_2$ is a set of correspondences between $\mathcal{O}_1$ and $\mathcal{O}_2$.*

Our approach is applicable to alignments between ontologies represented in Description Logics, e.g. to alignments between OWL-DL ontologies. In this work the matchable elements $Q(\mathcal{O})$ are restricted to be atomic concepts or atomic properties. Further r is a semantic relation expressing equivalence or subsumption. We use the symbols $\overset{\leftrightarrow}{\equiv}$, $\overset{\leftrightarrow}{\sqsubseteq}$ and $\overset{\leftrightarrow}{\sqsupseteq}$ to refer to these relations. The semantics of these symbols has not yet been specified, although we might have a rough idea about their interpretation. The confidence value n describes the trust in the correctness of a correspondence. Given a correspondence c, we use $\mathit{conf}(c) = n$ to refer to the confidence of c. Additionally, we require that in an alignment $\mathcal{A}$ there exist no $c \neq c' \in \mathcal{A}$ such that $\mathit{conf}(c) = \mathit{conf}(c')$. We know that most matching systems will not fullfill this requirement. Another source of evidence has to decide which correspondence should be annoted with higher confidence. Thus, we avoid an explicit treatment of different total orderings derivable from the partial order of confidence values.[2] In the following we frequently need to talk about concepts or properties of an ontology $\mathcal{O}_i$. We use prefix notation $i\#e$ to uniquely determine that an entity e belongs to the signature of $\mathcal{O}_i$.

A concept $i\#C$ is defined to be unsatisfiable iff all models of $\mathcal{O}_i$ interpret $i\#C$ as empty set. We use the notion of unsatisfiability in a wider sense and define it with respect to both concepts and properties.

Definition 2 (Unsatisfiability). *A concept or property $i\#e$ is unsatisfiable in ontology $\mathcal{O}_i$, iff for all models $\mathcal{I}$ of $\mathcal{O}_i$ we have $i\#e^{\mathcal{I}} = \emptyset$. Otherwise $i\#e$ is satisfiable in $\mathcal{O}_i$.*

Usually, an ontology is referred to as incoherent whenever it contains an atomic unsatisfiable concept. We define ontology incoherence as follows.

Definition 3 (Ontology Incoherence). *An ontology $\mathcal{O}$ is incoherent iff there exists an atomic unsatisfiable concept or property in $\mathcal{O}$. Otherwise $\mathcal{O}$ is coherent.*

There are two ways to introduce the notion of alignment incoherence. The first approach requires a *specific model-theoretic alignment semantics*. Distributed Description Logics (DDL)[1] is an example for such a specific semantics, which we focused on in previous work [11]. The second approach, already sketched in [7], is based on *interpreting an alignment as a set of axioms X in a merged ontology*. Given an alignment $\mathcal{A}$ between $\mathcal{O}_1$ and $\mathcal{O}_2$, the (in)coherence of $\mathcal{A}$ is reduced to the (in)coherence $\mathcal{O}_1 \cup \mathcal{O}_2 \cup X$. We refer to such a semantics as reductionistic alignment semantics.

Definition 4 (Reductionistic Semantics). *Given an alignment $\mathcal{A}$ between ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$. A reductionistic alignment semantics $\mathcal{S} = \langle ext, trans \rangle$ is a pair of functions where ext maps an ontology to a set of axioms (extension function) and $trans$ maps an alignment to a set of axioms (translation function).*

Considering its role in the context of a merged ontology, it becomes clear how to apply such a reductionistic alignment semantics, abbreviated as alignment semantics in the following.

Definition 5 (Merged ontology). *Given an alignment $\mathcal{A}$ between ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$ and an alignment semantics $\mathcal{S} = \langle ext, trans \rangle$. The merged ontology is defined as $\mathcal{O}_1 \cup^{\mathcal{S}}_{\mathcal{A}} \mathcal{O}_2 = \mathcal{O}_1 \cup \mathcal{O}_2 \cup ext(\mathcal{O}_1) \cup ext(\mathcal{O}_2) \cup trans(\mathcal{A})$.*

The merged ontology is merely a technical means to treat different semantics within a similar framework. Based on this framework we apply the definition of ontology incoherence in the context of a merged ontology resulting in the notion of alignment incoherence.

[2] For the experiments reported on in Section 5 we derived a total order - given correspondences with the same confidence value - from the lexicographical ordering of the URIs of the matched entities. Experiments with different orderings resulted in insignificant differences.

Definition 6 (Alignment Incoherence). *Given an alignment $\mathcal{A}$ between ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$ and an alignment semantics $\mathcal{S}$. $\mathcal{A}$ is incoherent with respect to $\mathcal{O}_1$ and $\mathcal{O}_2$ according to $\mathcal{S}$, iff there exists an atomic concept or property $i\#C$ with $i \in \{1,2\}$ that is satisfiable in $\mathcal{O}_i$ and unsatisfiable in $\mathcal{O}_1 \cup^{\mathcal{S}}_{\mathcal{A}} \mathcal{O}_2$. Otherwise $\mathcal{A}$ is coherent.*

We now introduce an example of a reductionistic alignment semantics, primarily defined in [7] and [8] with respect to a less general framework.

Definition 7 (Natural Semantics). *Given an alignment $\mathcal{A}$ and an ontology $\mathcal{O}$. The natural semantics $\mathcal{S}_n = \langle ext_n, trans_n \rangle$ is defined by a specification of its components $ext_n(\mathcal{O}) \mapsto \emptyset$ and $trans_n(\mathcal{A}) \mapsto \{t_n(c) | c \in \mathcal{A}\}$ where t_n is defined as*

$$t_n(c) \mapsto \begin{cases} 1\#e \equiv 2\#e' & \text{if } r = \overset{\leftrightarrow}{\equiv} \\ 1\#e \sqsubseteq 2\#e' & \text{if } r = \overset{\leftrightarrow}{\sqsubseteq} \\ 1\#e \sqsupseteq 2\#e' & \text{if } r = \overset{\leftrightarrow}{\sqsupseteq} \end{cases}$$

The natural alignment semantics consists of an empty extension function ext and a translation function $trans$ that maps correspondences one-to-one to axioms. It can be seen as self-evident and straightforward way to interpret correspondences as axioms.

An example for an alignment semantics with $ext(\mathcal{O}) \neq \emptyset$ is given by DDL. DDL is a formalism for supporting distributed reasoning based on a semantics where each ontology is interpreted within its own domain interrelated via bridge rules. Nevertheless, it is also possible to reduce DDL to ordinary DL [1]. As a result we obtain a reductionistic alignment semantics where the extension function maps $\mathcal{O}_1$ and $\mathcal{O}_2$ to a non empty set of additional axioms while the translation function differs significantly from the translation function of the natural semantics.

3 Problem Statement

In this section we show that the problem of debugging alignments can be understood as diagnostic problem and characterize a certain type of diagnosis. Throughout the remaining parts we use $\mathcal{A}$ to refer to an alignment, we use $\mathcal{O}$ with or without subscript to refer to an ontology, and $\mathcal{S}$ to refer to some reductionistic alignment semantics.

In ontology debugging a minimal incoherency preserving sub-TBox (MIPS) $\mathcal{M} \subseteq \mathcal{O}$ is an incoherent set of axioms while any proper subset $\mathcal{M}' \subset \mathcal{M}$ is coherent [15]. The same notion can be applied to the field of alignment debugging where we have to consider sets of correspondences instead of axioms.

Definition 8 (MIPS Alignment). *$\mathcal{M} \subseteq \mathcal{A}$ is a minimal incoherence preserving sub-alignment (MIPS alignment), iff $\mathcal{M}$ is incoherent with respect to $\mathcal{O}_1$ and $\mathcal{O}_2$ and there exists no $\mathcal{M}' \subset \mathcal{M}$ such that $\mathcal{M}'$ is coherent with respect to $\mathcal{O}_1$ and $\mathcal{O}_2$. The collection of all MIPS alignments is referred to as $MIPS_{\mathcal{S}}(\mathcal{A}, \mathcal{O}_1, \mathcal{O}_2)$.*

As already indicated in [11], the problem of debugging an incoherent alignment can be understood in terms of Reiters theory of diagnosis [14]. Reiter describes a diagnostic problem in terms of a system and its components. The need for a diagnosis arises, when the observed system behavior differs from the expected behaviour. According to Reiter, the diagnostic problem is to determine a set of those system components which, when assumed to be functioning abnormally, explain the discrepancy between observed and correct behaviour. If this set of components is minimal, it is referred to as diagnosis Δ. In our context a system is a tuple $\langle \mathcal{A}, \mathcal{O}_1, \mathcal{O}_2, \mathcal{S} \rangle$. The discrepancies between observed and correct behaviour are the terminological entities that were satisfiable in $\mathcal{O}_1$ and $\mathcal{O}_2$ and have become unsatisfiable in $\mathcal{O}_1 \cup_{\mathcal{A}}^{\mathcal{S}} \mathcal{O}_2$. The components of the system are the axioms of $\mathcal{O}_1$ and $\mathcal{O}_2$ as well as the correspondences of $\mathcal{A}$. Nevertheless, with respect to alignment debugging the set of possibly erroneous components is restricted to the correspondences of $\mathcal{A}$. We conclude, that an alignment diagnosis should be defined as a minimal set $\Delta \subseteq \mathcal{A}$ such that $\mathcal{A} \setminus \Delta$ is coherent.

Definition 9 (Alignment Diagnosis). *$\Delta \subseteq \mathcal{A}$ is a diagnosis for $\mathcal{A}$ with respect to $\mathcal{O}_1$ and $\mathcal{O}_2$ iff $\mathcal{A} \setminus \Delta$ is coherent with respect to $\mathcal{O}_1$ and $\mathcal{O}_2$ and for each $\Delta' \subset \Delta$ the alignment $\mathcal{A} \setminus \Delta'$ is incoherent with respect to $\mathcal{O}_1$ and $\mathcal{O}_2$.*

Reiter argues that a diagnosis is a minimal hitting set over the set of all minimal conflict sets. A minimal conflict set in the general theory of diagnosis is equivalent to a MIPS in the context of diagnosing ontology alignments. A diagnosis for an incoherent alignment $\mathcal{A}$ is thus a minimal hitting set for *MIPS*$_{\mathcal{S}}(\mathcal{A}, \mathcal{O}_1, \mathcal{O}_2)$.

Proposition 1 (Diagnosis and Minimal Hitting Set). *Given an alignment $\mathcal{A}$ between ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$. $\Delta \subseteq \mathcal{A}$ is a diagnosis for $\mathcal{A}$ with respect to $\mathcal{O}_1$ and $\mathcal{O}_2$, iff Δ is a minimal hitting set for MIPS$_{\mathcal{S}}(\mathcal{A}, \mathcal{O}_1, \mathcal{O}_2)$.*

Proposition 1 is a special case of corollary 4.5 in [14] where an accordant proof is given. In general there exist many different diagnosis for an incoherent alignment. Reiter proposes the hitting set tree algorithm for enumerating all minimal hitting sets. With respect to our problem we will not be able to compute a complete hitting set tree for large matching problems. Instead of that we focus on a specific type of diagnosis explained by discussing the example alignments $\mathcal{A}_I$ to $\mathcal{A}_{IV}$ depicted in Figure 1.

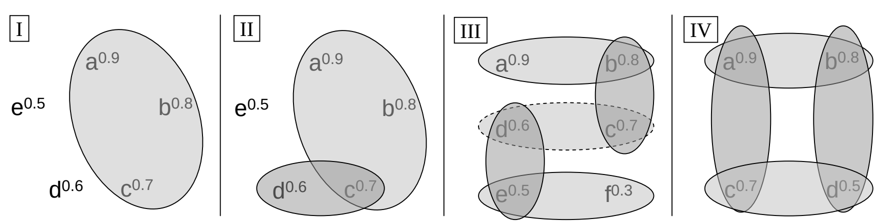

Fig. 1. Four examples for an alignment and its MIPS alignments. Correspondences are denoted by letters $a, b, \ldots$, their confidence values are specified in upper script.

$\mathcal{A}_I$ is an alignment that contains only one MIPS $\mathcal{M} = \{a, b, c\}$. Thus, there are exactly three diagnosis $\{a\}$, $\{b\}$ and $\{c\}$. Taking the confidence values into account, the most reasonable choice for fixing the incoherence is obviously the removal of the 'weakest correspondence' in $\mathcal{M}$, namely $argmin_{x \in \mathcal{M}} conf(x)$. Therefore, we prefer $\Delta = \{c\}$ as diagnosis. Does the naive strategy to remove the correspondence with lowest confidence from each MIPS always result in a diagnosis? $\mathcal{A}_{II}$ disproves this assumption. Following the naive approach we would remove both c and d, although, it is sufficient to remove c. The following recursive definition introduces the notion of an accused correspondence to cope with this problem.

Definition 10 (Accused Correspondence). *A correspondence $c \in \mathcal{A}$ is accused by $\mathcal{A}$ with respect to $\mathcal{O}_1$ and $\mathcal{O}_2$, iff there exists some $\mathcal{M} \in MIPS_{\mathcal{S}}(\mathcal{A}, \mathcal{O}_2, \mathcal{O}_2)$ with $c \in \mathcal{M}$ such that for all $c' \in \mathcal{M} \setminus \{c\}$ it holds that (1) $conf(c') > conf(c)$ and (2) c' is not accused by $\mathcal{A}$ with respect to $\mathcal{O}_1$ and $\mathcal{O}_2$.*

We have chosen the term 'accused correspondence' because the correspondence with lowest confidence in a MIPS alignment $\mathcal{M}$ is 'accused' to cause the problem. This charge will be rebuted if one of the other correspondences in $\mathcal{M}$ is already accused due to the existence of another MIPS alignment. We can apply this definition on the example alignment $\mathcal{A}_{II}$. Correspondence c is an accused correspondence, while correspondence d is not accused due to condidtion (2) in Definition 10. Obviously, the removal of the accused correspondence seems to be the most reasonable decision. In particular, it can be shown by induction that the set of accused correspondences is a diagnosis. Due to the lack of space we have to refer the reader to [9] where an accordant proof is given.

Proposition 2. *The alignment $\Delta \subseteq \mathcal{A}$ which consists of all correspondences accused by $\mathcal{A}$ with respect to $\mathcal{O}_1$ and $\mathcal{O}_2$ is a diagnosis for $\mathcal{A}$ with respect to $\mathcal{O}_1$ and $\mathcal{O}_2$.*

The set of accused correspondences is defined in a way where the whole collection $MIPS_{\mathcal{S}}(\mathcal{A}, \mathcal{O}_1, \mathcal{O}_2)$ is not taken into account from a global point of view. At the same time each removal decision seems to be the optimal choice with respect to the MIPS under discussion. Therefore, it is referred to as local optimal diagnosis in the following.

Definition 11 (Local Optimal Diagnosis). *A diagnosis Δ such that all $c \in \Delta$ are accused by $\mathcal{A}$ with respect to $\mathcal{O}_1$ and $\mathcal{O}_2$ is referred to as local optimal diagnosis.*

For the third alignment depicted in Figure 1 the set $\Delta = \{b, d, f\}$ is a local optimal diagnosis. The effects of a local removal decision can have strong effects on the whole diagnosis. One of the MIPS of $\mathcal{A}'$ is depicted with dashed lines. Suppose that we would not know this MIPS. As a result we would compute $\Delta = \{b, e\}$ as diagnosis. This small example indicates that each decision might have effects on a chain of consequent decisions. Thus, we need to construct an algorithm that is complete with respect to the detection of incoherence, because missing out a reason for incoherence might have significant effects on the whole diagnosis.

We discussed examples where the removal of the accused correspondences is a reasonable choice, nevertheless, it is disputable whether a local optimal diagnosis is the best choice among all diagnosis. Instead of comparing confidences within a MIPS, it is e.g. also possible to aggregate (e.g. sum up) the confidences of Δ as proposed in [7]. In our framework we would refer to such a diagnosis as a global optimal diagnosis. The fourth alignment $\mathcal{A}_{IV}$ is an example where local optimal diagnosis Δ_L and global optimal diagnosis Δ_G differ, in particular we have $\Delta_L = \{b, c\}$ and global optimal diagnosis $\Delta_G = \{a, d\}$. We will see in Section 4 that a local optimal diagnosis can be computed in polynomial time (leaving aside the complexity of the reasoning involved). Opposed to this, we have to solve the weighted variant of the hitting set problem to construct a global optimal solution, which is known to be a NP-complete problem [3]. The experimental results presented in Section 5 will also show that the removal of a local optimal diagnosis has positive effects on the quality of the alignment.

4 Algorithms

A straightforward way to check the coherence of an alignment can be described as follows. We have to iterate over the atomic entities $i\#e_{i\in\{1,2\}}$ of both $\mathcal{O}_1$ and $\mathcal{O}_2$ each time checking whether $i\#e$ is unsatisfiable in $\mathcal{O}_1 \cup^{\mathcal{S}}_{\mathcal{A}} \mathcal{O}_2$ and satisfiable in $\mathcal{O}_i$. The (un)satisfiability of a property $i\#R$ is decided via checking the (un)satisfiability of $\exists i\#R.\top$. Given a coherent alignment $\mathcal{A}$, we have to iterate over all atomic entities to conclude that $\mathcal{A}$ is coherent. If $\mathcal{A}$ is incoherent we can stop until we detect a first unsatisfiable class. Alternatively, we might also completely classify $\mathcal{O}_1 \cup^{\mathcal{S}}_{\mathcal{A}} \mathcal{O}_2$ and ask the reasoner for unsatisfiable classes. In the following we refer to the application of such a strategy by the procedure call ISCOHERENTALIGNMENT($\mathcal{A}$, $\mathcal{O}_1$, $\mathcal{O}_2$).

There exists an approach to decide the coherence *for most dual-element alignments* which outperforms ISCOHERENTALIGNMENT by far. This approach and its application requires to introduce the notion of a conflict pair. A conflict pair is an incoherent subset of an alignment that contains exactly two correspondences. Moreover, it turns out that most elements in $\mathit{MIPS}_{\mathcal{S}_n}(\mathcal{A}, \mathcal{O}_1, \mathcal{O}_2)$ are conflict pairs of a certain type. We believe that there exists a pattern based reasoning method for each alignment semantics that detects a (large) fraction of all conflict pairs within an alignment. We present such a reasoning method for the natural semantics $\mathcal{S}_n$ and argue finally how to develop a similar method for other alignment semantics using the example of DDL.

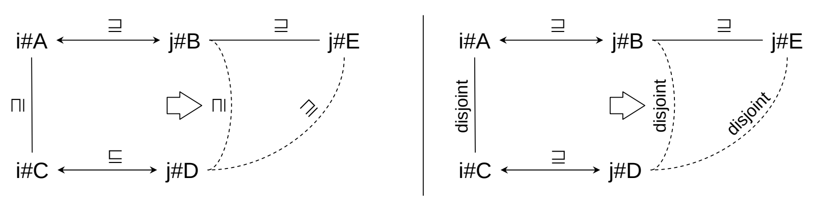

Fig. 2. Subsumption and disjointness propagation pattern. Arrows represent correspondences, solid lines represent axioms or entailed statements in $\mathcal{O}_i$ resp. $\mathcal{O}_j$, and dashed lines represent statements entailed by the merged ontology. Figure taken from [10], where these patterns have been used to support manual mapping revision.

First, we focus on the pattern depicted on the left of Figure 2. Given correspondences $\langle i\#A, j\#B, \overset{\leftrightarrow}{\sqsupseteq}, n\rangle$ and $\langle i\#C, j\#D, \overset{\leftrightarrow}{\sqsubseteq}, n'\rangle$ as well as axiom $i\#A \sqsubseteq i\#C$ we can conclude that $\mathcal{O}_i \cup^{\mathcal{S}_n}_{\mathcal{A}} \mathcal{O}_j \models j\#B \sqsubseteq j\#D$ and thus $\mathcal{O}_i \cup^{\mathcal{S}_n}_{\mathcal{A}} \mathcal{O}_j \models j\#E \sqsubseteq j\#D$ for each subconcept $j\#E$ of $j\#B$. Now we have $\mathcal{O}_i \cup_{\mathcal{A}}^{\mathcal{S}_n} \mathcal{O}_j \models \bot \sqsupseteq j\#E$ whenever $\mathcal{O}_j$ entails the disjointness of $j\#E$ and $j\#D$. In such a case we detected a conflict pair given the satisfiability of $j\#E$ in $\mathcal{O}_j$. The disjointness propagation pattern works similar. We abstain from a detailed description and refer the reader to the presentation in Figure 2. If we combine both patterns and check their occurrence in all possible combinations given a pair of correspondences, we end up with a sound but incomplete algorithm for deciding the incoherence of an alignment that contains exactly two correspondences. We will refer to this algorithm as POSSIBLYCOHERENT($c_1, c_2, \mathcal{O}_i, \mathcal{O}_j$) with $c_1, c_2 \in \mathcal{A}$. $\mathcal{S}_n$ might in general induce complex interdependences between $\mathcal{A}$, $\mathcal{O}_1$ and $\mathcal{O}_2$. Therefore, neither are all conflict pairs detectable by the pattern-based approach, nor are all MIPS conflict pairs.

We extend our algorithms (respectively the described pattern) to correspondences between properties by replacing $i\#A$ by $\exists i\#A.\top$ in case that $i\#A$ is a property (the same for $i\#B$, $i\#C$, and $i\#D$). This allows us to consider dependencies between domain restrictions and the subsumption hierarchy within our pattern based reasoning approach. The patterns depicted in Figure 2 are specific to the natural semantics $\mathcal{S}_n$. Similar patterns very likely exist for any reductionistic alignment semantics. For DLL e.g. it is possible to construct corresponding patterns easily. The subsumption propagation pattern is a specific case of (and in particular inspired by) the general propagation rule used within the tableau algorithm proposed in [16], while the disjointness propagation pattern does not hold in DDL.

In the following we need to enumerate the correspondences of an alignment to access elements or subsets of the alignment by index or range. Thus, we sometimes treat an alignment $\mathcal{A}$ as a field using a notation $\mathcal{A}[i]$ to refer to the i-th element of $\mathcal{A}$ and $\mathcal{A}[j \dots k]$ to refer to $\{\mathcal{A}[i] \in \mathcal{A} \mid j \leq i \leq k\}$. For the sake of convenience we use $\mathcal{A}[\dots k]$ to refer to $\mathcal{A}[0 \dots k]$, similar we use $\mathcal{A}[j \dots]$ to refer to $\mathcal{A}[j \dots |A| - 1]$. Further, let the index of an alignment start at 0.

Algorithm 1

BRUTEFORCELOD($\mathcal{A}$, $\mathcal{O}_1$, $\mathcal{O}_2$)

if ISCOHERENTALIGNMENT($\mathcal{A}$, $\mathcal{O}_1$, $\mathcal{O}_2$) **then**
    return $\emptyset$
else
    $\triangleright$ *sort $\mathcal{A}$ descending according to confidence values*
    $\mathcal{A}' \leftarrow \emptyset$
    for all $c \in \mathcal{A}$ **do**
        if ISCOHERENTALIGNMENT($\mathcal{A}' \cup \{c\}$, $\mathcal{O}_1$, $\mathcal{O}_2$) **then**
            $\mathcal{A}' \leftarrow \mathcal{A}' \cup \{c\}$
        end if
    end for
    return $\mathcal{A} \setminus \mathcal{A}'$
end if

We already argued that the set of accused correspondences forms a special kind of diagnosis referred to as local optimal diagnosis. Algorithm 1, which has been proposed in [12], is an iterative procedure that computes such a diagnosis. First, we check the coherence of $\mathcal{A}$ and return $\emptyset$ as diagnosis for a coherent alignment. Given $\mathcal{A}$'s incoherence, we have to order $\mathcal{A}$ by descending confidence values. Then an empty alignment $\mathcal{A}'$ is step by step extended by adding correspondences $c \in \mathcal{A}$. Whenever $\mathcal{A}' \cup c$ becomes incoherent, which is decided by reasoning in the merged ontology, c is not added. Finally, we end up with a local optimal diagnosis $\mathcal{A} \setminus \mathcal{A}'$.

Proposition 3. BRUTEFORCELOD*($\mathcal{A}$, $\mathcal{O}_1$, $\mathcal{O}_2$) is a local optimal diagnosis for $\mathcal{A}$ with respect to $\mathcal{O}_1$ and $\mathcal{O}_2$.*

Algorithm 1 is completely built on reasoning in the merged ontology and does not exploit efficient reasoning techniques. A more efficient algorithm requires to solve the following problem. Given an incoherent alignment $\mathcal{A}$ ordered descending according to its confidences, we want to find an index i such that $\mathcal{A}[\ldots i-1]$ is coherent and $\mathcal{A}[\ldots i]$ is incoherent. Obviously, a binary search can be used to detect this index. The accordant algorithm, referred to as SEARCHINDEXOFACCUSEDCORRESPONDENCE($\mathcal{A}$, $\mathcal{O}_1$, $\mathcal{O}_2$), starts with an index m that splits the incoherent alignment $\mathcal{A}$ in two parts of equal size. Let now i be the index we are searching for. If $\mathcal{A}[\ldots m]$ is coherent we know that $i > m$, otherwise $i \leq m$. Based on this observation we can start a binary search which finally requires $log_2(|\mathcal{A}|)$ iterations to terminate.

Algorithm 2

EFFICIENTLOD($\mathcal{A}$, $\mathcal{O}_1$, $\mathcal{O}_2$)

$\triangleright$ *sort $\mathcal{A}$ descending according to confidence values*
$\mathcal{A}' \leftarrow \mathcal{A}$, $k \leftarrow 0$
loop
  for $i \leftarrow k$ to $|\mathcal{A}'| - 1$ **do**
    for $j \leftarrow 0$ to $i - 1$ **do**
      if not POSSIBLYCOHERENT($\mathcal{A}'[j]$, $\mathcal{A}'[i]$, $\mathcal{O}_1$, $\mathcal{O}_2$) **then**
        $\mathcal{A}' \leftarrow \mathcal{A}' \setminus \{\mathcal{A}'[i]\}$
        $i \leftarrow i - 1$ $\triangleright$ *adjust i to continue with next element of $\mathcal{A}'$*
        break $\triangleright$ *exit inner for-loop*
      end if
    end for
  end for
  $k \leftarrow$ SEARCHINDEXOFACCUSEDCORRESPONDENCE($\mathcal{A}'$, $\mathcal{O}_1$, $\mathcal{O}_2$)
  if $k =$ NIL **then**
    return $\mathcal{A} \setminus \mathcal{A}'$
  end if
  $\triangleright$ *let k^* be the counterpart of k adjusted for $\mathcal{A}$ such that $\mathcal{A}[k^*] = \mathcal{A}'[k]$*
  $\mathcal{A}' \leftarrow \mathcal{A}'[\ldots k-1] \cup \mathcal{A}[k^* + 1 \ldots]$
end loop

We are now prepared to construct an efficient algorithm to compute a local optimal diagnosis (LOD) (Algorithm 2). First we have to sort the input alignment $\mathcal{A}$, prepare a copy $\mathcal{A}'$ of $\mathcal{A}$, and init an index $k = 0$. Variable k works as a separator between the part of $\mathcal{A}'$ that has already been processed successfully and the part of $\mathcal{A}'$ that has not yet been processed or has not been processed successfully. More precisely, it holds that $\mathcal{A}[\ldots k^*] \setminus \mathcal{A}'[\ldots k]$ is a LOD for $\mathcal{A}[\ldots k^*]$ where k^* is an index such that $\mathcal{A}'[k] = \mathcal{A}[k^*]$. Within the main loop we have two nested loops. These are used to check whether correspondence $\mathcal{A}'[i]_{i \geq k}$ possibly conflicts with one of $\mathcal{A}'[j]_{j<i}$. In case a conflict has been detected, $\mathcal{A}'[i]$ is removed from $\mathcal{A}'$. Notice that this approach would directly result in a LOD if both (1) all $\mathcal{M} \in$ *MIPS*$_{\mathcal{S}}$ $(\mathcal{A}, \mathcal{O}_1, \mathcal{O}_2)$ were conflict pairs, and all conflict pairs were detectable by procedure POSSIBLYCOHERENT. Obviously, these assumptions are not correct and thus we have to search for an index k such that $\mathcal{A}[\ldots k^*] \setminus \mathcal{A}'[\ldots k]$ is a LOD for $\mathcal{A}[\ldots k^*]$. Index k is determined by the binary search presented above. If no such index could be detected, we know that $\mathcal{A} \setminus \mathcal{A}'$ is a LOD (line 14-16). Otherwise, the value of $\mathcal{A}'$ is readjusted to the union of $\mathcal{A}'[\ldots k-1]$, which can be understood as the validated part of $\mathcal{A}'$, and $\mathcal{A}[k^*+1\ldots]$, which is the remaining part of $\mathcal{A}$ to be processed in the next iteration. $\mathcal{A}'[k]$ is removed from $\mathcal{A}'$ and thus becomes a part of the diagnosis returned finally.

Proposition 4. EFFICIENTLOD*($\mathcal{A}$, $\mathcal{O}_1$, $\mathcal{O}_2$) is a local optimal diagnosis for $\mathcal{A}$ with respect to $\mathcal{O}_1$ and $\mathcal{O}_2$.*

Suppose now that Δ' is a LOD for a subset of *MIPS*$_{\mathcal{S}}$ $(\mathcal{A}, \mathcal{O}_1, \mathcal{O}_2)$, namely those that are detected by our pattern based reasoning approach, while Δ is the LOD for the complete set *MIPS*$_{\mathcal{S}}$ $(\mathcal{A}, \mathcal{O}_1, \mathcal{O}_2)$. The correctness of Proposition 4 is based on the fact, that Δ' can be split in a correct and an incorrect part. The correspondence where the correct part ends is exactly the correspondences that is detected by the binary search. Due to the stable ordering, the correct part can be extended over several iterations until we finally end up with a complete and correct local optimal diagnosis Δ.

5 Experiments

Our experiments are based on datasets used within two subtracks of the Ontology Alignment Evaluation Initiative (OAEI). These tracks are the benchmark track about the domain of publications and the conference track. In opposite to the other OAEI tracks, the reference alignments of these tracks are open available.

The benchmark dataset consists of an ontology #101 and alignments to a set of artificial variations #1xx to #2xx. Furthermore, there are reference alignments to four real ontologies known as #301 to #304. We have chosen these four ontologies for our experiments to avoid any interdependencies between the specifics of the artificial test sets and our approach. For our experimental study we had to apply some minor modifications. Neither ontology #101 nor ontologies #301 to #304 contain disjointness axioms; even a highly incorrect alignment cannot introduce any incoherences. Therefore, we decided to extend ontology #101 by disjointness axioms between sibling classes. In the 2008 evaluation 8 matching systems submitted results to the benchmark track that were annotated with confidence values. In the following we refer to this dataset as B^d_{08}.

Our second dataset is based on the conference dataset. In 2008 for the first time reference alignments between five ontologies (= 10 alignments) have been used as part of the official OAEI evaluation. We had to reduce this set four ontologies (= 6 alignments), since one ontology, namely the IASTED ontology, resulted in reasoning problems when merging this ontology with one of the other ontologies. In particular, the runtime behaviour of our algorithms was strongly affected by underlying reasoning problems with IASTED. Unfortunately, only three systems participated in the conference track in 2008, only two of them distinguishing between different degrees of confidence. Therefore, we also used the submissions to the 2007 campaign were we also had two matching systems producing meaningful confidence values. We refer to the resulting dataset as C_{07}, respectively C_{08}. Disjointness is modeled in this dataset incompletely depending on the specific ontology. Thus, we decided to apply our approach to the official OAEI dataset as well as to a dataset enriched with obvious disjointness statements between sibling concepts. These disjointness statements have been manually added as part of the work reported in [12]. The resulting datasets are referred to as C^d_{07}, respectively C^d_{08}.

In our experiments we used the reasoner Pellet [17], in particular version 2RC2 together with the OWL API on a 2.26 GHz standard laptop with 2GB RAM. The complete dataset as well as a more detailed presentation of the results is available at `http://webrum.uni-mannheim.de/math/lski/matching/lod/`. Due to the lack of space we can only present aggregated results in the following paragraphs.

Runtimes Results related to runtime efficiency are presented in Table 1. In each row we aggregated the results of a specific matcher for one of the datasets explained above. For both Algorithm 2 and its brute-force counterpart Algorithm 1 the total of runtimes is displayed in milliseconds. Obviously, Algorithm 2 outperforms the brute force approach. Runtime performance increased by a coefficient of 1.8 to 9.3. To better understand under which circumstances Algorithm 2 performs better, we added columns presenting the size of the input alignment $\mathcal{A}$, the size of the debugged alignment $\mathcal{A}'$, and the size of the diagnosis $\Delta = \mathcal{A} \setminus \mathcal{A}'$. Furthermore, the column captioned with '$k \neq$ NIL' refers to the number of correspondences that have additionally been detected due to complete reasoning techniques. In particular, it displays how often $k \neq$ NIL is evaluated as true in line 14 of Algorithm 2. Finally, we analyze the fraction of those correspondences that have been detected by efficient reasoning techniques.

Although we observe that absolute runtimes are affected by the alignment size (see for example the C_{07}-OLA row), the coefficient of runtimes seems not to be affected directly. The same holds for the size of the diagnosis Δ. Instead of that and in accordance to our theoretical considerations the runtime coefficient correlates with the fraction of conflicts that can be detected efficiently. While for the conference testcases results have to be considered inconclusive, this pattern clearly emerges for the benchmark testcases. The efficiency of Algorithm 2 is thus directly affected by the degree of completeness of POSSIBLYCOHERENT invoked as subprocedure.

Diagnostic Quality In previous work we already argued that the coherence of an alignment is a quality of its own [8]. An incoherent alignment causes specific problems depending on the scenario in which the alignment is used. We now additionally investigate in how far the removal of the diagnosis increases the quality of the input alignment $\mathcal{A}$ by comparing it against reference alignment $\mathcal{R}$. In particular, we compute for both

Testcase		Runtime Comparison			Alignment Size & Deleted Correspondences				
DS	Matcher	Alg.2	Alg.1	Coeff.	$\|\mathcal{A}\|$	$\|\mathcal{A}'\|$	$\|\Delta\|$	'$k \neq$ NIL'	Frac.
B^d_{08}	Aroma	8656	71846	8.3	202	194	8	2	75%
	ASMOV	6226	47714	7.7	222	218	4	1	75%
	CIDER	7530	70028	9.3	195	181	14	1	93%
	DSSim	3922	36343	9.3	184	179	5	0	100%
	Lily	15468	74352	4.8	218	210	8	4	50%
	RiMOM	15942	78219	4.9	235	221	14	5	64%
	SAMBO	3800	28655	7.5	197	196	1	0	100%
	SAMBOdtf	8586	59211	6.9	206	202	4	2	50%
C_{07}	Falcon	6847	12414	1.8	70	56	14	4	71%
	OLA	39830	73497	1.8	404	228	176	27	85%
C_{08}	ASMOV	13289	23425	1.8	153	128	25	10	60%
	Lily	4604	14609	3.2	78	63	15	3	80%

Table 1. Aggregated Runtime of EFFICIENTLOD-Algorithm (Alg.2) and BRUTEFORCELOD-Algorithm (Alg.1) and related characteristics.

the input alignment $\mathcal{A}$ and the repaired alignment $\mathcal{A}' = \mathcal{A} \setminus \Delta$ the classical measures of precision and recall. The precision of an alignment describes its degree of correctness, while recall describes its degree of completeness. A definition of these measures with respect to alignment evaluation can be found in [2].

The results of our measurements are presented in Table 2. The first two columns identify datasets, followed by columns presenting the size of the input alignment $\mathcal{A}$, the size of the diagnosis $\Delta = \mathcal{A} \setminus \mathcal{A}'$, and the number of removed correspondences $\Delta \setminus \mathcal{R}$ that are actually incorrect i.e. those correspondences that have been removed correctly. The following three columns show how precision, recall and f-measure have been affected by the application of our algorithm. In the *Effect* column the results are aggregated as difference between the f-measure of the input alignment $\mathcal{A}$ and the f-measure of the repaired alignment $\mathcal{A}'$.

Based on the f-measure differences we conclude that in 13 of 16 testcases we increased the overall quality of the alignment. Notice again that these results are aggregated average values. Taking a closer look at the individual results for each generated alignment (not depicted in Table 2), we observe that in 15 cases our approach has negative effects on the f-measure, in 14 cases we observed no effects at all, and in 51 cases we measured an increased f-measure. Obviously, this effect is based on an increased precision and a stable or only slightly decreased recall. Nevertheless, there are some exceptions to this pattern.

On the one hand we have negative results for B^d_{08}-DSSim, C_{08}-ASMOV and C^d_{08}-ASMOV. Due to characteristics of a local optimal diagnosis an incorrect correspondence might cause the removal of all conflicting correspondences with lower confidence given that there exists no conflicting correspondence with higher confidence. An analysis of the individual results revealed that the negative effects are based on this pattern, i.e. an incorrect correspondence has been annotated with very high confidence and no 'antagonist' has been annotated with higher confidence.

DS	Matcher	$\|\mathcal{A}\|$	$\|\Delta\|$	$\|\Delta \setminus \mathcal{R}\|$	Prec. $\mathcal{A} \rightsquigarrow \mathcal{A}'$	Rec. $\mathcal{A} \rightsquigarrow \mathcal{A}'$	F-m. $\mathcal{A} \rightsquigarrow \mathcal{A}'$	Effect
B^d_{08}	Aroma	202	8	7	80.2 ⇝ 83.0	70.1 ⇝ 69.7	74.8 ⇝ 75.8	+0.9
	ASMOV	222	4	3	78.4 ⇝ 79.4	75.3 ⇝ 74.9	76.8 ⇝ 77.1	+0.2
	CIDER	195	14	5	87.2 ⇝ 89.0	73.6 ⇝ 69.7	79.8 ⇝ 78.2	-1.7
	DSSim	184	5	5	87.5 ⇝ 89.9	69.7 ⇝ 69.7	77.6 ⇝ 78.5	+0.9
	Lily	218	8	8	83.0 ⇝ 86.2	78.4 ⇝ 78.4	80.6 ⇝ 82.1	+1.5
	RiMOM	235	14	14	78.3 ⇝ 83.3	79.7 ⇝ 79.7	79.0 ⇝ 81.4	+2.4
	SAMBO	197	1	1	91.9 ⇝ 92.3	78.4 ⇝ 78.4	84.6 ⇝ 84.8	+0.2
	SAMBOdtf	206	4	4	88.3 ⇝ 90.1	78.8 ⇝ 78.8	83.3 ⇝ 84.1	+0.8
C_{07}	Falcon	70	14	11	65.7 ⇝ 76.8	60.5 ⇝ 56.6	63.0 ⇝ 65.2	+2.1
	OLA	404	176	174	12.4 ⇝ 21.1	65.8 ⇝ 63.2	20.8 ⇝ 31.6	+10.7
C_{08}	ASMOV	153	25	20	22.9 ⇝ 23.4	46.1 ⇝ 39.5	30.6 ⇝ 29.4	-1.2
	Lily	78	15	13	44.9 ⇝ 52.4	46.1 ⇝ 43.4	45.5 ⇝ 47.5	+2.0
C^d_{07}	Falcon	70	17	14	65.7 ⇝ 81.1	60.5 ⇝ 56.6	63.0 ⇝ 66.7	+3.7
	OLA	404	228	226	12.4 ⇝ 27.3	65.8 ⇝ 63.2	20.8 ⇝ 38.1	+17.3
C^d_{08}	ASMOV	153	33	27	22.9 ⇝ 24.2	46.1 ⇝ 38.2	30.6 ⇝ 29.6	-1.0
	Lily	78	21	17	44.9 ⇝ 54.4	46.1 ⇝ 40.8	45.5 ⇝ 46.6	+1.2

Table 2. Alignment size, size of diagnosis and number of correctly removed correspondences; effects on precision, recall, and f-measure.

On the other hand we measured strong positive effects for the OLA system on the conference dataset. These effects are associated with the large size of the alignments generated by OLA. It seems that, compared to the other submissions, the matching results of OLA have not been filtered or thresholded in an appropriate way. OLA generated a total of 404 correspondences with respect to our C datasets. For the original dataset C (no disjointness axioms added) 176 of these correspondences have been automatically removed by our approach and only 2 of these removals were incorrect, which raised the f-measure from 20.8% to 31.6% (from 20.8% to 38.1% for the C^d dataset). Notice that our algorithm expects no parameter which corresponds to a threshold or an estimated size of the reference alignment. Instead of that the algorithm automatically adapts to the quality of the input due to the fact that a highly incorrect alignment will be higly incoherent. Overall, the results indicate that our approach does not only ensure the quality of the input alignment but even more has significant positive effects.

Related work In [13] Qi et. al. propose a kernel revision operator for description logic-based ontologies. A revision deals with the problem of incorporating newly received information into accepted information consistently. Within their experiments the authors apply their approach amongst others to the revision of ontology alignments, where the matched ontologies are accepted information and the alignment between them is new and disputable information. Two of the algorithms proposed require to compute all $MIPS_{\mathcal{S}}(\mathcal{A}, \mathcal{O}_1, \mathcal{O}_2)$ in order to construct a minimal hitting set, while their third and most efficient algorithm cannot ensure the minimality of the constructed hitting set. We conducted additional experiments with the alignments used in [13]. We did not include these as part of the main experiments, because the datasets do not contain correspondences between properties and are not as comprehensive as the datasets used within our experiments. However, we observed runtimes between 50 and 250 milliseconds, while in [13] runtimes between 6 and 51 seconds have been reported for the fastest algorithm.

An approach, which aims to explain logical consequences of an alignment, has been proposed in [6]. Some of these consequences are unintended due to incorrect correspondences in $\mathcal{A}$ and cannot be accepted. An example of an unintended consequence is a concept becoming unsatisfiable due to $\mathcal{A}$. Such an alignment is referred to as incoherent within our framework. To generate plans for repairing a defect alignment, first, all justifications for the unintended consequences are computed. While in [13] all MIPS are used to compute a minimal hitting set, in [6] all justifications are used to compute minimal hitting sets referred to as a repair plans. The authors point out, that the bottleneck of their approach is the computation of all justifications.

In summary, both approaches suffer from the incorrect assumption that a minimal hitting set can only be constructed given complete knowledge about all MIPS respectively all justifications. Contrary to this, we have shown that it is possible to compute a specific hitting set, namely a local optimal diagnosis, that is not only minimal but also takes into account confidence values in an appropriate manner.

6 Conclusion

We have presented a basic algorithm for computing a local optimal diagnosis as well as an efficient variant, which makes use of an intertwined combination of incomplete and complete reasoning techniques. These algorithms are based on precise logic-based semantics of an alignment. Although, we only focused on specific type of semantics, namely the natural semantics, there is some evidence that the principles of our approach can be applied to each reductionistic alignment semantics.

It turned out that the efficient variant of our algorithm outperformed the basic algorithm by a factor of ≈ 2 to 10. In particular, we observed that the runtime is first and foremost determined by the fraction of conflicts detectable by the incomplete reasoning procedures. In future work we will add additional reasoning patterns in order to detect more conflicts by efficient reasoning strategies.

Our algorithm improves in most cases an alignments f-measure due to an increased precision. However, we detected some outliers where a highly confident but incorrect correspondence had negative impact on the repairing process. An approach that removes a minimum number of correspondences would probably remove such a correspondence. Generally, it is not clear whether the *principle of minimal change* is a good guideline for repairing alignments. Experiments we conducted so far show inconclusive results and require additional analysis.

We already pointed to some problems of other approaches. We believe that these problems are based on not taking into account three specifics of the problem under discussion. First, correspondences are annotated with confidence values. Second, there are significantly less correspondences in an alignment than axioms in the matched ontologies. Third, given the monotonicity of $\mathcal{S}$, everything that holds in $\mathcal{O}_1$ and $\mathcal{O}_2$ holds also in the merged ontology $\mathcal{O}_1 \cup_{\mathcal{A}}^{\mathcal{S}} \mathcal{O}_2$. The first observation was taken into account in the definition of a local optimal diagnosis, the second observation points to the possibility of iterating over all correspondences (the main loop in both algorithms), and the third observation is exploited within the combination of pattern-based reasoning and reasoning in the merged ontology.

Acknowledgement The work has been partially supported by the German Science Foundation (DFG) under contract STU 266/3-1 and STU 266/5-1.

References

1. Alex Borgida and Luciano Serafini. Distributed description logics: Assimilating information from peer sources. *Journal on Data Semantics*, 2003.
2. Jerome Euzenat and Pavel Shvaiko. *Ontology Matching*. Springer, 2007.
3. Michael R. Garey and David S. Johnson. *Computers and intractability: A guide to the theory of NP-completeness*. W. H. Freeman and Company, 1979.
4. Fausto Giunchiglia, Mikalai Yatskevich, and Pavel Shvaiko. Semantic matching: Algorithms and implementation. *Journal on Data Semantics*, 2007.
5. Yves R. Jean-Mary and Mansur R. Kabuka. Asmov: Results for OAEI 2008. In *Proc. of the ISWC 2008 workshop on ontology matching*, Karlsruhe, Germany, 2008.
6. Ernesto Jimenez-Ruiz, Bernardo Cuenca Grau, Ian Horrocks, and Rafael Berlanga. Ontology integration using mappings: Towards getting the right logical consequences. In *Proc. of the 6th Annual European Semantic Web Conference*, Heraklion, Crete, Greece, 2009.
7. Christian Meilicke and Heiner Stuckenschmidt. Applying logical constraints to ontology matching. In *Proc. of the 30th German Conference on Artificial Intelligence*, Osnabrück, Germany, 2007.
8. Christian Meilicke and Heiner Stuckenschmidt. Incoherence as a basis for measuring the quality of ontology mappings. In *Proc. of the ISWC 2008 Workshop on Ontology Matching*, Karlsruhe, Germany, 2008.
9. Christian Meilicke and Heiner Stuckenschmidt. An efficient method for computing a local optimal alignment diagnosis. Technical report, University Mannheim, Computer Science Institute, 2009.
10. Christian Meilicke, Heiner Stuckenschmidt, and Ondrej Svab-Zamazal. A reasoning-based support tool for ontology mapping evaluation. In *Proc. of the European Semantic Web Conference*, Heraklion, Greece, 2009.
11. Christian Meilicke, Andrei Tamilin, and Heiner Stuckenschmidt. Repairing ontology mappings. In *Proc. of the Twenty-Second Conference on Artificial Intelligence*, Vancouver, Canada, 2007.
12. Christian Meilicke, Johanna Völker, and Heiner Stuckenschmidt. Learning disjointness for debugging mappings between lightweight ontologies. In *Proc. of the 16th International Conference on Knowledge Engineering and Knowledge Management*, Acitrezza, Italy, 2008.
13. Guilin Qi, Peter Haase, Zhisheng Huang, Qiu Ji, Jeff Z. Pan, and Johanna Völker. A kernel revision operator for terminologies - algorithms and evaluation. In *Proc. of the International Semantic Web Conference*, Karlsruhe, Germany, 2008.
14. Raymond Reiter. A theory of diagnosis from first principles. *Artificial Intelligence*, 1987.
15. Stefan Schlobach and Ronald Cornet. Non-standard reasoning services for the debugging of description logic terminologies. In *Proc. of 18th International Joint Conference on Artificial Intelligence*, Acapulco, Mexico, 2003.
16. Luciano Serafini and Andrei Tamilin. Local tableaux for reasoning in distributed description logics. In *Proc. of the Int. Workshop on Description Logics*, Whistler, Canada, 2004.
17. Evren Sirin, Bijan Parsia, Bernardo Cuenca Grau, Aditya Kalyanpur, and Yarden Katz. Pellet: a practical OWL-DL reasoner. *Journal of Web Semantics*, 2007.