# Detecting and Correcting Conservativity Principle Violations in Ontology-to-Ontology Mappings

Alessandro Solimando[1], Ernesto Jiménez-Ruiz[2], and Giovanna Guerrini[1]

[1] Dipartimento di Informatica, Università di Genova, Italy
[2] Department of Computer Science, University of Oxford, UK

**Abstract.** In order to enable interoperability between ontology-based systems, ontology matching techniques have been proposed. However, when the generated mappings suffer from logical flaws, their usefulness may be diminished. In this paper we present an approximate method to detect and correct violations to the so-called conservativity principle where novel subsumption entailments between named concepts in one of the input ontologies are considered as unwanted. We show that this is indeed the case in our application domain based on the EU Optique project. Additionally, our extensive evaluation conducted with both the Optique use case and the data sets from the Ontology Alignment Evaluation Initiative (OAEI) suggests that our method is both useful and feasible in practice.

## 1 Introduction

Ontologies play a key role in the development of the Semantic Web and are being used in many diverse application domains, ranging from biomedicine to energy industry. An application domain may have been modeled with different points of view and purposes. This situation usually leads to the development of different ontologies that intuitively overlap, but they use different naming and modeling conventions.

In particular, this is the case we are facing in the EU Optique project.[3] Optique aims at facilitating scalable end-user access to big data in the oil and gas industry. The project is focused around two demanding use cases provided by *Siemens* and *Statoil*. Optique advocates for an Ontology Based Data Access (OBDA) approach [24] so that end-users formulate queries using the vocabulary of a domain ontology instead of composing queries directly against the database. Ontology-based queries (*e.g.*, SPARQL) are then automatically rewritten to SQL and executed over the database.

In Optique two independently developed ontologies co-exist. The first ontology has been directly bootstrapped from one of the relational databases in Optique and it is linked to the database via *direct ontology-to-database mappings*;[4] while the second ontology is a domain ontology based on the Norwegian Petroleum Directorate (NPD) FactPages[5] [41] and it is currently preferred by Optique end-users to feed the visual query formulation interface[6] [42]. This setting requires the "query formulation" ontology to be linked to the relational database. In Optique we follow two approaches that

[3] `http://www.optique-project.eu/`
[4] `http://www.w3.org/TR/rdb-direct-mapping/`
[5] `http://factpages.npd.no/factpages/`
[6] The query formulation interface has been evaluated with end-users at Statoil.

will complement each other: *(i)* creation of ontology-to-database mappings between the query formulation ontology and the database; *(ii)* creation of *ontology-to-ontology* mappings between the bootstrapped ontology and the query formulation ontology. In this paper we only deal with ontology-to-ontology mappings (or mappings for short). The creation, analysis and evolution of ontology-to-database mappings are also key research topics within Optique, however, they fall out of the scope of this paper.

The problem of (semi-)automatically computing mappings between independently developed ontologies is usually referred to as the *ontology matching problem*. A number of sophisticated ontology matching systems have been developed in the last years [11, 40]. Ontology matching systems, however, rely on lexical and structural heuristics and the integration of the input ontologies and the mappings may lead to many undesired logical consequences. In [19] three principles were proposed to minimize the number of potentially unintended consequences, namely: *(i) consistency principle*, the mappings should not lead to unsatisfiable classes in the integrated ontology, *(ii) locality principle*, the mappings should link entities that have similar *neighbourhoods*, *(iii) conservativity principle*, the mappings should not introduce new semantic relationships between concepts from one of the input ontologies.

The occurrence of these violations is frequent, even in the reference mapping sets of the Ontology Alignment Evaluation Initiative[7] (*OAEI*). Also manually curated alignments, such as *UMLS-Metathesaurus* [3] (*UMLS*), a comprehensive effort for integrating biomedical knowledge bases, suffer from these violations. Violations to these principles may hinder the usefulness of ontology mappings. In particular, in the Optique's scenario, violation of the consistency or conservativity principles will directly affect the quality of the query results, since queries will be rewritten according to the ontology axioms, the ontology-to-ontology mappings and the ontology-to-database mappings.

These principles has been actively investigated in the last years (*e.g.*, [31, 30, 15, 19, 17, 29, 37]). In this paper we focus on the conservativity principle and we explore a variant of violation of this principle which we consider appropriate for the application domain in Optique. Furthermore, this variant of the conservativity principle allows us to reduce the problem to a consistency principle problem. We have implemented a method which relies on the projection of the input ontologies to Horn propositional logic. This projection allows us to be efficient in both the reduction to the consistency principle and the subsequent repair process. Our evaluation suggests that our method is feasible even with the largest test cases of the OAEI campaign.

The remainder of the paper is organised as follows. Section 2 summarises the basics concepts and definitions we will rely on along the paper. In Section 3 we introduce our motivating scenario based on Optique. Section 4 describes our method. In Section 5 we present the conducted evaluation. A comparison with relevant related work is provided in Section 6. Finally, Section 7 gives some conclusions and future work lines.

## 2 Preliminaries

In this section, we present the formal representation of ontology mappings and the notions of semantic difference, mapping coherence and conservativity principle violation.

[7] `http://oaei.ontologymatching.org/`

### 2.1 Representation of Ontology Mappings

Mappings are conceptualised as 5-tuples of the form $\langle id, e_1, e_2, n, \rho \rangle$, with $id$ a unique identifier, $e_1, e_2$ entities in the vocabulary or signature of the relevant input ontologies (*i.e.*, $e_1 \in \mathsf{Sig}(\mathcal{O}_1)$ and $e_2 \in \mathsf{Sig}(\mathcal{O}_2)$), $n$ a confidence measure between 0 and 1, and $\rho$ a relation between $e_1$ and $e_2$, typically subsumption, equivalence or disjointness [10].

RDF Alignment [8] is the main format used in the OAEI campaign to represent mappings containing the aforementioned elements. Additionally, mappings are also represented as OWL 2 subclass, equivalence, and disjointness axioms [6]; mapping identifiers ($id$) and confidence values ($n$) are then represented as axiom annotations. Such a representation enables the reuse of the extensive range of OWL 2 reasoning infrastructure that is currently available. Note that alternative formal semantics for ontology mappings have been proposed in the literature (*e.g.*, [4]).

### 2.2 Semantic Consequences of the Integration

The ontology resulting from the integration of two ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$ via a set of mappings $\mathcal{M}$ may entail axioms that do not follow from $\mathcal{O}_1$, $\mathcal{O}_2$, or $\mathcal{M}$ alone. These new semantic consequences can be captured by the notion of *deductive difference* [25].

Intuitively, the deductive difference between $\mathcal{O}$ and $\mathcal{O}'$ w.r.t. a signature $\Sigma$ (*i.e.*, set of entities) is the set of entailments constructed over $\Sigma$ that do not hold in $\mathcal{O}$, but do hold in $\mathcal{O}'$. The notion of deductive difference, however, has several drawbacks in practice. First, there is no algorithm for computing the deductive difference in expressive DLs [25]. Second, the number of entailments in the difference can be infinite.

**Definition 1 (Approximation of the Deductive Difference).** *Let $A, B$ be atomic concepts (including $\top, \bot$), $\Sigma$ be a signature, $\mathcal{O}$ and $\mathcal{O}'$ be two OWL 2 ontologies. We define the approximation of the $\Sigma$-deductive difference between $\mathcal{O}$ and $\mathcal{O}'$ (denoted $\mathsf{diff}^{\approx}_{\Sigma}(\mathcal{O}, \mathcal{O}')$ as the set of axioms of the form $A \sqsubseteq B$ satisfying: (i) $A, B \in \Sigma$, (ii) $\mathcal{O} \not\models A \sqsubseteq B$, and (iii) $\mathcal{O}' \models A \sqsubseteq B$.*

In order to avoid the drawbacks of the deductive difference, in this paper we rely on the *approximation* given in Definition 1. This approximation only requires comparing the classification hierarchies of $\mathcal{O}$ and $\mathcal{O}'$ provided by an OWL 2 reasoner, and it has successfully been used in the past in the context of ontology integration [18].

### 2.3 Mapping Coherence and Mapping Repair

The consistency principle requires that the vocabulary in $\mathcal{O}_{\mathcal{U}} = \mathcal{O}_1 \cup \mathcal{O}_2 \cup \mathcal{M}$ be satisfiable, assuming the union of input ontologies $\mathcal{O}_1 \cup \mathcal{O}_2$ (without the mappings $\mathcal{M}$) does not contain unsatisfiable concepts. Thus $\mathsf{diff}^{\approx}_{\Sigma}(\mathcal{O}_1 \cup \mathcal{O}_2, \mathcal{O}_{\mathcal{U}})$ should not contain any axiom of the form $A \sqsubseteq \bot$, for any $A \in \Sigma = \mathsf{Sig}(\mathcal{O}_1 \cup \mathcal{O}_2)$.

**Definition 2 (Mapping Incoherence).** *A set of mappings $\mathcal{M}$ is incoherent with respect to $\mathcal{O}_1$ and $\mathcal{O}_2$, if there exists a class $A$, in the signature of $\mathcal{O}_1 \cup \mathcal{O}_2$, such that $\mathcal{O}_1 \cup \mathcal{O}_2 \not\models A \sqsubseteq \bot$ and $\mathcal{O}_1 \cup \mathcal{O}_2 \cup \mathcal{M} \models A \sqsubseteq \bot$.*

An incoherent set of mappings $\mathcal{M}$ can be fixed by removing mappings from $\mathcal{M}$. This process is referred to as *mapping repair* (or repair for short).

**Definition 3 (Mapping Repair).** *Let $\mathcal{M}$ be an incoherent set of mappings w.r.t. $\mathcal{O}_1$ and $\mathcal{O}_2$. A set of mappings $\mathcal{R} \subseteq \mathcal{M}$ is a mapping repair for $\mathcal{M}$ w.r.t. $\mathcal{O}_1$ and $\mathcal{O}_2$ iff $\mathcal{M} \setminus \mathcal{R}$ is coherent w.r.t. $\mathcal{O}_1$ and $\mathcal{O}_2$.*

A trivial repair is $\mathcal{R} = \mathcal{M}$, since an empty set of mappings is trivially coherent (according to Definition 2). Nevertheless, the objective is to remove as few mappings as possible. Minimal (mapping) repairs are typically referred to in the literature as *mapping diagnoses* [29] — a term coined by Reiter [36] and introduced to the field of ontology debugging in [39]. A repair or diagnosis can be computed by extracting the justifications for the unsatisfiable concepts (*e.g.*, [38, 22, 43]), and selecting a hitting set of mappings to be removed, following a minimality criteria (*e.g.*, the number of removed mappings). However, justification-based technologies do not scale when the number of unsatisfiabilities is large (a typical scenario in mapping repair problems [16]). To address this scalability issue, mapping repair systems usually compute an *approximate repair* using incomplete reasoning techniques (*e.g.*, [17, 29, 37]). An approximate repair $\mathcal{R}^{\approx}$ does not guarantee that $\mathcal{M} \setminus \mathcal{R}^{\approx}$ is coherent, but it will (in general) significantly reduce the number of unsatisfiabilities caused by the original set of mappings $\mathcal{M}$.

### 2.4 Conservativity Principle

The conservativity principle (general notion) states that the integrated ontology $\mathcal{O}_{\mathcal{U}} = \mathcal{O}_1 \cup \mathcal{O}_2 \cup \mathcal{M}$ should not induce any change in the concept hierarchies of the input ontologies $\mathcal{O}_1$ and $\mathcal{O}_2$. That is, the sets $\mathsf{diff}^{\approx}_{\Sigma_1}(\mathcal{O}_1, \mathcal{O}_{\mathcal{U}})$ and $\mathsf{diff}^{\approx}_{\Sigma_2}(\mathcal{O}_2, \mathcal{O}_{\mathcal{U}})$ must be empty for signatures $\Sigma_1 = \mathsf{Sig}(\mathcal{O}_1)$ and $\Sigma_2 = \mathsf{Sig}(\mathcal{O}_2)$, respectively.

In [19] a lighter variant of the conservativity principle was proposed. This variant required that the mappings $\mathcal{M}$ alone should not introduce new subsumption relationships between concepts from one of the input ontologies. That is, the set $\mathsf{diff}^{\approx}_{\Sigma}(\mathcal{O}_1, \mathcal{O}_1 \cup \mathcal{M})$ (resp. $\mathsf{diff}^{\approx}_{\Sigma}(\mathcal{O}_2, \mathcal{O}_2 \cup \mathcal{M})$) must be empty for $\Sigma = \mathsf{Sig}(\mathcal{O}_1)$ (resp. $\Sigma = \mathsf{Sig}(\mathcal{O}_2)$).

In this paper we propose a different variant of the conservativity principle where we require that the integrated ontology $\mathcal{O}_{\mathcal{U}}$ does not introduce new subsumption relationships between concepts from one of the input ontologies, unless they were already involved in a subsumption relationship or they shared a common descendant. Note that we assume that the mappings $\mathcal{M}$ are coherent with respect to $\mathcal{O}_1$ and $\mathcal{O}_2$.

**Definition 4 (Conservativity Principle Violations).** *Let $A, B, C$ be atomic concepts (not including $\top, \bot$), let $\mathcal{O}$ be one of the input ontologies, let $\mathsf{Sig}(\mathcal{O})$ be its signature, and let $\mathcal{O}_{\mathcal{U}}$ be the integrated ontology. We define the set of conservativity principle violations of $\mathcal{O}_{\mathcal{U}}$ w.r.t. $\mathcal{O}$ (denoted $\mathsf{consViol}(\mathcal{O}, \mathcal{O}_{\mathcal{U}})$) as the set of axioms of the form $A \sqsubseteq B$ satisfying: (i) $A, B, C \in \mathsf{Sig}(\mathcal{O})$, (ii) $A \sqsubseteq B \in \mathsf{diff}^{\approx}_{\mathsf{Sig}(\mathcal{O})}(\mathcal{O}, \mathcal{O}_{\mathcal{U}})$, (iii) $\mathcal{O} \not\models B \sqsubseteq A$, and (iv) there is no $C$ s.t. $\mathcal{O} \models C \sqsubseteq A$, and $\mathcal{O} \models C \sqsubseteq B$.*

This variant of the conservativity principle follows the *assumption of disjointness* proposed in [38]. That is, if two atomic concepts $A, B$ from one of the input ontologies are not involved in a subsumption relationship nor share a common subconcept (excluding $\bot$) they can be considered as disjoint. Hence, the conservativity principle can be reduced to the consistency principle, if the input ontologies are extended with sufficient disjointness axioms. This reduction will allow us to reuse the available infrastructure and techniques for mapping repair.

**Table 1.** Fragments of the ontologies used in Optique.

| Ontology $\mathcal{O}_1$ | Ontology $\mathcal{O}_2$ |
|---|---|
| $\alpha_1$ WellBore $\sqsubseteq$ $\exists$belongsTo.Well | $\beta_1$ Exploration_well $\sqsubseteq$ Well |
| $\alpha_2$ WellBore $\sqsubseteq$ $\exists$hasOperator.Operator | $\beta_2$ Explor_borehole $\sqsubseteq$ Borehole |
| $\alpha_3$ WellBore $\sqsubseteq$ $\exists$locatedIn.Field | $\beta_3$ Appraisal_exp_borehole $\sqsubseteq$ Explor_borehole |
| $\alpha_4$ AppraisalWellBore $\sqsubseteq$ WellBore | $\beta_4$ Appraisal_well $\sqsubseteq$ Well |
| $\alpha_5$ ExplorationWellBore $\sqsubseteq$ WellBore | $\beta_5$ Field $\sqsubseteq$ $\exists$hasFieldOperator.Field_operator |
| $\alpha_6$ Operator $\sqsubseteq$ Owner | $\beta_6$ Field_operator $\sqcap$ Owner $\sqsubseteq$ Field_owner |
| $\alpha_7$ Operator $\sqsubseteq$ Company | $\beta_7$ Company $\sqsubseteq$ Field_operator |
| $\alpha_8$ Field $\sqsubseteq$ $\exists$hasOperator.Company | $\beta_8$ Field_owner $\sqsubseteq$ Owner |
| $\alpha_9$ Field $\sqsubseteq$ $\exists$hasOwner.Owner | $\beta_9$ Borehole $\sqsubseteq$ Continuant $\sqcup$ Occurrent |

**Table 2.** Ontology mappings for the vocabulary in $\mathcal{O}_1$ and $\mathcal{O}_2$.

| Mappings $\mathcal{M}$ | | | | |
|---|---|---|---|---|
| **id** | $\mathbf{e_1}$ | $\mathbf{e_2}$ | **n** | $\boldsymbol{\rho}$ |
| $m_1$ | $\mathcal{O}_1$:Well | $\mathcal{O}_2$:Well | 0.9 | $\equiv$ |
| $m_2$ | $\mathcal{O}_1$:WellBore | $\mathcal{O}_2$:Borehole | 0.7 | $\equiv$ |
| $m_3$ | $\mathcal{O}_1$:ExplorationWellBore | $\mathcal{O}_2$:Exploration_well | 0.6 | $\sqsubseteq$ |
| $m_4$ | $\mathcal{O}_1$:ExplorationWellBore | $\mathcal{O}_2$:Explor_borehole | 0.8 | $\equiv$ |
| $m_5$ | $\mathcal{O}_1$:AppraisalWellBore | $\mathcal{O}_2$:Appraisal_exp_borehole | 0.7 | $\equiv$ |
| $m_6$ | $\mathcal{O}_1$:Field | $\mathcal{O}_2$:Field | 0.9 | $\equiv$ |
| $m_7$ | $\mathcal{O}_1$:Operator | $\mathcal{O}_2$:Field_operator | 0.7 | $\sqsupseteq$ |
| $m_8$ | $\mathcal{O}_1$:Company | $\mathcal{O}_2$:Company | 0.9 | $\equiv$ |
| $m_9$ | $\mathcal{O}_1$:hasOperator | $\mathcal{O}_2$:hasFieldOperator | 0.6 | $\equiv$ |
| $m_{10}$ | $\mathcal{O}_1$:Owner | $\mathcal{O}_2$:Owner | 0.9 | $\equiv$ |

## 3 Conservativity Principle Violations in Practice

In this section, we show the problems led by the violation of the conservativity principle when integrating ontologies via mappings in a real-world scenario. To this end, we consider as motivating example a use case based on the Optique's application domain.

Table 1 shows the fragments of two ontologies in the context of the oil and gas industry. The ontology $\mathcal{O}_1$ has been directly bootstrapped from a relational database in Optique, and it is linked to the data via direct ontology-to-database mappings. The ontology $\mathcal{O}_2$, instead, is a domain ontology, based on the NPD FactPages, preferred by Optique end-users to feed the visual query formulation interface.[8]

The integration via ontology matching of $\mathcal{O}_1$ and $\mathcal{O}_2$ is required since the vocabulary in $\mathcal{O}_2$ is used to formulate queries, but only the vocabulary of $\mathcal{O}_1$ is connected to the database.[9] Consider the set of mappings $\mathcal{M}$ in Table 2 between $\mathcal{O}_1$ and $\mathcal{O}_2$ generated by an off-the-shelf ontology alignment system. As described in Section 2.1, mappings are represented as 5-tuples; for example the mapping $m_2$ suggests an equivalence relationship between the entities $\mathcal{O}_1$:WellBore and $\mathcal{O}_2$:Borehole, with confidence 0.7.

The integrated ontology $\mathcal{O}_{\mathcal{U}} = \mathcal{O}_1 \cup \mathcal{O}_2 \cup \mathcal{M}$, however, violates the conservativity principle, according to Definition 4, and introduces non desired subsumption relationships (see Table 3). Note that the entailments $\sigma_4$ and $\sigma_5$ are not included in our variant of conservativity violation, since $\mathcal{O}_1$:Company and $\mathcal{O}_1$:Operator (resp. $\mathcal{O}_2$:Field_operator and $\mathcal{O}_2$:Company) are involved in a subsumption relationship in $\mathcal{O}_1$ (resp. $\mathcal{O}_2$). How-

[8] In Optique we use OWL 2 QL ontologies for query rewriting, while the query formulation may be based on much richer OWL 2 ontologies. The axioms that fall outside the OWL 2 QL profile are either approximated or not considered for the rewriting.

[9] As mentioned in Section 1, in this paper we only focus on ontology-to-ontology mappings.

**Table 3.** Example of conservativity principle violations.

| $\sigma$ | **Entailment:** | **follows from:** | **Violation?** |
|---|---|---|---|
| $\sigma_1$ | $\mathcal{O}_2$:Explor_borehole $\sqsubseteq$ $\mathcal{O}_2$:Exploration_well | $m_3, m_4$ | YES |
| $\sigma_2$ | $\mathcal{O}_1$:AppraisalWellBore $\sqsubseteq$ $\mathcal{O}_1$:ExplorationWellBore | $\beta_3, m_4, m_5$ | YES |
| $\sigma_3$ | $\mathcal{O}_2$:Field_operator $\sqsubseteq$ $\mathcal{O}_2$:Field_owner | $\alpha_6, \beta_6, m_7, m_{10}$ | YES |
| $\sigma_4$<br>$\sigma_5$ | $\mathcal{O}_1$:Company $\equiv$ $\mathcal{O}_1$:Operator<br>$\mathcal{O}_2$:Field_operator $\equiv$ $\mathcal{O}_2$:Company | $\alpha_7, \beta_7, m_7, m_8$ | NO (*) |
| $\sigma_6$ | $\mathcal{O}_1$:Company $\sqsubseteq$ $\mathcal{O}_1$:Owner | $\sigma_4, \alpha_6$ | YES |
| $\sigma_7$ | $\mathcal{O}_2$:Company $\sqsubseteq$ $\mathcal{O}_2$:Field_owner | $\sigma_3, \sigma_5$ | YES |

ever, these entailments lead to other violations included in our variant ($\sigma_6$ and $\sigma_7$), and may also be considered as violations. These conservativity principle violations may hinder the usefulness of the generated ontology mappings since may affect the quality of the results when performing OBDA queries over the vocabulary of $\mathcal{O}_2$.

*Example 1.* Consider the following conjunctive query $\mathsf{CQ(x)} \leftarrow \mathcal{O}_2{:}\mathsf{Well(x)}$. The query asks for wells and has been formulated from the Optique's query formulation interface, using the vocabulary of $\mathcal{O}_2$. The query is rewritten, according to the ontology axioms and mappings $\beta_1, \beta_4, m_1, m_3, m_4$ in $\mathcal{O}_{\mathcal{U}} = \mathcal{O}_1 \cup \mathcal{O}_2 \cup \mathcal{M}$, into the following union of conjunctive queries $\mathsf{UCQ(x)} \leftarrow \mathcal{O}_2{:}\mathsf{Well(x)} \cup \mathcal{O}_1{:}\mathsf{Well(x)} \cup \mathcal{O}_2{:}\mathsf{Exploration\_well(x)} \cup \mathcal{O}_2{:}\mathsf{Appraisal\_well(x)} \cup \mathcal{O}_1{:}\mathsf{ExplorationWellBore(x)} \cup \mathcal{O}_2{:}\mathsf{Explor\_borehole(x)}$. Since only the vocabulary of $\mathcal{O}_1$ is linked to the data, the union of conjunctive queries could be simplified as $\mathsf{UCQ(x)} \leftarrow \mathsf{Well(x)} \cup \mathsf{ExplorationWellBore(x)}$, which will clearly lead to non desired results. The original query was only asking for wells, while the rewritten query will also return data about exploration wellbores.

We have shown that the quality of the mappings in terms of conservativity principle violations will directly affect the quality of the query results. Therefore, the detection and repair of these violations arise as an important quality assessment step in Optique.

## 4 Methods

We have reduced the problem of detecting and solving conservativity principle violations, following our notion of conservativity (see Section 2), to a mapping (incoherence) repair problem. Currently, our method relies on the indexing and reasoning techniques implemented in *LogMap*, an ontology matching and mapping repair system [17, 20, 21].

Algorithm 1 shows the pseudocode of the implemented method. The algorithm accepts as input two OWL 2 ontologies, $\mathcal{O}_1$ and $\mathcal{O}_2$, and a set of mappings $\mathcal{M}$ which are coherent[10] with respect to $\mathcal{O}_1$ and $\mathcal{O}_2$. Additionally, an optimised variant to add disjointness axioms can be selected. The algorithm outputs the number of added disjointness during the process $disj$, a set of mappings $\mathcal{M}'$, and an (approximate) repair $\mathcal{R}^{\approx}$ such that $\mathcal{M}' = \mathcal{M} \setminus \mathcal{R}^{\approx}$. The (approximate) repair $\mathcal{R}^{\approx}$ aims at solving most of the conservativity principle violations of $\mathcal{M}$ with respect to $\mathcal{O}_1$ and $\mathcal{O}_2$. We next describe the techniques used in each step.

[10] Note that $\mathcal{M}$ may be the result of a prior mapping (incoherence) repair process.

**Algorithm 1** Algorithm to detect and solve conservativity principle violations

**Input:** $\mathcal{O}_1$, $\mathcal{O}_2$: input ontologies; $\mathcal{M}$: (coherent) input mappings; $Optimization$: Boolean value
**Output:** $\mathcal{M}'$: output mappings; $\mathcal{R}^{\approx}$: approximate repair; $disj$: number of disjointness rules

1: $\langle \mathcal{O}_1', \mathcal{O}_2' \rangle := \mathsf{ModuleExtractor}(\mathcal{O}_1, \mathcal{O}_2, \mathcal{M})$
2: $\langle \mathcal{P}_1, \mathcal{P}_2 \rangle := \mathsf{PropositionalEncoding}(\mathcal{O}_1', \mathcal{O}_2')$
3: $SI_1 := \mathsf{StructuralIndex}(\mathcal{O}_1')$
4: $SI_2 := \mathsf{StructuralIndex}(\mathcal{O}_2')$
5: **if** ($Optimization$ = true) **then**
6: $\quad SI_{\mathcal{U}} := \mathsf{StructuralIndex}(\mathcal{O}_1' \cup \mathcal{O}_2' \cup \mathcal{M})$
7: $\quad \langle \mathcal{P}_1^d, disj_1 \rangle := \mathsf{DisjointAxiomsExtensionOptimized}(\mathcal{P}_1, SI_1, SI_{\mathcal{U}})$ ▷ See Algorithm 3
8: $\quad \langle \mathcal{P}_2^d, disj_2 \rangle := \mathsf{DisjointAxiomsExtensionOptimized}(\mathcal{P}_2, SI_2, SI_{\mathcal{U}})$
9: **else**
10: $\quad \langle \mathcal{P}_1^d, disj_1 \rangle := \mathsf{DisjointAxiomsExtensionBasic}(\mathcal{P}_1, SI_1)$ ▷ See Algorithm 2
11: $\quad \langle \mathcal{P}_2^d, disj_2 \rangle := \mathsf{DisjointAxiomsExtensionBasic}(\mathcal{P}_2, SI_2)$
12: **end if**
13: $\langle \mathcal{M}', \mathcal{R}^{\approx} \rangle := \mathsf{MappingRepair}(\mathcal{P}_1^d, \mathcal{P}_2^d, \mathcal{M})$ ▷ See Algorithm 2 in [21]
14: $disj := disj_1 + disj_2$
15: **return** $\langle \mathcal{M}', \mathcal{R}^{\approx}, disj \rangle$

**Module Extraction.** In order to reduce the size of the problem our method extracts two locality-based modules [7], one for each input ontology, using the entities involved in the mappings $\mathcal{M}$ as seed signatures for the module extractor (step 1 in Algorithm 1). These modules preserve the semantics for the given entities, can be efficiently computed, and are typically much smaller than the original ontologies.

**Propositional Horn Encoding.** The modules $\mathcal{O}_1'$ and $\mathcal{O}_2'$ are encoded as the Horn propositional theories, $\mathcal{P}_1$ and $\mathcal{P}_2$ (step 2 in Algorithm 1). This encoding includes rules of the form $A_1 \wedge \ldots \wedge A_n \rightarrow B$. For example, the concept hierarchy provided by an OWL 2 reasoner (*e.g.*, [32, 23]) will be encoded as $A \rightarrow B$ rules, while the explicit disjointness relationships between concepts will be represented as $A_i \wedge A_j \rightarrow \mathsf{false}$. Note that the input mappings $\mathcal{M}$ can already be seen as propositional implications. This encoding is key to the mapping repair process.

*Example 2.* Consider the ontologies and mappings in Tables 1 and 2. The axiom $\beta_6$ is encoded as $\mathsf{Field\_operator} \wedge \mathsf{Owner} \rightarrow \mathsf{Field\_owner}$, while the mapping $m_2$ is translated into rules $\mathcal{O}_1$:$\mathsf{WellBore} \rightarrow \mathcal{O}_2$:$\mathsf{Borehole}$, and $\mathcal{O}_2$:$\mathsf{Borehole} \rightarrow \mathcal{O}_1$:$\mathsf{WellBore}$.

**Structural Index.** The concept hierarchies provided by an OWL 2 reasoner (excluding $\bot$) and the explicit disjointness axioms of the modules $\mathcal{O}_1'$ and $\mathcal{O}_2'$ are efficiently indexed using an interval labelling schema [1] (steps 3 and 4 in Algorithm 1). This structural index exploits an optimised data structure for storing directed acyclic graphs (DAGs), and allows us to answer many entailment queries over the concept hierarchy as an index lookup operation, and hence without the need of an OWL 2 reasoner. This kind of index has shown to significantly reduce the cost of answering taxonomic queries [5, 33] and disjointness relationships queries [17, 20].

**Disjointness Axioms Extension.** In order to reduce the conservativity problem to a mapping incoherence repair problem following the notion of *assumption of disjointness*, we need to automatically add sufficient disjointness axioms into each module $\mathcal{O}_i'$. However, the insertion of additional disjointness axioms $\delta$ may lead to unsatisfiable classes in $\mathcal{O}_i' \cup \delta$.

**Algorithm 2** Basic disjointness axioms extension

**Input:** $\mathcal{P}$: propositional theory; $SI$: structural index
**Output:** $\mathcal{P}^d$: extended propositional theory;$disj$: number of disjointness rules

1: $disj := 0$
2: $\mathcal{P}^d := \mathcal{P}$
3: **for each** pair $\langle A, B\rangle \in$ OrderedVariablePairs($\mathcal{P}$) **do**
4:     **if not** (areDisj($SI, A, B$) **or** inSubSupRel($SI, A, B$) **or** shareDesc($SI, A, B$)) **then**
5:         $\mathcal{P}^d := \mathcal{P}^d \cup \{A \wedge B \rightarrow false\}$
6:         $SI :=$ updateIndex($SI, A \sqcap B \rightarrow \bot$)
7:         $disj := disj + 1$
8:     **end if**
9: **end for**
10: **return** $\langle \mathcal{P}^d, disj\rangle$

*Example 3.* Consider the axiom $\beta_9$ from Table 1. Following the *assumption of disjointness* a very naïve algorithm would add disjointness axioms between Borehole, Continuant and Occurrent, which would make Borehole unsatisfiable.

In order to detect if each candidate disjointness axiom leads to an unsatisfiability, a non naïve algorithm requires to make an extensive use of an OWL 2 reasoner. In large ontologies, however, such extensive use of the reasoner may be prohibitive. Our method, in order to address this issue, exploits the propositional encoding and structural index of the input ontologies. Thus, checking if $\mathcal{O}'_i \cup \delta$ contains unsatisfiable classes is restricted to the Horn propositional case.

We have implemented two algorithms to extend the propositional theories $\mathcal{P}_1$ and $\mathcal{P}_2$ with disjointness rules of the form $A \wedge B \rightarrow \bot$ (see steps 5-12 in Algorithm 1). These algorithms guarantee that, for every propositional variable $A$ in the extended propositional theory $\mathcal{P}^d_i$ (with $i \in \{1,2\}$), the theory $\mathcal{P}^d_i \cup \{true \rightarrow A\}$ is satisfiable. Note that this does not necessarily hold if the disjointness axioms are added to the OWL 2 ontology modules, $\mathcal{O}'_1$ and $\mathcal{O}'_2$, as discussed above.

Algorithm 2 presents a (basic) algorithm to add as many disjointness rules as possible, for every pair of propositional variables $A, B$ in the propositional theory $\mathcal{P}$ given as input. In order to minimize the number of necessary disjointness rules, the variables in $\mathcal{P}$ are ordered in pairs following a top-down approach. The algorithm exploits the structural index $SI$ to check if two propositional variables (*i.e.*, classes in the input ontologies) are disjoint (areDisj($SI, A, B$)), they keep a sub/super-class relationship (inSubSupRel($SI, A, B$)), or they share a common descendant (shareDesc($SI, A, B$)) (step 4 in Algorithm 2). Note that the structural index is also updated to take into account the new disjointness rules (step 6 in Algorithm 2).

The addition of disjointness rules in Algorithm 2, however, may be prohibitive for large ontologies (see Section 5). Intuitively, in order to reduce the number of disjointness axioms, one should only focus on the cases where a conservativity principle violation occurs in the integrated ontology $\mathcal{O}_\mathcal{U} = \mathcal{O}'_1 \cup \mathcal{O}'_2 \cup \mathcal{M}$, with respect to one of the ontology modules $\mathcal{O}'_i$ (with $i \in \{1,2\}$); *i.e.*, adding a disjointness axiom between each pair of classes $A, B \in \mathcal{O}'_i$ such that $A \sqsubseteq B \in$ consViol($\mathcal{O}'_i, \mathcal{O}_\mathcal{U}$), as in Definition 4. Algorithm 3 implements this idea for the Horn propositional case and extensively exploits the structural indexing to identify the conservativity principle violations (step 3 in Algorithm 3). Note that this algorithm also requires to compute the structural index

**Algorithm 3** Optimised disjointness axioms extension

**Input:** $\mathcal{P}$: propositional theory; $SI$: structural index $SI_{\mathcal{U}}$: structural index of the union ontology
**Output:** $\mathcal{P}^d$: extended propositional theory;$disj$: number of disjointness rules

$disj := 0$
$\mathcal{P}^d := \mathcal{P}$
**for** $A \rightarrow B \in \mathsf{ConservativityViolations}(SI, SI_{\mathcal{U}})$ **do**
  **if not** $(\mathsf{areDisj}(SI, A, B))$ **then**
    $\mathcal{P}^d := \mathcal{P}^d \cup \{A \wedge B \rightarrow false\}$
    $SI := \mathsf{updateIndex}(SI, A \sqcap B \rightarrow \bot)$
    $disj := disj + 1$
  **end if**
**end for**
**return** $\langle \mathcal{P}^d, disj \rangle$

of the integrated ontology, and thus its classification with an OWL 2 reasoner (step 6 in Algorithm 1). The classification of the integrated ontology is known to be typically much higher than the classification of the input ontologies individually [16]. However, this was not a bottleneck in our experiments, as shown in Section 5.

**Mapping Repair.** The step 13 of Algorithm 1 uses the mapping (incoherence) repair algorithm presented in [17, 21] for the extended Horn propositional theories $\mathcal{P}_1^d$ and $\mathcal{P}_2^d$, and the input mappings $\mathcal{M}$. The mapping repair process exploits the Dowling-Gallier algorithm for propositional Horn satisfiability [9] and checks, for every propositional variable $A \in \mathcal{P}_1^d \cup \mathcal{P}_2^d$, the satisfiability of the propositional theory $\mathcal{P}_A = \mathcal{P}_1^d \cup \mathcal{P}_2^d \cup \mathcal{M} \cup \{true \rightarrow A\}$. Satisfiability of $\mathcal{P}_A$ is checked in worst-case linear time in the size of $\mathcal{P}_A$, and the number of Dowling-Gallier calls is also linear in the number of propositional variables in $\mathcal{P}_1^d \cup \mathcal{P}_2^d$. In case of unsatisfiability, the algorithm also allows us to record *conflicting* mappings involved in the unsatisfiability, which will be considered for the subsequent repair process. The unsatisfiability will be fixed by removing some of the identified mappings. In case of multiple options, the mapping confidence will be used as a differentiating factor.[11]

*Example 4.* Consider the propositional encoding $\mathcal{P}_1$ and $\mathcal{P}_2$ of the axioms of Table 1 and the mappings $\mathcal{M}$ in Table 2, seen as propositional rules. $\mathcal{P}_1^d$ and $\mathcal{P}_2^d$ have been created by adding disjointness rules to $\mathcal{P}_1$ and $\mathcal{P}_2$, according to Algorithm 2 or 3. For example, $\mathcal{P}_2^d$ includes the rule $\psi = \mathcal{O}_2{:}\mathsf{Well} \wedge \mathcal{O}_2{:}\mathsf{Borehole} \rightarrow false$. The mapping repair algorithm identifies the propositional theory $\mathcal{P}_1^d \cup \mathcal{P}_2^d \cup \mathcal{M} \cup \{true \rightarrow \mathcal{O}_1{:}\mathsf{ExplorationWellbore}\}$ as unsatisfiable. This is due to the combination of the mappings $m_3$ and $m_4$, the propositional projection of axioms $\beta_1$ and $\beta_2$, and the rule $\psi$. The mapping repair algorithm also identifies $m_3$ and $m_4$ as the cause of the unsatisfiability, and discards $m_3$, since its confidence is smaller than that of $m_4$ (see Table 2).

Algorithm 1 gives as output the number of added disjointness rules during the process $disj$, a set of mappings $\mathcal{M}'$, and an (approximate) repair $\mathcal{R}^{\approx}$ such that $\mathcal{M}' = \mathcal{M} \setminus \mathcal{R}^{\approx}$. $\mathcal{M}'$ is coherent with respect to $\mathcal{P}_1^d$ and $\mathcal{P}_2^d$ (according to the propositional case of Definition 2). Furthermore, the propositional theory $\mathcal{P}_1 \cup \mathcal{P}_2 \cup \mathcal{M}'$ does not

[11] In scenarios where the confidence of the mapping is missing (*e.g.*, in reference or manually created mapping sets) or unreliable, our mapping repair technique computes fresh confidence values based on the locality principle [19].

**Algorithm 4** Conducted evaluation over the Optique and OAEI data sets

**Input:** $\mathcal{O}_1, \mathcal{O}_2$: input ontologies $\mathcal{M}$: reference mappings for $\mathcal{O}_1$ and $\mathcal{O}_2$

$\mathcal{O}_{\mathcal{U}} := \mathcal{O}_1 \cup \mathcal{O}_2 \cup \mathcal{M}$

Store size of $\mathsf{Sig}(\mathcal{O}_1)$ (**I**), $\mathsf{Sig}(\mathcal{O}_2)$ (**II**) and $\mathcal{M}$ (**III**)

Compute number of conservativity principle violations (our variant as in Definition 4): $\mathsf{consViol} := |\mathsf{consViol}(\mathcal{O}_1, \mathcal{O}_{\mathcal{U}})| + |\mathsf{consViol}(\mathcal{O}_2, \mathcal{O}_{\mathcal{U}})|$ (**IV**)

Compute number of conservativity principle violations (general notion as in Section 2.4): $\mathsf{diff}^{\approx} := |\mathsf{diff}^{\approx}_{\mathsf{Sig}(\mathcal{O}_1)}(\mathcal{O}_1, \mathcal{O}_{\mathcal{U}})| + |\mathsf{diff}^{\approx}_{\mathsf{Sig}(\mathcal{O}_2)}(\mathcal{O}_2, \mathcal{O}_{\mathcal{U}})|$ (**V**)

Compute two repairs $\mathcal{R}^{\approx}$ using Algorithm 1 for $\mathcal{O}_1$, $\mathcal{O}_2$, $\mathcal{M}$, with the $Optimization$ set to $\mathsf{false}$ (see Table 5) and $\mathtt{true}$ (see Table 6)

Store number of added disjointness $disj$ (**VI** and **XII**), size of repair $|\mathcal{R}^{\approx}|$ (**VII** and **XIII**), time to compute disjointness rules $\mathrm{t}_d$ (**VIII** and **XIV**), and time to compute the mapping repair $\mathrm{t}_r$ (**IX** and **XV**)

$\mathcal{O}_{\mathcal{U}} := \mathcal{O}_1 \cup \mathcal{O}_2 \cup \mathcal{M} \setminus \mathcal{R}^{\approx}$

Compute number of remaining conservativity principle violations (our variant): $\mathsf{consViol} := |\mathsf{consViol}(\mathcal{O}_1, \mathcal{O}_{\mathcal{U}})| + |\mathsf{consViol}(\mathcal{O}_2, \mathcal{O}_{\mathcal{U}})|$ (**X** and **XVI**)

Compute number of remaining conservativity principle violations (general notion): $\mathsf{diff}^{\approx} := |\mathsf{diff}^{\approx}_{\mathsf{Sig}(\mathcal{O}_1)}(\mathcal{O}_1, \mathcal{O}_{\mathcal{U}})| + |\mathsf{diff}^{\approx}_{\mathsf{Sig}(\mathcal{O}_2)}(\mathcal{O}_2, \mathcal{O}_{\mathcal{U}})|$ (**XI** and **XVII**)

contain any conservativity principle violation with respect to $\mathcal{P}_1$ and $\mathcal{P}_2$ (according to the propositional case of Definition 4). However, our encoding is incomplete, and we cannot guarantee that $\mathcal{O}'_1 \cup \mathcal{O}'_2 \cup \mathcal{M}'$ does not contain conservativity principle violations with respect to $\mathcal{O}'_1$ and $\mathcal{O}'_2$. Nonetheless, our evaluation suggests that the number of remaining violations after repair is typically small (see Section 5).

## 5 Evaluation

In this section we evaluate the feasibility of using our method to detect and correct conservativity principle violations in practice. To this end we have conducted the evaluation in Algorithm 4 (the Roman numbers refer to stored measurements) over the Optique's use case and the ontologies and reference mapping sets of the OAEI 2013 campaign:[12]

i *Optique*'s use case is based on the NPD ontology and a bootstrapped ontology (BootsOnto) from one of the Optique databases. The mappings between these ontologies were semi-automatically created using the ontology matcher *LogMap* [20]. Although the NPD ontology is small with respect to the size of the bootstrapped ontology, its vocabulary covers a large portion of the current query catalog in Optique.

ii *LargeBio*: this dataset includes the biomedical ontologies FMA, NCI and (a fragment of) SNOMED, and reference mappings based on the UMLS [3].

iii *Anatomy*: the Anatomy dataset involves the Adult Mouse Anatomy (MO) ontology and a fragment of the NCI ontology ($\mathrm{NCI_{Anat}}$), describing human anatomy. The reference alignment has been manually curated [48].

iv *Library*: this OAEI dataset includes the real-word thesauri STW and TheSoz from the social sciences. The reference mappings have been manually validated.

v *Conference*: this dataset uses a collection of 16 ontologies from the domain of academic conferences [46]. Currently, there are 21 manually created mapping sets among 7 of the ontologies.

[12] Note that the reference mappings of the OAEI 2013 campaign are coherent with respect to the test case ontologies [13]. More information about the used ontology versions can be found in `http://oaei.ontologymatching.org/2013/`

**Table 4.** Test cases and violations with original reference mappings. BootsOnto contains around 3,000 concepts, and a large number of properties.

| Dataset | $\mathcal{O}_1 \sim \mathcal{O}_2$ | Problem size | | | Original Violations | |
|---|---|---|---|---|---|---|
| | | I | II | III | IV | V |
| | | $\lvert\mathsf{Sig}(\mathcal{O}_1)\rvert$ | $\lvert\mathsf{Sig}(\mathcal{O}_2)\rvert$ | $\lvert\mathcal{M}\rvert$ | consViol | $\mathsf{diff}^{\approx}$ |
| **Optique** | **NPD∼BootsOnto** | 757 | 40,671 | 102 | 214 | 220 |
| **LargeBio** | **SNOMED∼NCI** | 122,519 | 66,914 | 36,405 | >525,515 | >546,181 |
| | **FMA∼SNOMED** | 79,042 | 122,519 | 17,212 | 125,232 | 127,668 |
| | **FMA∼NCI** | 79,042 | 66,914 | 5,821 | 19,740 | 19,799 |
| **Anatomy** | **MO∼$\mathbf{NCI_{Anat}}$** | 2,747 | 3,306 | 3,032 | 1,321 | 1,335 |
| **Library** | **STW∼TheSoz** | 6,575 | 8,376 | 6,322 | 42,045 | 42,872 |
| **Conference** | **cmt∼confof** | 89 | 75 | 32 | 11 | 11 |
| | **conference∼edas** | 124 | 154 | 34 | 8 | 8 |
| | **conference∼iasted** | 124 | 182 | 28 | 9 | 9 |
| | **confof∼ekaw** | 75 | 107 | 40 | 6 | 6 |
| | **edas∼iasted** | 154 | 182 | 38 | 7 | 7 |

**Table 5.** Results of our basic method to detect and solve conservativity principle violations.

| Dataset | $\mathcal{O}_1 \sim \mathcal{O}_2$ | Solution size | | Times | | Remaining Violations | |
|---|---|---|---|---|---|---|---|
| | | VI | VII | VIII | IX | X | XI |
| | | #disj | $\lvert\mathcal{R}^{\approx}\rvert$ | $t_d$(s) | $t_r$(s) | consViol | $\mathsf{diff}^{\approx}$ |
| **Optique** | **NPD∼BootsOnto** | 4,716,685 | 49 | 9,840 | 121 | 0 | 0 |
| **LargeBio** | **SNOMED∼NCI** | – | – | – | – | – | – |
| | **FMA∼SNOMED** | 1,106,259 | 8,234 | 35,817 | 1,127 | 0 | 121 |
| | **FMA∼NCI** | 347,801 | 2,176 | 2,471 | 38 | 103 | 112 |
| **Anatomy** | **MO∼$\mathbf{NCI_{Anat}}$** | 1,331,374 | 461 | 397 | 56 | 0 | 3 |
| **Library** | **STW∼TheSoz** | 591,115 | 2,969 | 4,126 | 416 | 0 | 24 |
| **Conference** | **cmt∼confof** | 50 | 6 | 0.01 | 0.01 | 0 | 0 |
| | **conference∼edas** | 774 | 6 | 0.03 | 0.01 | 0 | 0 |
| | **conference∼iasted** | 2,189 | 4 | 0.06 | 0.02 | 0 | 0 |
| | **confof∼ekaw** | 296 | 6 | 0.02 | 0.01 | 0 | 0 |
| | **edas∼iasted** | 1,210 | 4 | 0.06 | 0.02 | 1 | 1 |

Table 4 shows the size of the evaluated ontologies and mappings (**I**, **II** and **III**). For the Conference dataset we have selected only 5 pair of ontologies for which the reference mappings lead to more than five conservativity principle violations. Note that we count equivalence mappings as two subsumption mappings, and hence $\mathcal{M}$ represents subsumption mappings. Table 4 also shows the conservativity principle violations for the reference mappings (**IV** and **V**). For LargeBio and Library the number is expecially large using both our variant and the general notion of the conservativity principle.[13]

Tables 5 and 6 show the obtained results for our method using both the basic and optimised algorithms to add disjointness axioms.[14]

[13] In the SNOMED-NCI case no OWL 2 reasoner could succeed in classifying the integrated ontology via mappings [16], so we used the OWL 2 EL reasoner ELK [23] for providing a lower bound on the number of conservativity principle violations.

[14] The computation times of Steps 1-4 in Algorithm 1 were negligible with respect to the repair and disjointness addition times ($t_r$ and $t_d$) and thus they were not included in the result tables.

**Table 6.** Results of our optimised method to detect and solve conservativity principle violations.

| Dataset | $\mathcal{O}_1 \sim \mathcal{O}_2$ | Solution size | | Times | | Remaining Violations | |
|---|---|---|---|---|---|---|---|
| | | XII | XIII | XIV | XV | XVI | XVII |
| | | #disj | $\lvert\mathcal{R}^{\approx}\rvert$ | $t_d$(s) | $t_r$(s) | consViol | diff$^{\approx}$ |
| **Optique** | **NPD∼BootsOnto** | 214 | 41 | 2.54 | 0.17 | 0 | 0 |
| **LargeBio** | **SNOMED∼NCI** | 525,515 | 15,957 | 275 | 3,755 | >411 | >1,624 |
| | **FMA∼SNOMED** | 125,232 | 8,342 | 30 | 251 | 0 | 131 |
| | **FMA∼NCI** | 19,740 | 2,175 | 34 | 6.18 | 103 | 112 |
| **Anatomy** | **MO∼NCI$_{\text{Anat}}$** | 1,321 | 491 | 1.39 | 0.53 | 0 | 3 |
| **Library** | **STW∼TheSoz** | 42,045 | 3,058 | 4.93 | 41 | 0 | 40 |
| **Conference** | **cmt∼confof** | 11 | 6 | 0.05 | 0.01 | 0 | 0 |
| | **conference∼edas** | 8 | 6 | 0.07 | 0.01 | 0 | 0 |
| | **conference∼iasted** | 9 | 1 | 0.22 | 0.01 | 0 | 0 |
| | **confof∼ekaw** | 6 | 5 | 0.04 | 0.01 | 0 | 0 |
| | **edas∼iasted** | 7 | 4 | 0.21 | 0.02 | 1 | 1 |

We have run the experiments on a desktop computer with an *AMD Fusion A6-3670K* CPU and allocating 12 GB of RAM. The obtained results are summarized as follows:

i The number of added disjointness rules $disj$ (**VI**), as expected, is very large in the basic algorithm and the required time prohibitive (**VIII**) when involving SNOMED (it did not finish for SNOMED-NCI). This is clearly solved in our optimised algorithm that considerably reduces the number of necessary disjoitness rules (**XII**) and it requires only 275 seconds to compute them in the SNOMED-NCI case (**XIV**).

ii The computed repairs $\mathcal{R}^{\approx}$ (**VII** and **XIII**) using both the basic and optimised algorithms are of comparable size. This suggests that the large number of added disjointness in the basic algorithm does not have a negative impact (in terms of aggressiveness) on the repair process.

iii Repair times $t_r$ (**IX** and **XV**) are small and they do not represent a bottleneck in spite of the large number of added disjointness rules.

iv The conservativity principle violations using both algorithms and considering our variant (**X** and **XVI**) are completely removed in the Optique, Anatomy and Library cases, and almost completely removed in the Conference and LargeBio datasets.

v The number of missed violations is only slightly higher when considering the general notion of the conservativity principle (**XI** and **XVII**), which suggests that our (approximate) variant is also suitable in practice. Furthermore, in several test cases these violations are also almost removed.

vi The computed repairs $\mathcal{R}^{\approx}$, using both algorithms (**VII** and **XIII**), are rather aggressive and they can remove from 16% (Anatomy) up to 48% (Optique) of the mappings. In the Optique's use case, however, we follow a *better safe than sorry* approach and we prefer to remove as many violations as possible, rather than preserving potentially conflicting mapping sets.

In summary, the results suggest that our method to repair conservativity principle violations is suitable for Optique, and it is feasible in practice, even when considering the largest datasets of the OAEI.

## 6 Related Work

The conservativity principle problem, although indirectly, has been actively studied in the literature. For example, the assumption of disjointness was originally introduced by Schlobach [38] to enhance the repair of ontologies that were underspecified in terms of disjointness axioms. In [30], a similar assumption is followed in the context of repairing ontology mappings, where the authors restricted the number of disjointness axioms by using learning techniques [45]. These techniques, however, typically require a manually created training set. In [12] the authors present an interactive system to guide the expert user in the manual enrichment of the ontologies with disjointness axioms. In this paper, as in [45, 30, 12], we have also focused on the addition of a small set of disjointness axioms, since adding all possible disjointness may be unfeasible for large ontologies. However, our method does not require manual intervention. Furthermore, to address the scalability problem when dealing with large ontologies and mapping sets, our method relies on the propositional projection of the input ontologies.

Ontology matching systems have also dealt with the conservativity principle in order to improve the precision (with respect to a reference mapping set) of the computed mappings. For example, systems such as *ASMOV* [15], *Lily* [47] and *YAM++* [34] have implemented different heuristics and patterns to avoid violations of the conservativity principle. Another relevant approach has been presented in [2], where a set of sanity checks and best practices are proposed for computing ontology mappings. In this paper we present an elegant way to detect and solve conservativity principle violations by reducing the problem to a consistency principle violation problem. Concretely, we have reused and adapted the infrastructure provided by *LogMap* [17, 20]. However, other mapping repair systems, such as *Alcomo* [29] or *AML* [37], could be considered. Note that, to the best of our knowledge, these mapping repair systems have only focused on solving violations of the consistency principle.

The work presented in [26, 14, 27] deserves a special attention since they propose an opposite approach with respect to ours. Authors consider the violations of the conservativity principle as false positives, based on the potential incompleteness of the input ontologies. Hence, the correction strategy does not aim at removing mappings but at inserting subsumption axioms to the input ontologies to enrich their concept hierarchies. Authors in [35] also suggest that removing mapping may not be the best solution in a mapping repair process, and fixing the input ontologies may be an alternative.

Currently, in the Optique use case, we consider that the input ontologies are not modifiable. The query formulation ontology is based on the NPD ontology, which includes knowledge already agreed by the community, while the bootstrapped ontology is *directly* linked to the information represented in the database. Nevertheless, future extensions in Optique may consider appropriate the extension of the input ontologies.

## 7 Conclusions and Future Work

In this paper we have presented an approximate and fully-automatic method to detect and correct conservativity principle violations in practice. We have characterised the conservativity principle problem, following the assumption of disjointness, as a consistency principle problem. We have also presented an elegant and scalable way to detect and repair violations in the Horn propositional case. Thus, our method is incomplete and it may fail to detect and repair all violations. However, the conducted evaluation suggests that our method produces competitive results in practice. In the close future we plan to consider extensions of the current projection to Horn propositional logic while keeping the nice scalability properties of the current method.

The implemented method follows a "better safe than sorry" approach, which we currently consider suitable for the Optique project since we do not want ontology-to-ontology mappings to lead to unexpected results for the OBDA queries, as motivated in Section 3. Hence, we currently delegate complex relationhips between ontology entities and the database to the (hand-crafted) schema-to-ontology mappings, which will also play an important role in Optique. Nevertheless we do not discard in the future to explore alternative methods to detect and repair conservative principle violations. In particular, we plan to study the potential application of approaches based on graph-theory, in order to extend the detection and repair of conservativity principle violations. Strongly connected compontents of a graph representation of the subsumption relation between named concepts (as defined in [29]), for instance, may be used to capture violations between pairs of concepts already involved in a subsumption relationship.

Additionally, we will also consider exploring the use of learning techniques for the addition of disjointness axioms [45], and to involve the domain experts in the assessment/addition of such disjointness [18, 12]. This manual assessment may also be used to consider violations as false positives, as proposed in [26, 14, 27], and suggest them as candidate extensions of the input ontologies.

We consider that the proposed method has also potential in scenarios others than Optique. For instance, the authors in [28] apply ontology matching in a multi-agent system scenario in order to allow the exchange and extension of ontology-based *action plans* among agents. In such a context, violations of the conservativity principle should be taken into account and highly critical tasks should not be performed if violations are detected. In [44], authors present an ontology-based data integration (OBDI) system, which integrates ontology mapping and query reformulation techniques. As in Optique, mappings violating the conservativity principle may compromise the quality of the query results in the proposed OBDI system.

Finally, we have short-term plans for deployment in the Optique industry partners Statoil and Siemens. The techniques described in this paper have already been integrated within the "ontology and mapping management module" (see [24] for details about the Optique architecture).

## Acknowledgements

This work was supported by the EU FP7 IP project Optique (no. 318338), the MIUR project CINA (Compositionality, Interaction, Negotiation, Autonomicity for the future ICT society) and the EPSRC project Score!. We also thank the unvaluable help provided by Bernardo Cuenca and Ian Horrocks. Finally, we are also very grateful for the support of the Optique colleagues that facilitated our understanding of the domain, especially: Dag Hovland, Evgeny Kharlamov, Dmitry Zheleznyakov and Martin G. Skjæveland.

## References


1. Agrawal, R., Borgida, A., Jagadish, H.V.: Efficient Management of Transitive Relationships in Large Data and Knowledge Bases. In: ACM SIGMOD Conf. on Manag. of Data (1989)
2. Beisswanger, E., Hahn, U., et al.: Towards valid and reusable reference alignmentsten basic quality checks for ontology alignments and their application to three different reference data sets. J. Biomed. Semant. 3(Suppl 1), S4 (2012)
3. Bodenreider, O.: The Unified Medical Language System (UMLS): integrating biomedical terminology. Nucleic Acids Research 32, 267–270 (2004)
4. Borgida, A., Serafini, L.: Distributed Description Logics: Assimilating Information from Peer Sources. J. Data Sem. 1, 153–184 (2003)
5. Christophides, V., Plexousakis, D., Scholl, M., Tourtounis, S.: On Labeling Schemes for the Semantic Web. In: Int'l World Wide Web Conf. (WWW). pp. 544–555 (2003)
6. Cuenca Grau, B., Horrocks, I., Motik, B., Parsia, B., Patel-Schneider, P.F., Sattler, U.: OWL 2: The next step for OWL. J. Web Sem. 6(4), 309–322 (2008)
7. Cuenca Grau, B., Horrocks, I., Kazakov, Y., Sattler, U.: Modular Reuse of Ontologies: Theory and Practice. J. Artif. Intell. Res. 31, 273–318 (2008)
8. David, J., Euzenat, J., Scharffe, F., Trojahn, C.: The Alignment API 4.0. J. Sem. Web 2(1), 3–10 (2011)
9. Dowling, W.F., Gallier, J.H.: Linear-Time Algorithms for Testing the Satisfiability of Propositional Horn Formulae. J. Log. Prog. 1(3), 267–284 (1984)
10. Euzenat, J.: Semantic Precision and Recall for Ontology Alignment Evaluation. In: Int'l Joint Conf. on Artif. Intell. (IJCAI). pp. 348–353 (2007)
11. Euzenat, J., Meilicke, C., Stuckenschmidt, H., Shvaiko, P., Trojahn, C.: Ontology Alignment Evaluation Initiative: Six Years of Experience. J. Data Sem. 15, 158–192 (2011)
12. Ferré, S., Rudolph, S.: Advocatus Diaboli - Exploratory Enrichment of Ontologies with Negative Constraints. In: Int'l Conf. on Knowl. Eng. (EKAW). pp. 42–56 (2012)
13. Grau, B.C., Dragisic, Z., Eckert, K., et al.: Results of the Ontology Alignment Evaluation Initiative 2013. In: Ontology Matching (OM) (2013)
14. Ivanova, V., Lambrix, P.: A Unified Approach for Aligning Taxonomies and Debugging Taxonomies and their Alignments. In: Eur. Sem. Web Conf. (ESWC), pp. 1–15. Springer (2013)
15. Jean-Mary, Y.R., Shironoshita, E.P., Kabuka, M.R.: Ontology Matching With Semantic Verification. J. Web Sem. 7(3), 235–251 (2009)
16. Jiménez-Ruiz, E., Cuenca Grau, B., Horrocks, I.: On the feasibility of using OWL 2 DL reasoners for ontology matching problems. In: OWL Reasoner Evaluation Workshop (2012)
17. Jiménez-Ruiz, E., Cuenca Grau, B.: LogMap: Logic-based and Scalable Ontology Matching. In: Int'l Sem. Web Conf. (ISWC). pp. 273–288 (2011)
18. Jiménez-Ruiz, E., Cuenca Grau, B., Horrocks, I., Berlanga, R.: Ontology integration using mappings: Towards getting the right logical consequences. In: Eur. Sem. Web Conf. (2009)
19. Jiménez-Ruiz, E., Cuenca Grau, B., Horrocks, I., Berlanga, R.: Logic-based Assessment of the Compatibility of UMLS Ontology Sources. J. Biomed. Semant. 2(Suppl 1), S2 (2011)
20. Jiménez-Ruiz, E., Cuenca Grau, B., Zhou, Y., Horrocks, I.: Large-scale Interactive Ontology Matching: Algorithms and Implementation. In: Eur. Conf. on Artif. Intell. (ECAI) (2012)
21. Jiménez-Ruiz, E., Meilicke, C., Grau, B.C., Horrocks, I.: Evaluating Mapping Repair Systems with Large Biomedical Ontologies. In: Description Logics. pp. 246–257 (2013)
22. Kalyanpur, A., Parsia, B., Horridge, M., Sirin, E.: Finding all justifications of OWL DL entailments. Int'l Sem. Web Conf. (ISWC) pp. 267–280 (2007)
23. Kazakov, Y., Krötzsch, M., Simancik, F.: Concurrent Classification of EL Ontologies. In: Int'l Sem. Web Conf. (ISWC). pp. 305–320 (2011)

24. Kharlamov, E., Jiménez-Ruiz, E., Zheleznyakov, D., et al.: Optique: Towards OBDA Systems for Industry. In: Eur. Sem. Web Conf. (ESWC) Satellite Events. pp. 125–140 (2013)
25. Konev, B., Walther, D., Wolter, F.: The Logical Difference Problem for Description Logic Terminologies. In: Int'l Joint Conf. on Automated Reasoning (IJCAR). pp. 259–274 (2008)
26. Lambrix, P., Dragisic, Z., Ivanova, V.: Get My Pizza Right: Repairing Missing Is-a Relations in $\mathcal{ALC}$ Ontologies. In: Semantic Technology, pp. 17–32. Springer (2013)
27. Lambrix, P., Liu, Q.: Debugging the Missing Is-a Structure Within Taxonomies Networked by Partial Reference Alignments. Data Knowl. Eng. (DKE) 86, 179–205 (2013)
28. Mascardi, V., Ancona, D., Barbieri, M., Bordini, R.H., Ricci, A.: CooL-AgentSpeak: Endowing AgentSpeak-DL Agents with Plan Exchange and Ontology Services. Web Intelligence and Agent Systems 12(1), 83–107 (2014)
29. Meilicke, C.: Alignments Incoherency in Ontology Matching. Ph.D. thesis, University of Mannheim (2011)
30. Meilicke, C., Völker, J., Stuckenschmidt, H.: Learning Disjointness for Debugging Mappings between Lightweight Ontologies. In: Int'l Conf. on Knowl. Eng. (EKAW). pp. 93–108 (2008)
31. Melnik, S., Garcia-Molina, H., Rahm, E.: Similarity Flooding: A Versatile Graph Matching Algorithm and Its Application to Schema Matching. In: IEEE Int'l Conf. on Data Eng. (2002)
32. Motik, B., Shearer, R., Horrocks, I.: Hypertableau Reasoning for Description Logics. J. Artif. Intell. Res. (JAIR) 36, 165–228 (2009)
33. Nebot, V., Berlanga, R.: Efficient Retrieval of Ontology Fragments Using an Interval Labeling Scheme. Inf. Sci. 179(24), 4151–4173 (2009)
34. Ngo, D., Bellahsene, Z.: YAM++ : A Multi-strategy Based Approach for Ontology Matching Task. In: Int'l Conf. on Knowl. Eng. (EKAW). pp. 421–425 (2012)
35. Pesquita, C., Faria, D., Santos, E., Couto, F.M.: To repair or not to repair: reconciling correctness and coherence in ontology reference alignments. In: Ontology Matching (OM) (2013)
36. Reiter, R.: A Theory of Diagnosis from First Principles. Artif. Intell. 32(1) (1987)
37. Santos, E., Faria, D., Pesquita, C., Couto, F.: Ontology Alignment Repair Through Modularization and Confidence-based Heuristics. arXiv:1307.5322 preprint (2013)
38. Schlobach, S.: Debugging and Semantic Clarification by Pinpointing. In: Eur. Sem. Web Conf. (ESWC), pp. 226–240. Springer (2005)
39. Schlobach, S., Cornet, R.: Non-standard Reasoning Services for the Debugging of Description Logic Terminologies. In: Int'l Joint Conf. on Artif. Intell. (IJCAI). pp. 355–362 (2003)
40. Shvaiko, P., Euzenat, J.: Ontology Matching: State of the Art and Future Challenges. IEEE Transactions on Knowl. and Data Eng. (TKDE) (2012)
41. Skjæveland, M.G., Lian, E.H., Horrocks, I.: Publishing the Norwegian Petroleum Directorate's FactPages as Semantic Web Data. In: Int'l Sem. Web Conf. (ISWC) (2013)
42. Soylu, A., et al.: A Preliminary Approach on Ontology-Based Visual Query Formulation for Big Data. In: 7th Research Conf. on Metadata and Semantics Research (MTSR) (2013)
43. Suntisrivaraporn, B., Qi, G., Ji, Q., Haase, P.: A Modularization-Based Approach to Finding All Justifications for OWL DL Entailments. In: Asian Sem. Web Conf. (ASWC) (2008)
44. Tian, A., Sequeda, J., Miranker, D.P.: QODI: Query as Context in Automatic Data Integration. In: Int'l Sem. Web Conf. (ISWC). pp. 624–639 (2013)
45. Völker, J., Vrandecic, D., Sure, Y., Hotho, A.: Learning Disjointness. In: Eur. Sem. Web Conf. (ESWC). pp. 175–189 (2007)
46. Šváb, O., Svátek, V., Berka, P., Rak, D., Tomášek, P.: OntoFarm: Towards an Experimental Collection of Parallel Ontologies. In: Int'l Sem. Web Conf. (ISWC). Poster Session (2005)
47. Wang, P., Xu, B.: Debugging Ontology Mappings: A Static Approach. Computing and Informatics 27(1), 21–36 (2012)
48. Zhang, S., Mork, P., Bodenreider, O.: Lessons Learned from Aligning two Representations of Anatomy. In: Int'l Conf. on Principles of Knowl. Repr. and Reasoning (KR) (2004)