

An Algebra of Qualitative Taxonomical Relations for Ontology Alignments

Armen Inants and Jérôme Euzenat

Inria & Univ. Grenoble Alpes, Grenoble, France
{Armen.Inants, Jerome.Euzenat}@inria.fr

Abstract. Algebras of relations were shown useful in managing ontology alignments. They make it possible to aggregate alignments disjunctively or conjunctively and to propagate alignments within a network of ontologies. The previously considered algebra of relations contains taxonomical relations between classes. However, compositional inference using this algebra is sound only if we assume that classes which occur in alignments have nonempty extensions. Moreover, this algebra covers relations only between classes. Here we introduce a new algebra of relations, which, first, solves the limitation of the previous one, and second, incorporates all qualitative taxonomical relations that occur between individuals and concepts, including the relations “is a” and “is not”. We prove that this algebra is coherent with respect to the simple semantics of alignments.

Keywords: relation algebra, ontology alignment, network of ontologies.

1 Introduction

The heterogeneity of ontologies on the semantic web requires finding correspondences between them in order to achieve semantic interoperability. The operation of finding correspondences is called ontology matching and its result is a set of correspondences called an alignment [6]. Alignments are used for importing data from one ontology to another or for translating queries.

In previous work [5], we put forward a framework for manipulating alignments based on algebras of relations. This allows for merging alignments conjunctively or disjunctively, amalgamate alignments with relations of different granularity and compose alignments.

The general approach was illustrated in [5] on a particular algebra \mathbb{A}_5 . It is generated by 5 atoms: $=, >, <, \overline{\cap}, \perp$, which stand for “equivalent to”, “more/less general than”, “partially overlaps with” and “disjoint with” respectively. The composition table of \mathbb{A}_5 is given in Table 1. It was shown that an algebra of relations induces composition, union, intersection and conversion operations on alignments.

This may be particularly useful as a fast way to reason about alignments without resorting to full reasoning. For instance, this may be used for generating new alignments from existing ones or for checking the unsatisfiability of a network of ontologies.

Example 1. For instance, in Figure 1, there are two correspondences: “ \mathcal{O}_2 :Serial writer is subsumed by \mathcal{O}_1 :Successful creator” and “ \mathcal{O}_2 :Serial writer is equivalent to \mathcal{O}_3 :Popular writer”. Subsumption and Equivalence are encoded in $\mathbb{A5}$ as $\{=, <\}$ and $\{=\}$ respectively. By composing these relations we infer a correspondence between \mathcal{O}_1 :Successful creator and \mathcal{O}_3 :Popular writer:

$$\begin{aligned}
& (\text{Successful creator, Serial writer, } \{=\}) * (\text{Serial writer, Popular writer, } \{=, <\}) \\
& = (\text{Successful creator, Popular writer, } \{=\} * \{=, <\}) \\
& = (\text{Successful creator, Popular writer, } (= * =) \cup (= * <)) \\
& = (\text{Successful creator, Popular writer, } \{=\} \cup \{<\}) \\
& = (\text{Successful creator, Popular writer, } \{=, <\})
\end{aligned}$$

However, the algebra of relations $\mathbb{A5}$ suffers from two problems:

1. $\mathbb{A5}$ covers relations only between classes. This leaves out of scope the relations `owl:sameAs` (noted $=$), `owl:differentFrom` (noted \neq), which are defined between instances, and the instance-class relation `rdf:type` (\in). Compositional reasoning with these relations may be used for debugging link sets as shown by Example 1.

Example 2. In Figure 1, composing $\{<, =\} * \{=\} * \{\perp\}$ is equivalent to $\{\perp\}$, i.e., “Mystery novelist” and “Academic” are disjoint classes. This leaves aside further relation composition. Indeed, one would like that $\{\in\} * \{\perp\} * \{\ni\}$ actually yields $\{\neq\}$, i.e., “Amanda Cross” is different from “Carolyn Gold Heilbrun”. This would be very useful for debugging data sets since the actual relation between these individuals is $\{=\}$ so the intersection of these relations is empty revealing unsatisfiability.

However, this requires to compose class relations (\perp) and individual-class relations (\in). Moreover, this composition yields an individual relation (\neq).

To make this work within the considered framework, one needs an algebra incorporating all these relations. This would allow for encoding such RDF triples as correspondences and use them for the refinement and evolution of alignments.

2. The algebraic calculus that $\mathbb{A5}$ induces on alignments does not allow for distinguishing between unsatisfiability and incoherence of alignments. An alignment

Table 1: Composition table of $\mathbb{A5}$.

*	=	>	<	\emptyset	\perp
=	=	>	<	\emptyset	\perp
>	>	>	$=><\emptyset$	$>\emptyset$	$>\emptyset\perp$
<	<	$=><\emptyset\perp$	<	$<\emptyset\perp$	\perp
\emptyset	\emptyset	$>\emptyset\perp$	$<\emptyset$	$=><\emptyset\perp$	$>\emptyset\perp$
\perp	\perp	\perp	$<\emptyset\perp$	$<\emptyset\perp$	$=><\emptyset\perp$

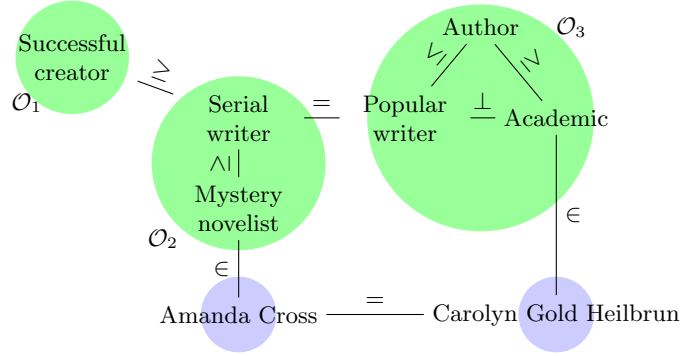


Fig. 1: An example of unsatisfiability in a linked data sets that can be detected through simple composition of relations across ontologies, data, links and correspondences.

is satisfiable if it has a model, and coherent, if it does not force incoherence on any of its entities. If, by applying algebraic reasoning on alignments, we deduce a correspondence (C, D, \emptyset) , then it means that the alignments are algebraically inconsistent. However, algebraic inconsistency does not imply unsatisfiability, as one would expect. This is illustrated in Example 3.

Example 3. Consider an alignment \mathcal{A} with two correspondences between the same pair of entities: $\mu = (C, D, \{\perp\})$ and $\nu = (C, D, \{<, =\})$. Their conjunction is equal to (C, D, \emptyset) :

$$\mu \wedge \nu = (C, D, \{\perp\}) \wedge (C, D, \{<, =\}) = (C, D, \{\perp\} \cap \{<, =\}) = (C, D, \emptyset).$$

This means that \mathcal{A} is algebraically inconsistent. But \mathcal{A} has models, thus it is not unsatisfiable. Indeed, if C is interpreted as the empty set, then, whatever the interpretation of D , both μ and ν are satisfied by this interpretation. However, \mathcal{A} is incoherent, since it does not allow the class C to have instances.

In this paper, we introduce a new algebra of relations $\mathbb{A}16$, which solves these limitations of $\mathbb{A}5$. $\mathbb{A}16$ incorporates the relations “same as” (`owl:sameAs`), “different from” (`owl:differentFrom`), “is a” (`rdf:type`), “is not”, “equivalent to” (`owl:equivalentClass`), “subsumed by” (`rdfs:subClassOf`), “disjoint with” (`owl:disjointWith`), “partially overlaps with” in compliance with OWL semantics. The calculus that $\mathbb{A}16$ induces on alignments allows to differentiate between unsatisfiability and incoherence of alignments.

The paper is structured as follows. In Section 2, we discuss the related work. Section 3 covers some preliminaries, including networks of ontologies and algebras of relations. In Section 4, we build the algebra $\mathbb{A}16$ and establish its soundness with respect to the direct semantics of alignments. In Section 5, we discuss some changes that $\mathbb{A}16$ brings to the calculus of alignments.

2 Related work

This paper is related to formal frameworks for distributed ontologies on the one hand, and to the theory of qualitative calculi on the other hand.

Languages for distributed ontologies There are several languages that allow for expressing relations across ontologies. Among them are Distributed Description Logics (DDL) [3], ε -connections [12], Package-based Description Logics (P-DL) [2], Integrated Distributed Description Logics (IDDL) [19] and Distributed Ontology Language (DOL) [14].

Mappings between ontologies in DDL assert relations from the perspective of the target ontology. Mappings between concepts are expressed as *bridge rules*, and those between individuals as *individual correspondences*. The key feature of DDL reasoning is subsumption propagation from one ontology to another. Subsumption is not transitive in DDL, thus cannot be propagated by composition. ε -connections are a framework for modular ontologies. Connection between ontological modules are established with *links*, which act as inter-ontology properties. A distinctive feature of ε -connections is that each ontology module is supposed to model a portion of the domain that is complementary and non-overlapping with respect to the other ontology modules. As the domains of ontologies in an ε -connection system must be disjoint, it is not possible to have a concept in some ontology module that has subconcepts or instances in another ontology. *Ontology importing*, which is implemented in P-DL, allows for reusing concepts, relations and individuals defined in one ontology inside another ontology. *Alignments* in IDDL constitute a separate layer and can be regarded independently from ontologies. This makes possible to reason about alignments alone, considering them as first class citizens. Some comparative analysis of DDL, ε -connections, P-DL and IDDL can be found in [10, 20].

An algebraic calculus of alignments is not intended as a proof theory for a particular semantics of alignments. It is a framework, which allows to use custom algebras of relations for inducing operations on alignments. In this paper, we limit ourselves with taxonomical relations between classes and instances. We design the algebra of taxonomical relations to be sound with respect to the OWL semantics of relations [7]. In principle, algebras of ontology alignment relations can be designed in compliance with other semantics as well, e.g., with the integrated semantics used in IDDL.

Qualitative calculi Algebras of relations have been used in knowledge representation and reasoning, particularly in the spatio-temporal domain, since the pioneering work of Allen [1]. Allen considered the universe of time intervals, formalized as pairs of real-valued endpoints, and defined 13 binary relations on that universe. Between any two time intervals one and only one out of the 13 base relations holds. This important property allows to factorize the infinite Boolean algebra of binary relations over the universe into a finite Boolean algebra of (qualitative) relations of interest. Moreover, these 13 base relations and their arbitrary unions are closed under composition and converse. This allowed Allen

to induce composition and converse on the Boolean algebra of relation symbols, yielding an algebra called the *interval algebra*.

Allen put forward an algorithm for reasoning over relations between time intervals based on constraint propagation. This algorithm decides the satisfiability of a network of temporal constraints, consisting of variables ranging over the universe of time intervals, and binary constraints on variables expressed by elements of the interval algebra.

The interval algebra is an instance of Tarski's relation algebra [18]. [13] defined a class of *qualitative binary constraint satisfaction problems* (QBCSP), which is a class of binary constraint satisfaction problems abstracted to relation algebras. They generalized the *constraint propagation* algorithm of Allen to an arbitrary relation algebra. Reasoning with relation algebras is studied in [4, 9].

Meanwhile, calculi similar to Allen's were developed in both temporal and spatial domains. Among them are two variants of the *Region Connection Calculus*: RCC5 and RCC8. In order to study the properties of RCC5, RCC8 and many other calculi existing by that time within a single framework, Ligozat and Renz [15] proposed a formal definition of the implicit concept of a qualitative calculus. In the framework of Ligozat and Renz, a qualitative calculus arises from a partition scheme over some universe. Such calculi can have a weaker algebraic structure than relation algebras: they can be non-associative.

The development of this paper was initially based on the framework of Ligozat and Renz. However, the algebra $\mathbb{A}16$ that we construct does not arise from a partition scheme, but from a more general construct which we call a general partition scheme. We extend the result of Ligozat and Renz to general partition schemes.

3 Preliminaries

3.1 Networks of ontologies

Here we give a logical account of networks of ontologies in the sense of [6].

Definition 1 (Correspondence). *Given two ontologies \mathcal{O} and \mathcal{O}' , with associated entity languages $Ent(\mathcal{O})$ and $Ent(\mathcal{O}')$, and a set of alignment relations \mathbf{R} , a correspondence is a triple (e, e', r) , such that $e \in Ent(\mathcal{O})$, $e' \in Ent(\mathcal{O}')$ and $r \in \mathbf{R}$.*

A correspondence (e, e', r) is an assertion that a certain pragmatic relation denoted by the symbol r holds between the entities e and e' .

The entities can be restricted to a particular kind of terms of the ontology language based on the ontology vocabulary, e.g., named entities. The entity language can also be an extension of the ontology language. For instance, it can be a query language, such as SPARQL [8], adding operations for manipulating ontology entities that are not available in the ontology language itself, like concatenating strings or joining relations. The developments of this paper are independent of the chosen entity language.

An important component of a correspondence is the relation that holds between the entities. We fix a set of relations \mathbf{R} that is used for expressing the relations between the entities. The set \mathbf{R} can contain relation symbols like $=$, which is used by matching algorithms, or IRIs like <http://www.w3.org/2004/02/skos/extensions#broaderPartitive>. Relations from ontology languages, such as `owl:sameAs`, `owl:differentFrom`, `owl:equivalentClass`, `owl:disjointWith`, `rdfs:subClassOf` or `rdf:type`, can also be used.

An alignment is defined as a set of correspondences.

Definition 2 (Alignment). *Given two ontologies \mathcal{O} and \mathcal{O}' , an alignment is a set of correspondences between pairs of entities belonging to $Ent(\mathcal{O})$ and $Ent(\mathcal{O}')$ respectively.*

Definition 3 (Network of ontologies). *A network of ontologies (Ω, Λ) is made up of a set Ω of ontologies and a set Λ of alignments between these ontologies. We denote by $\Lambda(\mathcal{O}, \mathcal{O}')$ the set of alignments in Λ between \mathcal{O} and \mathcal{O}' .*

A correspondence is interpreted with respect to three features: a pair of models from each ontology and a semantic structure, denoted as Δ . The class of models of an ontology \mathcal{O} is denoted as $\mathcal{M}(\mathcal{O})$.

Definition 4 (Satisfied correspondence). *A correspondence $\mu = (e, e', r)$ is satisfied by two models m, m' of $\mathcal{O}, \mathcal{O}'$ for some semantic structure Δ if and only if $(m(e), m'(e')) \in r^\Delta$, such that r^Δ provides the interpretation of the relation r in the structure. This is denoted by $m, m' \models_\Delta \mu$.*

Three different kinds of semantic structures are outlined in [21]: simple, contextualized and integrated. Let us fix two ontologies \mathcal{O}_1 and \mathcal{O}_2 and their models m_1 and m_2 with domains of interpretation D_1 and D_2 respectively. An integrated semantic structure consists of functions ε_i from the local domains D_i ($i = 1, 2$) to a global domain D . A simple semantic structure is a particular case of integrated structure: when $D = \cup_i D_i$ and ε_i are canonical embeddings of D_i into D . Contextualized semantics is given by a family of binary relations r_{ij} ($i = 1, 2$) between the local domains D_i and D_j .

Below is an example of how relation symbols are interpreted with respect to each semantics. As an example consider the semantics of the relation symbol \sqsubseteq depending on Δ .

$$\begin{aligned} \sqsubseteq^{simple(\Delta)} &= \{(X, Y) : X \subseteq D_1, Y \subseteq D_2 \text{ and } X \subseteq Y\} \\ \sqsubseteq^{integrated(\Delta)} &= \{(X, Y) : X \subseteq D_1, Y \subseteq D_2 \text{ and } \varepsilon_1(X) \subseteq \varepsilon_2(Y)\} \\ \sqsubseteq^{contextual(\Delta)} &= \{(X, Y) : X \subseteq D_1, Y \subseteq D_2 \text{ and } r_{12}(X) \subseteq Y\} \end{aligned}$$

If Δ is simple, then \sqsubseteq^Δ depends only on D_1 and D_2 . In this case the semantics of \sqsubseteq corresponds to the interpretation of `rdfs:subClassOf` if we consider \mathcal{O}_1 and \mathcal{O}_2 as one large ontology. Likewise, the simple semantics of relation symbols \perp (disjointness) and $=$ (equivalence) corresponds to `owl:disjointWith` and `owl:equivalentClass`.

Definition 5 (Models of alignments). *Given two ontologies \mathcal{O} and \mathcal{O}' and an alignment A between these ontologies, a model of this alignment is a triple (m, m', Δ) with $m \in \mathcal{M}(\mathcal{O})$, $m' \in \mathcal{M}(\mathcal{O}')$, and Δ a semantic structure, such that $\forall \mu \in A, m, m' \models_{\Delta} \mu$ (denoted by $m, m' \models_{\Delta} A$).*

An alignment is said to be *satisfiable* if it has a model. An alignment is said to be *coherent* if, for any of its class entities, it has a model that makes this class non empty.

3.2 Algebraic calculus of ontology alignments

It was shown that algebras of relations are useful for managing ontologies [5]. The adopted algebraic formalism is Tarskian relation algebras.

Definition 6 (Relation algebra). *A relation algebra is an algebra*

$$\mathbb{A} = (A, +, \cdot, -, 0, 1, ;, \smile, 1'), \quad (3.1)$$

with binary operations $+$ (Boolean sum), \cdot (Boolean product) and $;$ (composition, or relative product), unary operations $-$ (complement) and \smile (converse), and constants $0, 1, 1' \in A$ called zero, unit and identity respectively, such that

- 1) *the reduct $(A, +, \cdot, -, 0, 1)$ is a Boolean algebra,*
- 2) *identity: $1'; x = x; 1' = x$, for all $x \in A$,*
- 3) *Peircean law: $(x; y) \cdot z = 0 \Leftrightarrow (x \smile; z) \cdot y = 0 \Leftrightarrow x \cdot (z; y \smile) = 0$, for all $x, y, z \in A$,*
- 4) *associativity: $(x; y); z = x; (y; z)$, for all $x, y, z \in A$.*

We will denote by \mathbb{A} both the algebra and its carrier set. The class of relation algebras is denoted as RA. In the sequel, by “algebra of relations” we will imply an instance of RA, if not stated otherwise.

Consider an algebra of relations \mathbb{A} . The approach put forward in [5] is that we allow any element of \mathbb{A} to be used in a correspondence. In other words, referring to the previous subsection, we take $\mathbf{R} = \mathbb{A}$.

Each alignment may be normalized through *norm* to contain exactly one correspondence between any two entities. \mathbb{A} induces the following operations on alignments:

$$\mathcal{A} \wedge \mathcal{A}' = \text{norm}(\mathcal{A} \cup \mathcal{A}') \quad (3.2)$$

$$\mathcal{A} \vee \mathcal{A}' = \{(e, e', r + r') : (e, e', r) \in \text{norm}(\mathcal{A}) \wedge (e, e', r') \in \text{norm}(\mathcal{A}')\} \quad (3.3)$$

$$\mathcal{A} \smile = \{(e', e, r \smile) : (e, e', r) \in \mathcal{A}\} \quad (3.4)$$

If there exists an alignment between ontology \mathcal{O} and ontology \mathcal{O}' , and another alignment between \mathcal{O}' and a third ontology \mathcal{O}'' , we would like to find which correspondences hold between \mathcal{O} and \mathcal{O}'' . The operation that returns this set of correspondences is called composition.

$$\mathcal{A} \circ \mathcal{A}' = \text{norm}(\{(e, e'', r; s) : \exists (e, e', r) \in \mathcal{A} \text{ and } \exists (e', e'', s) \in \mathcal{A}'\}) \quad (3.5)$$

We can regard a network of ontologies as a directed graph, with ontologies being vertices and alignments being edges. Moreover, one can assume that there is at most one alignment between any pair of ontologies in the network. A *closure* of a network of ontologies can be computed by applying a path-consistency algorithm, e.g., PC2 [16], which in essence is an iterative application of

$$\mathcal{A}_{\mathcal{O}_i, \mathcal{O}_j} \leftarrow \mathcal{A}_{\mathcal{O}_i, \mathcal{O}_j} \wedge (\mathcal{A}_{\mathcal{O}_i, \mathcal{O}_k} \circ \mathcal{A}_{\mathcal{O}_k, \mathcal{O}_j}), \quad (3.6)$$

for every triple $(\mathcal{O}_i, \mathcal{O}_j, \mathcal{O}_k)$ of ontologies in Λ , until a fixed point is reached.

3.3 Algebras of JEPD binary relations

In general terms, a connection between an algebra of relations and some domain of knowledge is given by specifying a *universe* of objects and a set of *base relations*. For example, the universe of Allen's interval algebra is the set of time intervals encoded as pairs of real numbers (x, y) , where $x > y$, whereas base relations are defined by certain conditions on the endpoints of two intervals. Here we give an account of algebras that arise from a set of jointly exhaustive and pairwise disjoint (JEPD) binary relations.

A *binary relation* over a nonempty set \mathcal{U} is a subset of the Cartesian product $\mathcal{U} \times \mathcal{U}$. The *converse* (also called inverse) of a binary relation R is a relation symmetric to R , defined as $R^{-1} = \{(x, y) : (y, x) \in R\}$. The relation $Id_{\mathcal{U}} = \{(x, y) \in \mathcal{U} \times \mathcal{U} : x = y\}$ is called the *identity* over \mathcal{U} . *Composition* of binary relations R and S is defined as $R \circ S = \{(x, y) \in \mathcal{U} \times \mathcal{U} : \exists z \in \mathcal{U} \text{ such that } (x, z) \in R \text{ and } (z, y) \in S\}$. The *field* of a binary relation R is defined as $Fd(R) = \{x \in \mathcal{U} : \exists y, (x, y) \in R \text{ or } (y, x) \in R\}$.

A set \mathcal{P} of binary relations over \mathcal{U} is called *jointly exhaustive and pairwise disjoint*, if $\cup_{R \in \mathcal{P}} R = \mathcal{U} \times \mathcal{U}$ and $R \cap R' = \emptyset$ for each $R \neq R' \in \mathcal{P}$. Such \mathcal{P} is called a *partition* of $\mathcal{U} \times \mathcal{U}$. We will assume \mathcal{P} to be finite. Relations in \mathcal{P} are called *base relations*. We call an arbitrary union of base relations a \mathcal{P} -*relation*. \mathcal{P} -relations form a subalgebra of the Boolean algebra $\mathcal{U} \times \mathcal{U}$, in which the base relations are atoms. If a \mathcal{P} -relation is a union of two or more base relations, it is called a *disjunctive relation*. Each \mathcal{P} -relation is identified by the set of constituting base relations, thus there is a one-to-one correspondence between the set of \mathcal{P} -relations and the powerset $2^{\mathcal{P}}$. In the sequel a set of base relations $\{R_1, \dots, R_n\} \subseteq \mathcal{P}$ will denote their union.

Generally speaking, composition of \mathcal{P} -relations may not be a \mathcal{P} -relation. In other words, \circ may not be closed on the set of \mathcal{P} -relations. One can approximate the composition by a so-called *weak composition* \diamond , defined as the least \mathcal{P} -relation which contains the composition. Weak composition is a binary operation on \mathcal{P} -relations ($\diamond : 2^{\mathcal{P}} \times 2^{\mathcal{P}} \rightarrow 2^{\mathcal{P}}$).

A *partition scheme* is a pair $(\mathcal{U}, \mathcal{P})$, where \mathcal{U} is a nonempty set and $\mathcal{P} = (R_i)_{i \in I}$ is a partition of $\mathcal{U} \times \mathcal{U}$, which is closed under converse and contains the identity over \mathcal{U} . Given a partition scheme, both weak composition and converse become operations on the Boolean algebra $2^{\mathcal{P}}$. The algebra

$$\mathbb{A}_{\mathcal{P}} = (2^{\mathcal{P}}, \cup, \cap, -_{\mathcal{U} \times \mathcal{U}}, \emptyset, \mathcal{P}, \diamond, ^{-1}, Id_{\mathcal{U}}) \quad (3.7)$$

is said to be generated by \mathcal{P} .

Proposition 1 (Ligozat and Renz [15]). *If $(\mathcal{U}, \mathcal{P})$ is a partition scheme, then $\mathbb{A}_{\mathcal{P}}$ satisfies all axioms of RA, except, possibly, the associativity axiom.*

An algebra of relations which satisfies all axioms of RA except possibly the associativity axiom is called a *non-associative algebra* [17]. The class of non-associative algebras NA is broader than RA. Assume $\mathbb{A} = (A, +, \cdot, -, 0, 1, ;, \checkmark, 1')$ \in NA, $x \leq y$ is used as a shortcut for $x + y = y$. By $At(\mathbb{A})$ we denote the set of atoms of the Boolean reduct of \mathbb{A} .

Any finite non-associative algebra \mathbb{A} is fully specified by its *atom structure*. The atom structure of an algebra \mathbb{A} consists of the set of atoms $At(\mathbb{A})$, the set of identity atoms $At(1') \subseteq At(\mathbb{A})$, the converse restricted to atoms $\checkmark : At(\mathbb{A}) \rightarrow At(\mathbb{A})$ and the composition restricted to atoms, which is a function $CT : At(\mathbb{A}) \times At(\mathbb{A}) \rightarrow 2^{At(\mathbb{A})}$ defined by $z \in CT(x, y)$ iff $x; y \leq z$. CT is usually specified by a *composition table*. The triples (x, y, z) , where x, y and z are atoms and which satisfy $x; y \leq z$ are called *consistent scenarios*. In order to find the consistent scenarios of $\mathbb{A}_{\mathcal{P}}$, one has to find all triples (R_i, R_j, R_k) of base relations, for which $\exists x, y, z$, such that $(x, y) \in R_i$, $(y, z) \in R_j$ and $(x, z) \in R_k$.

A non-associative algebra is said to be *integral* if the composition of any non-zero elements, i.e., those different from 0, is non-zero. A finite $\mathbb{A}_{\mathcal{P}}$ is integral iff the composition of any two base relations is not equal to the empty set.

4 An algebra of qualitative taxonomical relations

In this section, we define an algebra, which contains the relations “equivalent to”, “subsumed by”, “disjoint with”, “same as”, “different from”, “is a” and “is not”. We call an ontology alignment relation *taxonomical*, if it is associated with some set-theoretic relation (predicate) R . For instance, subsumption \sqsubseteq is associated with the set-theoretic inclusion \subseteq . A taxonomical relation holds between two ontological entities iff the relation R holds between the interpretations of these entities. A set-theoretic relation R is said to be *qualitative*, if, for any pair of sets (x, y) , xRy is characterized by 3 parameters: whether each of the sets $x \cap y$, $x \setminus y$, $y \setminus x$ is empty or not.

All relations listed above are taxonomical and qualitative (if interpreted with the simple semantics of alignments). The simple semantics of alignments assumes a common domain of interpretation for all ontologies in a network. Given an arbitrary infinite domain D , the relations “same as” and “different from” correspond to binary relations $=$ and \neq on D , “equivalent to”, “subsumed by” and “disjoint with” correspond to binary relations \equiv , \subseteq and \perp on 2^D , and finally “is a” and “is not” correspond to \in and \notin on $D \times 2^D$. Thus, the binary relations $=$, \neq , \equiv , \subseteq , \perp , \in and \notin are defined on the set $D \cup 2^D$, which we will call a universe and denote as $\mathcal{U}^{(D)}$. We will refer to the elements of D as *individuals*, and to the elements of 2^D as *sets*.

The relations

$$\mathbf{G} = \{=, \neq, \equiv, \subseteq, \supseteq, \perp, \in, \notin, \ni, \not\ni\}, \quad (4.1)$$

where $\supseteq = \subseteq^{-1}$, $\ni = \in^{-1}$ and $\not\supseteq = \not\subseteq^{-1}$, are not JEPD. What is the minimal partition \mathcal{P} of $\mathcal{U}^{(D)} \times \mathcal{U}^{(D)}$ that has relations in \mathbf{G} as \mathcal{P} -relations? Since $\mathcal{U}^{(D)} \times \mathcal{U}^{(D)}$ is a Boolean algebra, the question is to find its subalgebra generated by \mathbf{G} . Atoms of the sought-after subalgebra are nonempty intersections of generators (the elements of \mathbf{G} in our case) and their complements. This yields a partition \mathcal{P}^{14} with 14 base relations, which are defined below:

$$\begin{aligned}
=_{n}(\alpha, \beta) &\text{ iff } \alpha, \beta \text{ are nonempty sets and } \alpha = \beta \\
<(\alpha, \beta) &\text{ iff } \alpha, \beta \text{ are nonempty sets and } \alpha \subset \beta \\
>(\alpha, \beta) &\text{ iff } \alpha, \beta \text{ are nonempty sets and } \alpha \supset \beta \\
\setminus(\alpha, \beta) &\text{ iff } \alpha, \beta \text{ are sets and } \alpha \setminus \beta, \alpha \cap \beta, \beta \setminus \alpha \neq \emptyset \\
\parallel(\alpha, \beta) &\text{ iff } \alpha, \beta \text{ are nonempty sets and } \alpha \cap \beta = \emptyset \\
EN(\alpha, \beta) &\text{ iff } \alpha = \emptyset, \beta \text{ is a nonempty set} \\
NE(\alpha, \beta) &\text{ iff } \alpha \text{ is a nonempty set and } \beta = \emptyset \\
=_{e}(\alpha, \beta) &\text{ iff } \alpha = \beta = \emptyset \\
\in(\alpha, \beta) &\text{ iff } \alpha \text{ is an individual, } \beta \text{ is a set and } \alpha \in \beta \\
\ni(\alpha, \beta) &\text{ iff } \alpha \text{ is a set, } \beta \text{ is an individual and } \alpha \ni \beta \\
\notin(\alpha, \beta) &\text{ iff } \alpha \text{ is an individual, } \beta \text{ is a set and } \alpha \notin \beta \\
\ni\supseteq(\alpha, \beta) &\text{ iff } \alpha \text{ is a set, } \beta \text{ is an individual and } \alpha \ni\supseteq \beta \\
=_{i}(\alpha, \beta) &\text{ iff } \alpha, \beta \text{ are individuals and } \alpha = \beta \\
\neq_{i}(\alpha, \beta) &\text{ iff } \alpha, \beta \text{ are individuals and } \alpha \neq \beta
\end{aligned}$$

The relation \perp , for instance, is equal to the \mathcal{P}^{14} -relation $\{\parallel, EN, NE, =_{e}\}$. \mathcal{P}^{14} is closed under converse, but the identity relation $Id_{\mathcal{U}^{(D)}}$ does not belong to \mathcal{P}^{14} . It is a disjunctive \mathcal{P}^{14} -relation with three *identity atoms*: $=_{n}$, $=_{e}$ and $=_{i}$. Therefore, $(\mathcal{U}^{(D)}, \mathcal{P}^{14})$ is not a partition scheme. Below we define a broader class of partitions that includes \mathcal{P}^{14} .

Definition 7 (General partition scheme). Let $\mathcal{P} = (R_i)_{i \in I}$ be a partition of $\mathcal{U} \times \mathcal{U}$. $(\mathcal{U}, \mathcal{P})$ is called a general partition scheme if \mathcal{P} is closed under converse $^{-1}$ and the identity $Id_{\mathcal{U}}$ is a \mathcal{P} -relation (possibly disjunctive).

The following proposition ensures that the algebra $\mathbb{A}_{\mathcal{P}^{14}}$ (or simply $\mathbb{A}14$) generated by \mathcal{P}^{14} is a non-associative algebra.

Proposition 2. Let $\mathcal{P} = (R_i)_{i \in I}$ be a partition of $\mathcal{U} \times \mathcal{U}$, such that it is closed under converse and the identity $Id_{\mathcal{U}}$ is a \mathcal{P} -relation (possibly disjunctive). Then the algebra $\mathbb{A}_{\mathcal{P}}$ generated by \mathcal{P} is a non-associative algebra.

Proof. $Id_{\mathcal{U}} \diamond R_i = Id_{\mathcal{U}} \circ R_i = R_i = R_i \circ Id_{\mathcal{U}} = R_i \diamond Id_{\mathcal{U}}$. The Peircean law follows from the fact that $R_i \diamond R_j \cap R_k = \emptyset$ iff $R_i \circ R_j \cap R_k = \emptyset$ [see 15, Lemma 2].

The composition table of $\mathbb{A}_{\mathcal{P}^{14}}$ (or simply $\mathbb{A}14$) is given in Table 2. The presence of empty cells in the composition table means that the algebra $\mathbb{A}14$ is nonintegral.

Table 3: Composition tables of $\mathbb{A}16$.

\diamond	$=_n$	$>$	$<$	\emptyset	\parallel	\ni	$\not\exists_{ni}$	NE
$=_n$	$=_n$	$>$	$<$	\emptyset	\parallel	\ni	$\not\exists_{ni}$	NE
$<$	$<$	$=_n > < \emptyset \parallel$	$<$	$< \emptyset \parallel$	\parallel	$\ni \not\exists_{ni}$	$\not\exists_{ni}$	NE
$>$	$>$	$>$	$=_n > < \emptyset$	$> \emptyset$	$> \emptyset \parallel$	\ni	$\ni \not\exists_{ni}$	NE
\emptyset	\emptyset	$> \emptyset \parallel$	$< \emptyset$	$=_n > < \emptyset \parallel$	$> \emptyset \parallel$	$\ni \not\exists_{ni}$	$\ni \not\exists_{ni}$	NE
\parallel	\parallel	\parallel	$< \emptyset \parallel$	$< \emptyset \parallel$	$=_n > < \emptyset \parallel$	$\not\exists_{ni}$	$\ni \not\exists_{ni}$	NE
\in	\in	$\in \not\in_{in}$	\in	$\in \not\in_{in}$	$\not\in_{in}$	$=_i \neq_i$	\neq_i	IE
$\not\in_{in}$	$\not\in_{in}$	$\not\in_{in}$	$\in \not\in_{in}$	$\in \not\in_{in}$	$\in \not\in_{in}$	\neq_i	$=_i \neq_i$	IE
EN	EN	EN	EN	EN	EN	EI	EI	$=_e$

\diamond	$=_i$	\neq_i	\in	$\not\in_{in}$	IE	\diamond	$=_e$	EN	EI
$=_i$	$=_i$	\neq_i	\in	$\not\in_{in}$	IE	$=_e$	$=_e$	EN	EI
\neq_i	\neq_i	$=_i \neq_i$	$\in \not\in_{in}$	$\in \not\in_{in}$	IE	NE	NE	$=_n > < \emptyset \parallel$	$\ni \not\exists_{ni}$
\ni	\ni	$\ni \not\exists_{ni}$	$=_n > < \emptyset$	$> \emptyset \parallel$	NE	IE	IE	$\in \not\in_{in}$	$=_i \neq_i$
$\not\exists_{ni}$	$\not\exists_{ni}$	$\ni \not\exists_{ni}$	$< \emptyset \parallel$	$=_n > < \emptyset \parallel$	NE				
EI	EI	EI	EN	EN	$=_e$				

$\mathbb{A}14$ is not associative. For instance,

$$=_e \diamond (1 \diamond 1) = (=_e \diamond 1) = \{EN, =_e, \not\exists\},$$

whereas

$$(=_e \diamond 1) \diamond 1 = \{EN, =_e, \not\exists\} \diamond 1 = \{=_n, >, <, \emptyset, \parallel, =_e EN, NE, \ni, \not\exists\}.$$

How to refine the partition \mathcal{P}^{14} so that it generates an associative algebra, i.e., a relation algebra? The following proposition defines a condition, which a sought-for refined partition scheme must satisfy.

Proposition 3. *Let $(\mathcal{U}, \mathcal{P})$ be a general partition scheme. If the algebra $\mathbb{A}_{\mathcal{P}}$ is associative, then for any base relation $R \in \mathcal{P}$ there exist identity atoms $Id_i, Id_j \in \mathcal{P}$ such that $R \subseteq Fd(Id_i) \times Fd(Id_j)$.*

Proof. From Theorem 3.5 [17] it follows that if $\mathbb{A} \in RA$, then $x; x^\vee \cdot 1', x^\vee; x \cdot 1' \in At(\mathbb{A})$ for all $x \in At(\mathbb{A})$. Applied to $\mathbb{A}_{\mathcal{P}}$ we obtain that $(\forall R \in \mathcal{P}) (\exists Id_i, Id_j \in \mathcal{P}) (R \circ R^{-1}) \cap Id_{\mathcal{U}} = Id_i$ and $(R^{-1} \circ R) \cap Id_{\mathcal{U}} = Id_j$. $(\forall (x, y) \in R) (x, x) \in (R \circ R^{-1}) \cap Id_{\mathcal{U}}$ and $(y, y) \in (R^{-1} \circ R) \cap Id_{\mathcal{U}}$, hence $x \in Fd(Id_i)$ and $y \in Fd(Id_j)$. Therefore, $(x, y) \in Fd(Id_i) \times Fd(Id_j)$, from which follows that $R \subseteq Fd(Id_i) \times Fd(Id_j)$.

The fields of the identity atoms $=_n$, $=_e$ and $=_i$ are:

$$Fd(=_n) = 2^D \setminus \{\emptyset\}, \quad Fd(=_e) = \{\emptyset\}, \quad Fd(=_i) = D.$$

Proposition 3 contains a necessary condition for a general partition scheme to generate a relation algebra. The fact that the algebra generated by \mathcal{P}^{14} is not associative implies that \mathcal{P}^{14} violates this condition. More concretely, this condition fails on the base relations $\not\subseteq$ and $\not\supseteq$. Indeed, $\not\subseteq$ is contained in $D \times 2^D$, but it is contained in neither $D \times (2^D \setminus \{\emptyset\})$ nor $D \times \{\emptyset\}$. The necessary refinement of \mathcal{P}^{14} is the following: $\not\subseteq$ splits into $\not\subseteq_{in}$ and IE , likewise $\not\supseteq$ splits into $\not\supseteq_{ni}$ and EI . We denote the refined partition as \mathcal{P}^{16} . The refined base relations are:

$$\begin{aligned} \not\subseteq_{in}(\alpha, \beta) &\text{ iff } \alpha \text{ is an individual, } \beta \text{ is a nonempty set and } \alpha \not\subseteq \beta \\ IE(\alpha, \beta) &\text{ iff } \alpha \text{ is an individual, } \beta = \emptyset \\ \not\supseteq_{ni}(\alpha, \beta) &\text{ iff } \alpha \text{ is a nonempty set, } \beta \text{ is an individual and } \alpha \not\supseteq \beta \\ EI(\alpha, \beta) &\text{ iff } \beta \text{ is an individual, } \alpha = \emptyset \end{aligned}$$

Weak composition of the algebra $\mathbb{A}16$, which is generated by \mathcal{P}^{16} , is specified in Table 3. It is given by three composition tables. If the composition of two relations is not given by either table, then it is equal to zero.

Proposition 4. *$\mathbb{A}16$ is a relation algebra.*

Proof. Associativity can be checked manually, whereas satisfiability of the remaining axioms of RA follows from Proposition 2.

5 The calculus of alignments revisited

The calculus of alignments defined in [5] assumes that all ontology alignment relations are elements of an algebra of relations, and vice versa. However, this scheme does not work with $\mathbb{A}16$. In $\mathbb{A}16$, not all relations are meaningful enough to be used in alignments. Here we consider the set of base relations independently from the algebra $\mathbb{A}16$.

Let \mathbf{R} be the set of base ontology alignment relations (relation symbols):

$$\mathbf{R} = \{\equiv, \sqsubset, \sqsupset, \not\subseteq, \not\supseteq, \in, \notin, \ni, \not\ni, =, \neq\}. \quad (5.1)$$

\mathbf{R} is so to speak an interface for the algebra $\mathbb{A}16$. Ontology alignment relations are then refined as disjunctions of symbols in \mathbf{R} , denoted as \mathbf{R}^\vee . The relations $\equiv \vee \sqsubset$ and $\equiv \vee \sqsupset$ are abbreviated as \sqsubseteq and \sqsupseteq respectively.

We distinguish between two kinds of atoms in $\mathbb{A}16$: coherent and incoherent.

$$At_{coh}(\mathbb{A}16) = \{=_n, <, >, \not\subseteq, \not\supseteq, \in, \notin, \ni, \not\ni, =_i, \neq_i\} \quad (5.2)$$

$$At_{incoh}(\mathbb{A}16) = \{=_e, EN, NE, EI, IE\} \quad (5.3)$$

Coherent atoms correspond to base ontology alignment relations. A relation $r \in \mathbb{A}16$ is said coherent if all its atoms are coherent. $Coherent(r)$ denotes the set of coherent atoms in r , and $Coherent(\mathbb{A})$, the set of coherent relations in \mathbb{A} .

Table 4: The function ϕ from ontology alignment relations \mathbf{R}^\vee to the algebra $\mathbb{A}16$.

Base OA relation r	Synonyms	$\phi(r)$
\equiv	owl:equivalentClass	$=_n, =_e$
\sqsubset	less general than	$<, EN$
\sqsupset	more general than	$>, NE$
$\overline{\cap}$	partially overlaps with	$\overline{\cap}$
\perp	owl:disjointWith	$\parallel, =_e, EN, NE$
\in	is a, rdf:type	\in
\notin	is not	\notin_{in}, IE
\ni		\ni
$\not\equiv$		$\not\equiv_{ni}, EI$
$=$	owl:sameAs	$=_i$
\neq	owl:differentFrom	\neq_i

We further define a function ϕ from \mathbf{R} to $\mathbb{A}16$ (given in Table 4). This function is naturally extended on \mathbf{R}^\vee , so that $\phi(r \vee s) = \phi(r) \cup \phi(s)$. Coherent relations of $\mathbb{A}16$ are in a one-to-one correspondence with ontology alignment relations. This is given by a function η :

$$\eta : \text{Coherent}(\mathbb{A}16) \rightarrow \mathbf{R}^\vee. \quad (5.4)$$

The operations on alignments with relations from $\mathbb{A}16$ are defined in the same way as in Section 3.2. The difference is that we add a correspondence interpretation level. Let $\mu = (e, e', r)$ be a correspondence, in which $r \in \mathbb{A}16$.

- If $r = \emptyset$, then μ is *inconsistent*.
- If r contains only incoherent atoms ($\text{Coherent}(r) = \emptyset$), then μ is *incoherent*.
- In all other cases r is interpreted as $\eta(\text{Coherent}(r)) \in \mathbf{R}^\vee$.

For instance, assume that we want to compose the correspondences $\mu = (e, e', \perp)$ and $\nu = (e', e'', \sqsubseteq)$. In $\mathbb{A}5$ the relations \perp and \sqsubseteq are considered as shortcuts for the elements $\{\perp\}$ and $\{<, =\}$ respectively. Thus,

$$\begin{aligned} \mu \circ \nu &= (e, e', \{\perp\}) \circ (e', e'', \{<, =\}) \\ &= (e, e'', \{\perp\} \circ \{<, =\}) = (e, e'', \{<, \overline{\cap}, \perp\}). \end{aligned}$$

To compose the correspondences with $\mathbb{A}16$, we first get the elements which correspond to \perp and \sqsubseteq (using the function ϕ) and then compose them:

$$\begin{aligned} &(e, e', \{\parallel, =_e, EN, NE\}) \circ (e', e'', \{=_n, <, =_e, EN\}) \\ &= (e, e', \{\parallel, =_e, EN, NE\} \circ \{=_n, <, =_e, EN\}) \\ &= (e, e'', \{=, >, <, \overline{\cap}, \parallel, =_e, EN, NE\}). \end{aligned}$$

Finally,
$$\begin{aligned} \mu \circ \nu &= (e, e'', \eta(\text{Coherent}(\{=_n, >, <, \overline{\cap}, \parallel, =_e, EN, NE\}))) \\ &= (e, e'', \equiv \vee \sqsubset \vee \sqsupset \vee \overline{\cap} \vee \perp). \end{aligned}$$

This means that we have not deduced anything useful between e and e'' . Indeed, if e' is interpreted as an empty set, then for any interpretation of e and e'' the correspondences μ and ν will hold. But if we add one more correspondence, which guarantees that e' is not empty, e.g., (e''', e', \in) or (e''', e', \sqsubset) , then, by computing the algebraic closure of these three correspondences, we would deduce $(e, e'', \sqsubset \vee \checkmark \vee \perp)$.

6 Discussion and future work

The algebra that we introduced in this paper, $\mathbb{A}16$, covers all qualitative relations between ontology entities from the taxonomy perspective. It is better than $\mathbb{A}5$, which we considered in previous work, in two ways:

- $\mathbb{A}16$ combines class-level and instance-level relations within a single algebra,
- the calculus that it induces on alignments allows to tell between unsatisfiability and incoherence of alignments.

Here are some issues that should be worth of consideration.

Non-taxonomical relations Algebras of relations for ontology alignment are not limited to taxonomical relations. For example, one may build an algebra of relations from biomedical ontologies like SNOMED CT or NDF-RT, using relations like `snomed:hasActiveIngredient` or `ndfirt:mayTreat`. Such an algebra can have, for example, a composition rule `hasActiveIngredient o mayTreat = mayTreat`.

Algebraic formalism The algebra considered in this paper satisfies the axioms of RA. But weaker structures can be used as algebras of ontology alignment relations, e.g., non-associative algebras [17], or Boolean algebras with operators [11].

Non-simple semantics of alignments Here we adopted the simple semantics of alignments. However, algebraic calculi can potentially be adopted to other semantics as well, like the integrated semantics, which is more tolerant to heterogeneity.

Acknowledgements: This research has been partially supported by the joint NSFC-ANR Lindicle project (12-IS01-0002) in cooperation with Tsinghua university.

References

1. J. F. Allen. “Maintaining Knowledge About Temporal Intervals”. In: *Communications of the ACM* 26(11) (1983), pp. 832–843.
2. J. Bao, D. Caragea, and V. Honavar. “On the Semantics of Linking and Importing in Modular Ontologies”. In: *Proc. of ISWC-06*. 2006, pp. 72–86.

3. A. Borgida and L. Serafini. “Distributed description logics: assimilating information from peer sources”. In: *Journal of Data Semantics* 1 (2003), pp. 153–184.
4. I. Düntsch. “Relation algebras and their application in temporal and spatial reasoning”. In: *Artificial Intelligence Review* 23.4 (2005), pp. 315–357.
5. J. Euzenat. “Algebras of Ontology Alignment Relations”. In: *Proc. of ISWC-08*. 2008, pp. 387–402.
6. J. Euzenat and P. Shvaiko. *Ontology Matching*. Second edition. Heidelberg: Springer, 2013.
7. B. C. Grau, P. Patel-Schneider, and B. Motik. *OWL 2 Web Ontology Language Direct Semantics (Second Edition)*. W3C Recommendation. W3C, Dec. 2012. URL: <http://www.w3.org/TR/2012/REC-owl2-direct-semantics-20121211/>.
8. S. Harris, A. Seaborne, and E. Prud’hommeaux. “SPARQL 1.1 Query Language (2013)”. In: *W3C Recommendation* (2013).
9. R. Hirsch. “Expressive Power and Complexity in Algebraic Logic”. In: *J. Log. Comput* 7.3 (1997), pp. 309–351.
10. M. Homola. “Semantic Investigations in Distributed Ontologies”. PhD thesis. Bratislava, Slovakia: Comenius University, 2010.
11. B. Jónsson and A. Tarski. “Boolean algebras with operators. Part II.” In: *American Journal of Mathematics* 74 (1952), pp. 127–162.
12. O. Kutz, C. Lutz, F. Wolter, and M. Zakharyashev. “E-connections of abstract description systems”. In: *Artif. Intell.* 156.1 (2004), pp. 1–73.
13. P. Ladkin and R. D. Maddux. “On binary constraint problems”. In: *Journal of the ACM (JACM)* 41.3 (1994), pp. 435–469.
14. C. Lange, T. Mossakowski, O. Kutz, C. Galinski, M. Grüninger, and D. C. Vale. “The Distributed Ontology Language (DOL): Use Cases, Syntax, and Extensibility”. In: *Proc. of TKE-12*. 2012.
15. G. Ligozat and J. Renz. “What Is a Qualitative Calculus? A General Framework”. In: *Proc. of PRICAI-04*. 2004, pp. 53–64.
16. A. K. Mackworth and E. C. Freuder. “The Complexity of Some Polynomial Network Consistency Algorithms for Constraint Satisfaction Problems”. In: *Artif. Intell.* 25.1 (1985), pp. 65–74.
17. R. D. Maddux. “Some Varieties Containing Relation Algebras”. In: *Trans. Amer. Math. Soc.* 272 (1982), pp. 501–526.
18. A. Tarski. “On the Calculus of Relations”. In: *J. Symb. Log.* 6.3 (1941), pp. 73–89.
19. A. Zimmermann. “Integrated Distributed Description Logics”. In: *Proc. of DL-07*. 2007, pp. 507–514.
20. A. Zimmermann. “Logical Formalisms for Agreement Technologies”. In: *Agreement Technologies*. Ed. by S. Ossowski. Vol. 8. Law, Governance and Technology Series. Springer Netherlands, 2013, pp. 69–82.
21. A. Zimmermann and J. Euzenat. “Three Semantics for Distributed Systems and Their Relations with Alignment Composition”. In: *Proc. of ISWC-06*. 2006, pp. 16–29.