



The category of networks of ontologies

Jérôme Euzenat

**RESEARCH
REPORT**

N° 8652

December 2014

Project-Team Exmo



The category of networks of ontologies

Jérôme Euzenat

Project-Team Exmo

Research Report n° 8652 — December 2014 — 19 pages

Abstract: The semantic web has led to the deployment of ontologies on the web connected through various relations and, in particular, alignments of their vocabularies. There exists several semantics for alignments which make difficult interoperation between different interpretation of networks of ontologies. Here we present an abstraction of these semantics which allows for defining the notions of closure and consistency for networks of ontologies independently from the precise semantics. We also show that networks of ontologies with specific notions of morphisms define categories of networks of ontologies.

Key-words: Network of ontologies, Ontology alignment, Alignment semantics, Inconsistency, Distributed system semantics, Category, Pullback

**RESEARCH CENTRE
GRENOBLE – RHÔNE-ALPES**

Inovallée
655 avenue de l'Europe Montbonnot
38334 Saint Ismier Cedex

La catégorie des réseaux d'ontologies

Résumé : Le web sémantique a suscité le déploiement d'ontologies sur le web liées par diverses relations, et, en particulier, l'alignement de leur vocabulaire. Il existe différentes sémantiques pour les alignements qui rendent difficile l'interopération entre différentes interprétations des réseaux d'ontologies. Ce rapport présente une abstraction de ces sémantiques qui permet de définir les notions de clôture et de consistance de réseaux d'ontologies indépendamment de leur sémantique précise. On montre aussi que les réseaux d'ontologies dotés d'homomorphismes spécifiques définissent des catégories de réseaux d'ontologies.

Mots-clés : Réseau d'ontologies, Alignement d'ontologies, Sémantique des alignements, Inconsistance, Sémantique des systèmes distribués, Catégorie, Produit fibré

1 Motivation

The semantic web relies on knowledge deployed and connected over the web. This knowledge is based on ontologies expressed in languages such as RDF, RDF Schema and OWL. Because of the multiplicity of ontologies, they may be connected through alignments expressing correspondences between their concepts. This allows for translating assertions across ontologies or merging them. This can be seen as a network of ontologies related by alignments.

The goal of this report is to provide a formal account of such networks of ontologies. It aims at abstracting properties of networks so that work relying on such properties can apply independently of their interpretation. In particular, it can contribute to define algebras or revision operators of networks of ontologies.

We first precisely define what alignments and networks of ontologies are through their syntax (§2) before addressing their semantics. There is no “standard” semantics for networks of ontologies, so we provide here an abstract view that aims at covering those which have been proposed so far (§3). Based on this framework, we define the notions of closure from this semantics (§4). Finally, we show that networks of ontologies form a category and we exhibit some properties of these categories (§5). Relation to other work is discussed (§6) before concluding.

In this report, ontologies are considered as logical theories and may contain ground assertions. The languages used in the semantic web such as RDF or OWL are indeed logics [12, 2]. The semantics of ontologies are only considered in this paper through their sets of models ($\mathcal{M}(o)$) and consequence relation (\models). Such a relation satisfies three properties (o, o' are ontologies, i.e., sets of assertions, δ and γ are assertions):

extensivity $\{\delta\} \models \delta$

monotony if $o \models \delta$ then $o \cup o' \models \delta$

idempotency if $o \models \delta$ and $o \cup \{\delta\} \models \gamma$ then $o \models \gamma$

We assume a consequence closure function $Cn^\omega(o) = \{\delta \mid o \models \delta\}$.

2 Alignments and networks of ontologies

Alignments express the correspondences between entities of different ontologies [8]. Given an ontology o in a language L , we use an *entity language* ($Q_L(o)$) for characterising those entities that will be put in correspondence. The entity language can be simply made of all the terms or formulas of the ontology language based on the ontology vocabulary. It can restrict them to the named terms or, on the contrary, extend them to all the queries that may be expressed on this vocabulary. Alignments express relations between such entities through a finite set Θ of relations which are independent from ontology relations.

Definition 1 (Alignment, correspondence). *Given two ontologies o and o' with associated entity languages Q_L and $Q_{L'}$ and a set of alignment relations Θ , a correspondence is a triple: $\langle e, e', r \rangle \in Q_L(o) \times Q_{L'}(o') \times \Theta$ expressing that the relation r holds between entity e and e' . An alignment is a set of correspondences between two ontologies.*

For the sake of examples, we will consider that ontologies are description logic theories (T-box and A-box) and their entities are classes and individuals identified by URIs. Classes are denoted by lower case letters, sometimes subscripted by an integer referring to their ontology, and individuals are denoted by i . In the examples, we will only consider $=, \leq, \geq$ and \perp for relations of Θ . They will be interpreted as relations expressing equivalence, subsumption and disjointness between classes. However, results are not restricted to these relations.

Example 1 (Alignment). *The alignment $A_{1,3}$ of Figure 1 (p.4), is described by:*

$$\{\langle e_1, f_3, \geq \rangle, \langle b_1, e_3, \geq \rangle\}$$

also described as:

$$A_{1,3} = \left\{ \begin{array}{l} e_1 \geq f_3, \\ b_1 \geq e_3 \end{array} \right\}$$

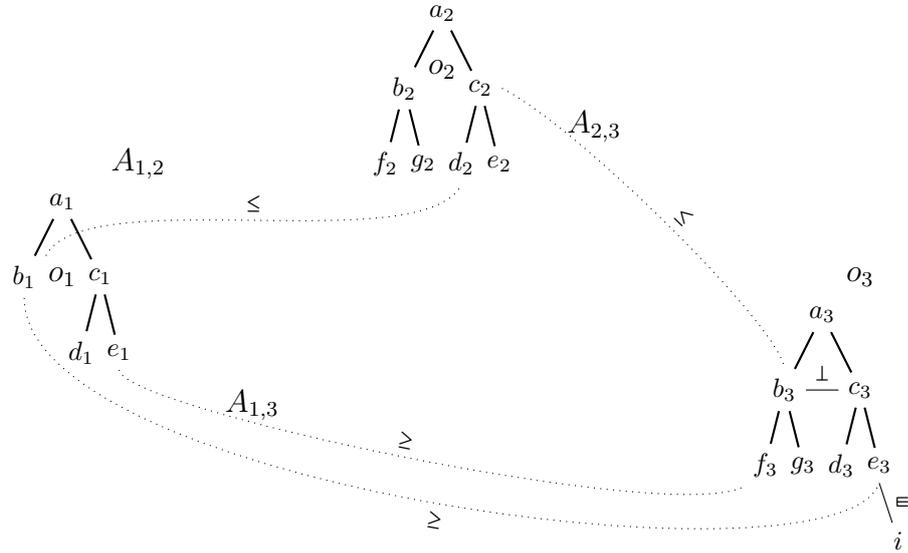


Figure 1: A network of ontologies made of three ontologies (o_1 , o_2 , and o_3) and three alignments ($A_{1,2}$, $A_{1,3}$, and $A_{2,3}$).

The above definition can be generalised to an arbitrary number of alignments and ontologies captured in the concept of a network of ontologies (or distributed system [11, 9]), i.e., sets of ontologies and alignments.

Definition 2 (Network of ontologies). *A network of ontologies $\langle \Omega, \Lambda \rangle$ is made of a finite set Ω of ontologies and a set Λ of alignments between these ontologies. We denote by $\Lambda(o, o')$ the set of alignments in Λ between o and o' .*

Example 2 (Network of ontologies). *Figure 1 presents three ontologies (in all examples, $c \sqsubseteq c'$ denotes subsumption between concepts c and c' , $c \perp c'$ denotes disjointness between concepts c and c' , and $i \sqsubseteq c$*

denotes membership of individual i to concept c):

$$\begin{aligned} o_1 &= \left\{ \begin{array}{l} b_1 \sqsubseteq a_1, c_1 \sqsubseteq a_1 \\ d_1 \sqsubseteq c_1, e_1 \sqsubseteq c_1 \end{array} \right\} \\ o_2 &= \left\{ \begin{array}{l} b_2 \sqsubseteq a_2, c_2 \sqsubseteq a_2, g_2 \sqsubseteq b_2 \\ d_2 \sqsubseteq c_2, e_2 \sqsubseteq c_2, f_2 \sqsubseteq b_2 \end{array} \right\} \\ o_3 &= \left\{ \begin{array}{l} b_3 \sqsubseteq a_3, c_3 \sqsubseteq a_3, g_3 \sqsubseteq b_3 \\ d_3 \sqsubseteq c_3, e_3 \sqsubseteq c_3, f_3 \sqsubseteq b_3 \\ i \sqsubseteq e_3, b_3 \perp c_3 \end{array} \right\} \end{aligned}$$

together with three alignments $A_{1,2}$, $A_{2,3}$, and $A_{3,1}$. These alignments can be described as follows:

$$\begin{aligned} A_{1,2} &= \{ b_1 \leq d_2 \} \\ A_{2,3} &= \{ c_2 \leq b_3 \} \\ A_{1,3} &= \{ e_1 \geq f_3, b_1 \geq e_3 \} \end{aligned}$$

Hereafter, we consider normalised networks of ontologies, i.e., networks with exactly one alignment between each pair of ontologies.

Definition 3 (Normalised network of ontologies). *A network of ontologies $\langle \Omega, \Lambda \rangle$ is said normalised if and only if for any two ontologies o and o' , $|\Lambda(o, o')| = 1$.*

In a normalised network of ontologies, we denote by $\lambda(o, o')$ the unique alignment between o and o' . Any network of ontologies may easily be normalised by:

- if $|\Lambda(o, o')| = 0$, adding an empty alignment between o and o' ,
- if $|\Lambda(o, o')| > 1$, replacing $\Lambda(o, o')$ by a unique alignment containing all the correspondences of the alignments of $\Lambda(o, o')$,

We call this standard normalisation:

Definition 4 (Standard normalisation). *Given a network ontology $\langle \Omega, \Lambda \rangle$, its standard normalisation $\langle \Omega, \bar{\Lambda} \rangle$ is defined by $\forall o, o' \in \Omega$,*

$$\bar{\Lambda}(o, o') = \begin{cases} \{\{\}\}, & \text{if } \Lambda(o, o') = \emptyset, \\ \{\cup_{A \in \Lambda(o, o')} A\}, & \text{otherwise} \end{cases}$$

The unique element of $\bar{\Lambda}(o, o')$ is denoted by $\bar{\lambda}(o, o')$.

There are other ways to normalise such networks, but this simple one is sufficient for obtaining equivalent normalised networks (see Property 4).

Comparing networks of ontologies is not in general simple. For that purpose, we introduce the notion of morphism between two networks of ontologies.

Definition 5 (Syntactic morphism between networks of ontologies). *Given two networks of ontologies, $\langle \Omega, \Lambda \rangle$ and $\langle \Omega', \Lambda' \rangle$, a syntactic morphism between $\langle \Omega, \Lambda \rangle$ and $\langle \Omega', \Lambda' \rangle$, is $\langle h, k \rangle$, a pair of morphisms: $h : \Omega \rightarrow \Omega'$ and $k : \Lambda \rightarrow \Lambda'$ such that $\forall o \in \Omega, \exists h(o) \in \Omega'$ and $o \sqsubseteq h(o)$ and $\forall A \in \Lambda(o, o'), \exists k(A) \in \Lambda'(h(o), h(o'))$ and $A \sqsubseteq k(A)$.*

Such a morphism exists when one network can be projected into another one and keep its structure and syntactic content. This means that any ontology (respectively any alignment) of the former has a counterpart in the latter one which contains at least all of its axioms (respectively correspondences) and that the graph structure of the former network is preserved in the latter. It is possible that several ontologies or alignments have the same counterpart as long as these conditions are met.

Morphisms can be used for defining syntactic subsumption between networks of ontologies.

Definition 6 (Syntactic subsumption between networks of ontologies). *Given two networks of ontologies, $\langle \Omega, \Lambda \rangle$ and $\langle \Omega', \Lambda' \rangle$, $\langle \Omega, \Lambda \rangle$ is syntactically subsumed by $\langle \Omega', \Lambda' \rangle$, denoted by $\langle \Omega, \Lambda \rangle \sqsubseteq \langle \Omega', \Lambda' \rangle$, iff there exist a syntactic morphism between $\langle \Omega, \Lambda \rangle$ and $\langle \Omega', \Lambda' \rangle$.*

We note:

$$\begin{aligned} \langle \Omega, \Lambda \rangle \sqsubseteq \langle \Omega', \Lambda' \rangle &\text{ iff } \langle \Omega, \Lambda \rangle \sqsubseteq \langle \Omega', \Lambda' \rangle \text{ and } \langle \Omega', \Lambda' \rangle \sqsubseteq \langle \Omega, \Lambda \rangle \\ \langle \Omega, \Lambda \rangle \subset \langle \Omega', \Lambda' \rangle &\text{ iff } \langle \Omega, \Lambda \rangle \sqsubseteq \langle \Omega', \Lambda' \rangle \text{ and } \langle \Omega', \Lambda' \rangle \not\sqsubseteq \langle \Omega, \Lambda \rangle \end{aligned}$$

This definition is purely syntactic because semantically equivalent networks may not be syntactically subsumed (it suffices to use one ontology whose axioms are equivalent but different).

The empty network of ontologies $\langle \emptyset, \emptyset \rangle$ (containing no ontology and no alignment) is subsumed by any other network of ontologies.

It is possible to simplify the above definition in case of normalised networks of ontologies.

Property 1. *Given two normalised networks of ontologies $\langle \Omega, \Lambda \rangle$ and $\langle \Omega', \Lambda' \rangle$, $\langle \Omega, \Lambda \rangle \sqsubseteq \langle \Omega', \Lambda' \rangle$ iff $\exists h : \Omega \rightarrow \Omega'$ a morphism such that $\forall o \in \Omega, \exists h(o) \in \Omega'$ and $o \subseteq h(o)$ and $\forall o, o' \in \Omega, \lambda(o, o') \subseteq \lambda'(h(o), h(o'))$.*

Proof. \Rightarrow) In normalised networks, there always exists a single alignment between each pair of ontologies. Hence, if k is such that $\forall A \in \Lambda(o, o'), k(A) \in \Lambda'(h(o), h(o'))$, this means that $k(A) = k(\lambda(o, o')) = \lambda'(h(o), h(o'))$ and since $A \subseteq k(A)$, then $\lambda(o, o') \subseteq \lambda'(h(o), h(o'))$.

\Leftarrow) In $\langle \Omega, \Lambda \rangle, \Lambda(o, o') = \{\lambda(o, o')\}$ and the same holds for $\langle \Omega', \Lambda' \rangle$, if $k(\lambda(o, o')) = \lambda'(h(o), h(o'))$ it satisfies the constraint that $k(A) \in \Lambda'(h(o), h(o'))$ and $A \subseteq k(A)$. \square

Moreover, networks are subsumed by their standard normalisation.

Property 2. *Let $\langle \Omega, \Lambda \rangle$ a network of ontologies and $\langle \Omega, \bar{\Lambda} \rangle$ its standard normalisation,*

$$\langle \Omega, \Lambda \rangle \sqsubseteq \langle \Omega, \bar{\Lambda} \rangle$$

Proof. Consider, the pair of morphisms $\langle h, k \rangle$ such that $h(o) = o$ and $\forall o, o' \in \Omega, \forall A \in \Lambda(o, o'), k(A) = \bar{\lambda}(o, o')$, then $A \subseteq k(A)$ because $A \subseteq \bigcup_{A' \in \Lambda(o, o')} A'$. \square

From subsumption, conjunction (meet) can be introduced in a standard way:

Definition 7 (Syntactic conjunction of networks of ontologies). *Given a finite family of networks of ontologies, $\{\langle \Omega_i, \Lambda_i \rangle\}_{i \in I}, \prod_{i \in I} \langle \Omega_i, \Lambda_i \rangle = \langle \Omega', \Lambda' \rangle$ such that $\forall i \in I, \langle \Omega', \Lambda' \rangle \sqsubseteq \langle \Omega_i, \Lambda_i \rangle$ and $\forall \langle \Omega'', \Lambda'' \rangle; \langle \Omega'', \Lambda'' \rangle \sqsubseteq \langle \Omega_i, \Lambda_i \rangle, \langle \Omega'', \Lambda'' \rangle \sqsubseteq \langle \Omega', \Lambda' \rangle$.*

Such a conjunction always exists because the empty network of ontologies is subsumed by all network of ontologies. However, it is likely not unique because of the choice of homomorphisms.

We can also define simple operations on networks of ontologies.

Definition 8 (Substitution in networks of ontologies). *Given a network of ontologies $\langle \Omega, \Lambda \rangle$, given $o \in \Omega$ and o' another ontology, $\langle \Omega, \Lambda \rangle[o/o'] = \langle \Omega \setminus \{o\} \cup \{o'\}, \Lambda \setminus \bigcup_{o'' \in \Omega} (\Lambda(o, o'') \cup \Lambda(o'', o)) \rangle$.*

Given a network of ontologies $\langle \Omega, \Lambda \rangle$, given $A \in \Lambda(o, o')$ and A' another alignment between o and o' , $\langle \Omega, \Lambda \rangle[A/A'] = \langle \Omega, \Lambda \setminus \{A\} \cup \{A'\} \rangle$.

Property 3. Given a network of ontologies $\langle \Omega, \Lambda \rangle$ with $o \in \Omega$ and $A \in \Lambda$, if $o' \subseteq o$, then $\langle \Omega, \Lambda \rangle[o/o'] \subseteq \langle \Omega, \Lambda \rangle$ and if $A' \subseteq A$, then $\langle \Omega, \Lambda \rangle[A/A'] \subseteq \langle \Omega, \Lambda \rangle$

Proof. Simply, there exists a pair of morphisms $\langle h, k \rangle$ which identifies each ontology to itself but o' which is identified to o , and each alignment to itself, but A' which is identified to A in the second case. In the case of $\langle \Omega, \Lambda \rangle[o/o']$, the set of alignments is strictly included in Λ . It is thus clear that $h(o'') \subseteq o''$ and $k(A'') \subseteq A''$, including for o' and A' (by hypothesis). The structure is preserved because ontologies and alignments are the same, except in the second case in which the substituted alignment preserves the structure. \square

3 Semantics of networks of ontologies

The semantics of aligned ontologies, or networks of ontologies, must remain compatible with the classical semantics of ontologies: connecting ontologies to other ontologies should not radically change the manner to interpret them.

When ontologies are independent, i.e., not related with alignments, it is natural that their semantics is the classical semantics for these ontologies, i.e., a set of models $\mathcal{M}(o)$. A model is a map m from the entities of the ontologies to a particular domain D . Such models have to apply to all the elements of the entity language $Q_L(o)$ (when it is larger than the ontology language, this is usually defined inductively on the structure of its elements).

Different semantics provide alternative ways to record the constraints imposed by alignments: through relations between domains of interpretation [11, 4], through equalising functions [16, 14], by imposing equal [13] or disjoint [5] domains. These models have been compared elsewhere [16]; we provide an informal unified view of these semantics.

For that purpose, each correspondence is interpreted with respect to three features: a model for each ontology and a semantic structure, denoted by Δ [15]. This loosely defined semantic structure has two purposes:

- providing an interpretation to the correspondence relations in Θ (which are independent from the ontology semantics);
- memorising the constraints imposed on models by the alignments.

In this work, it can simply be considered that Δ is used, in each semantics, to define the satisfaction of a correspondence μ by two ontology models o and o' (which is denoted by $m_o, m_{o'} \models_{\Delta} \mu$).

Such a simple notion of satisfaction, imposing no constraints on models, is provided by Example 3.

Example 3 (Interpretation of correspondences). *In the language used as example, c and c' stand for classes and i and i' for individuals. If m_o and $m_{o'}$ are respective models of o and o' :*

$$\begin{aligned} m_o, m_{o'} \models_{\Delta} \langle c, c', = \rangle & \text{ iff } m_o(c) = m_{o'}(c') \\ m_o, m_{o'} \models_{\Delta} \langle c, c', \leq \rangle & \text{ iff } m_o(c) \subseteq m_{o'}(c') \\ m_o, m_{o'} \models_{\Delta} \langle c, c', \geq \rangle & \text{ iff } m_o(c) \supseteq m_{o'}(c') \\ m_o, m_{o'} \models_{\Delta} \langle i, c', \in \rangle & \text{ iff } m_o(i) \in m_{o'}(c') \\ m_o, m_{o'} \models_{\Delta} \langle c, i', \ni \rangle & \text{ iff } m_{o'}(i') \in m_o(c) \\ m_o, m_{o'} \models_{\Delta} \langle c, c', \perp \rangle & \text{ iff } m_o(c) \cap m_{o'}(c') = \emptyset \end{aligned}$$

Example 4 (Interpretation of correspondences in first-order logic). *The semantics can be given with respect to first-order logic theories. In such a case, correspondences relate predicates p and p' of the*

same arity between ontologies o and o' . If m_o and $m_{o'}$ are their respective first-order models:

$$\begin{aligned} m_o, m_{o'} \models_{\Delta} \langle p, p', = \rangle &\text{ iff } m_o(p) = m_{o'}(p') \\ m_o, m_{o'} \models_{\Delta} \langle p, p', \leq \rangle &\text{ iff } m_o(p) \subseteq m_{o'}(p') \\ m_o, m_{o'} \models_{\Delta} \langle p, p', \geq \rangle &\text{ iff } m_o(p) \supseteq m_{o'}(p') \\ m_o, m_{o'} \models_{\Delta} \langle p, p', \perp \rangle &\text{ iff } m_o(p) \cap m_{o'}(p') = \emptyset \end{aligned}$$

This semantics selects first-order theory interpretations by setting constraints on predicate interpretations.

Example 5 (Interpretation of correspondences in the equalising semantics). *An alternative interpretation in the equalising semantics [14] relies on a family of functions indexed by each ontology γ from the domains of interpretations of each ontologies to a universal domain U . Hence, $\Delta = \langle \gamma, U \rangle$. Then, in the case of Example 3, if m_o and $m_{o'}$ are respective models of o and o' :*

$$\begin{aligned} m_o, m_{o'} \models_{\Delta} \langle c, c', = \rangle &\text{ iff } \gamma_o \circ m_o(c) = \gamma_{o'} \circ m_{o'}(c') \\ m_o, m_{o'} \models_{\Delta} \langle c, c', \leq \rangle &\text{ iff } \gamma_o \circ m_o(c) \subseteq \gamma_{o'} \circ m_{o'}(c') \text{ or } \gamma_o \circ m_o(c) \in \gamma_{o'} \circ m_{o'}(c') \\ m_o, m_{o'} \models_{\Delta} \langle c, c', \geq \rangle &\text{ iff } \gamma_o \circ m_o(c) \supseteq \gamma_{o'} \circ m_{o'}(c') \text{ or } \gamma_{o'} \circ m_{o'}(c') \in \gamma_o \circ m_o(c) \\ m_o, m_{o'} \models_{\Delta} \langle i, c', \in \rangle &\text{ iff } \gamma_o \circ m_o(i) \in \gamma_{o'} \circ m_{o'}(c') \text{ or } \gamma_o \circ m_o(i) \subseteq \gamma_{o'} \circ m_{o'}(c') \end{aligned}$$

This semantics allows for changing the interpretation of an individual as a set and vice-versa.

Hence, the semantics of two aligned ontologies may be given as a set of models which are pairs of compatible models.

Definition 9 (Models of alignments). *Given two ontologies o and o' and an alignment A between these ontologies, a model of this alignment is a triple $\langle m_o, m_{o'}, \Delta \rangle$ with $m_o \in \mathcal{M}(o)$, $m_{o'} \in \mathcal{M}(o')$, and Δ a semantic structure, such that $\forall \mu \in A$, $m_o, m_{o'} \models_{\Delta} \mu$ (denoted by $m_o, m_{o'} \models_{\Delta} A$).*

We note $A \models \mu$ iff $\forall \langle m_o, m_{o'}, \Delta \rangle$ such that $m_o, m_{o'} \models_{\Delta} A$, $m_o, m_{o'} \models_{\Delta} \mu$. Similarly as for ontologies, the semantics of alignments can be given by the relation \models such that (A and A' are alignments and μ and ν are correspondences all between the same pair of ontologies):

extensivity $\{\mu\} \models \mu$
monotony if $A \models \mu$ then $A \cup A' \models \mu$
idempotency if $A \models \mu$ and $A \cup \{\mu\} \models \nu$ then $A \models \nu$

Similarly, we assume a consequence closure function $Cn^{\alpha}(A) = \{\mu \mid A \models \mu\}$.

Models of networks of ontologies extend models of alignments. They select compatible models for each ontology in the network [10]. Compatibility consists of satisfying all the alignments of the network.

Definition 10 (Models of networks of ontologies). *Given a network of ontologies $\langle \Omega, \Lambda \rangle$, a model of $\langle \Omega, \Lambda \rangle$ is a pair $\langle m, \Delta \rangle$ with m a family of models indexed by Ω with $\forall o \in \Omega$, $m_o \in \mathcal{M}(o)$ such that for each alignment $A \in \Lambda(o, o')$, $m_o, m_{o'} \models_{\Delta} A$. The set of models of $\langle \Omega, \Lambda \rangle$ is denoted by $\mathcal{M}(\langle \Omega, \Lambda \rangle)$.*

In that respect, alignments act as model filters for the ontologies. They select the ontology interpretations which are coherent with the alignments. This allows for transferring information from one ontology to another since reducing the set of models entails more consequences in each aligned ontology.

Example 6 (Model of a network of ontologies). *Hence, a model for the network of ontologies of Figure 1 with Δ as defined in Example 3, is $\langle \{m_1, m_2, m_3\}, \Delta \rangle$ built on any models m_1 , m_2 and m_3 of ontology o_1 , o_2 and o_3 such that $m_3(e_3) \subseteq m_1(b_1) \subseteq m_2(d_2)$, $m_2(c_2) \subseteq m_3(b_3)$ and $m_3(f_3) \subseteq m_1(e_1)$.*

Standard normalisation provides a semantically equivalent network according to this semantics.

Property 4 (Soundness of standard normalisation). *Let $\langle \Omega, \Lambda \rangle$ be a network of ontologies and $\langle \Omega, \bar{\Lambda} \rangle$ its standard normalisation,*

$$\mathcal{M}(\langle \Omega, \Lambda \rangle) = \mathcal{M}(\langle \Omega, \bar{\Lambda} \rangle)$$

Proof. $\mathcal{M}(\langle \Omega, \Lambda \rangle) = \mathcal{M}(\langle \Omega, \bar{\Lambda} \rangle)$ because each model is made of a set of models of the same ontologies Ω and a semantic structure Δ which has to satisfy the same set of correspondences.

More precisely, if $\langle m, \Delta \rangle \in \mathcal{M}(\langle \Omega, \Lambda \rangle)$, then $\forall o, o' \in \Omega, \forall A \in \Lambda(o, o'), \forall \mu \in A, m_o, m_{o'} \models_{\Delta} \mu$ which means that $\forall o, o' \in \Omega, \forall \mu \in \bar{\lambda}(o, o'), m_o, m_{o'} \models_{\Delta} \mu$ because either $\Lambda(o, o') = \emptyset$ and then $\bar{\lambda}(o, o')$ does not contain any μ or $\forall \mu \in \bar{\lambda}(o, o'), \exists A \in \Lambda(o, o')$ such that $\mu \in A$ and thus $m_o, m_{o'} \models_{\Delta} \mu$. Hence, $\langle m, \Delta \rangle \in \mathcal{M}(\langle \Omega, \bar{\Lambda} \rangle)$

In the reverse direction, if $\langle m, \Delta \rangle \in \mathcal{M}(\langle \Omega, \bar{\Lambda} \rangle)$, then $\forall o, o' \in \Omega, \forall \mu \in \bar{\lambda}(o, o'), m_o, m_{o'} \models_{\Delta} \mu$, but $\forall o, o' \in \Omega, \forall A \in \Lambda(o, o'), \forall \mu \in A, \mu \in \bar{\lambda}(o, o')$ because $\bar{\lambda}(o, o') = \bigcup_{A \in \Lambda(o, o')} A$, thus $m_o, m_{o'} \models_{\Delta} \mu$. Hence, $\langle m, \Delta \rangle \in \mathcal{M}(\langle \Omega, \Lambda \rangle)$. \square

This justifies the position to only consider normalised networks of ontologies.

We consider an order relation \ll between semantic structures denoting the reinforcement of constraints (more complete relations between domains of interpretations or more disjunctions between domains). The stronger the semantic structure, the less models it accepts.

Definition 11 (Constraint reinforcement). *Given two semantic structures Δ and Δ' ,*

$$\Delta \ll \Delta' \text{ iff } \forall m, m', \forall \mu, m, m' \models_{\Delta'} \mu \Rightarrow m, m' \models_{\Delta} \mu$$

The models of a network and that of its standard normalisation can be compared because they are indexed by the same set of ontologies. However, it is not generally possible to directly compare two sets of models of two networks because the set of ontologies that index them is not the same. Therefore, this is again done up to homomorphism.

Definition 12 (Model inclusion). *Given two networks of ontologies $\langle \Omega, \Lambda \rangle$ and $\langle \Omega', \Lambda' \rangle$, the set of models of the latter is said included in that of the former, and denoted by $\mathcal{M}(\langle \Omega', \Lambda' \rangle) \trianglelefteq \mathcal{M}(\langle \Omega, \Lambda \rangle)$, if and only if there exists a map $h : \Omega \rightarrow \Omega'$ such that $\forall \langle m', \Delta' \rangle \in \mathcal{M}(\langle \Omega', \Lambda' \rangle), \exists \langle m, \Delta \rangle \in \mathcal{M}(\langle \Omega, \Lambda \rangle); \forall o \in \Omega, m_o = m'_{h(o)}$ and $\Delta \ll \Delta'$.*

Property 5 shows that syntactically subsumed networks of ontologies have more models.

Property 5 (Model antitony). *Let $\langle \Omega, \Lambda \rangle$ and $\langle \Omega', \Lambda' \rangle$ be two networks of ontologies,*

$$\langle \Omega, \Lambda \rangle \sqsubseteq \langle \Omega', \Lambda' \rangle \Rightarrow \mathcal{M}(\langle \Omega', \Lambda' \rangle) \trianglelefteq \mathcal{M}(\langle \Omega, \Lambda \rangle)$$

Proof. $\langle \Omega, \Lambda \rangle \sqsubseteq \langle \Omega', \Lambda' \rangle$ means that $\exists \langle h, k \rangle; \forall o \in \Omega, o \subseteq h(o)$ and $\forall o, o' \in \Omega, \forall A \in \Lambda(o, o'), A \subseteq k(A) \wedge k(A) \in \Lambda'(h(o), h(o'))$. Hence, $\exists \langle h, k \rangle; \forall m \in \mathcal{M}(h(o)), m \in \mathcal{M}(o)$ and $\forall \langle m, m' \rangle \in \mathcal{M}(h(o)) \times \mathcal{M}(h(o')), \forall \Delta', m, m' \models_{\Delta'} k(A) \Rightarrow m, m' \models_{\Delta} A$ (because $A \subseteq k(A)$). In addition, because A imposes less constraints on the models than $k(A)$, models may be defined with $\Delta \ll \Delta'$, but in such a case, $m, m' \models_{\Delta} A$. Thus, there exists a map $h : \Omega \rightarrow \Omega'$ such that $\forall \langle m', \Delta' \rangle \in \mathcal{M}(\langle \Omega', \Lambda' \rangle), \exists \langle m, \Delta \rangle \in \mathcal{M}(\langle \Omega, \Lambda \rangle); \forall o \in \Omega, m_o = m'_{h(o)}$ and $\Delta \ll \Delta'$. In consequence, $\mathcal{M}(\langle \Omega', \Lambda' \rangle) \trianglelefteq \mathcal{M}(\langle \Omega, \Lambda \rangle)$. \square

Property 6 (Downward consistency preservation). *Let $\langle \Omega, \Lambda \rangle$ and $\langle \Omega', \Lambda' \rangle$ be two networks of ontologies, If $\langle \Omega, \Lambda \rangle \sqsubseteq \langle \Omega', \Lambda' \rangle$ and $\langle \Omega', \Lambda' \rangle$ is consistent, then $\langle \Omega, \Lambda \rangle$ is consistent.*

Proof. Straightforward from Property 5, since if $\langle \Omega', \Lambda' \rangle$ is consistent, it has a model, and so does $\langle \Omega, \Lambda \rangle$. \square

It is expected that all constraints applying to the semantics are preserved by the syntactic morphisms. However, the converse is not guarantee: if a network preserves the constraints of another then there does not necessarily imply that there exist a syntactic morphism.

4 Consistency, consequence and closure

A network of ontologies is consistent if it has a model. By extension, an ontology or an alignment is consistent within a network of ontologies if the network of ontologies is consistent. Hence even if an ontology is consistent when taken in isolation, it may be inconsistent when inserted in a network of ontologies. Moreover, if one of the ontologies in the network is inconsistent, then the network as a whole is inconsistent.

Example 7 (Inconsistency). *A model of the network of ontologies presented in Example 2 according to the interpretation of correspondence of Example 3, retains families of models $\{m_1, m_2, m_3\}$ satisfying all alignments, i.e., in particular, satisfying:*

$$\begin{aligned} m_3(b_3) &\supseteq m_2(c_2) \\ m_2(c_2) &\supseteq m_1(b_1) \\ m_1(b_1) &\supseteq m_3(e_3) \end{aligned}$$

But, all models m_3 of o_3 must satisfy $m_3(b_3) \cap m_3(c_3) = \emptyset$, $m_3(i) \in m_3(e_3)$, and $m_3(i) \in m_3(c_3)$. Moreover, all models m_2 of o_2 must satisfy $m_2(d_2) \subseteq m_2(c_2)$. Hence, $m_3(b_3) \supseteq m_1(b_1)$, $m_3(b_3) \supseteq m_3(e_3)$ and then $m_3(i) \in m_3(b_3)$, which is contradictory with previous assertions. Thus, there cannot exist such a family of models and there is no model for this network of ontologies.

If the interpretation of Example 5 were retained, there may be models in which $\gamma \circ m_3(i) = \emptyset$.

In this example, taking any of the ontologies with only the alignments which involve them, e.g., $\langle \Omega, \{A_{1,3}, A_{2,3}\} \rangle$, is a consistent network of ontologies. The following examples will also consider the network of ontologies $\langle \Omega', \Lambda \rangle = \langle \{o_1, o_2, o'_3\}, \{A_{1,2}, A_{1,3}, A_{2,3}\} \rangle$ such that o'_3 is $o_3 \setminus \{i \equiv e_3\}$.

So far, we have not defined what it means for a formula to be the consequence of a network. There are two notions of consequences called ω -consequence and α -consequence.

α -consequences are correspondences which are consequences of networks of ontologies [6].

Definition 13 (α -Consequence of networks of ontologies). *Given a finite set of ontologies Ω and a finite set of alignments Λ between pairs of ontologies in Ω , a correspondence μ between two ontologies o and o' in Ω is an α -consequence of $\langle \Omega, \Lambda \rangle$ (denoted by $\models_{\Omega, \Lambda} \mu$ or $\langle \Omega, \Lambda \rangle \models \mu$) if and only if for all models $\langle m, \Delta \rangle$ of $\langle \Omega, \Lambda \rangle$, $m_o, m_{o'} \models \Delta \mu$.*

The set of α -consequences between o and o' is denoted by $Cn_{\Omega, \Lambda}^\alpha(o, o')$. For homogeneity of notation, we will use $Cn_{\Omega, \Lambda}^\alpha(A)$ for denoting $Cn_{\Omega, \Lambda}^\alpha(o, o')$ when $A \in \Lambda(o, o')$. The α -closure of a network of ontologies is its set of α -consequences: the correspondences which are satisfied in all models of the network of ontologies.

From the alignment semantics, it is possible to decide if an alignment is a consequence of another or if the alignment makes the set of ontologies and alignments inconsistent.

Example 8 (α -consequences). *The closure of $A_{1,3}$ in the network of ontology $\langle \Omega', \Lambda \rangle$ of Example 7 is:*

$$Cn_{\Omega', \Lambda}^\alpha(o_1, o'_3) = \left\{ \begin{array}{l} e_1 \geq f_3, b_1 \geq e_3 \\ c_1 \geq f_3, a_1 \geq f_3 \\ a_1 \geq e_3, b_1 \leq b_3 \\ b_1 \leq a_3, b_1 \perp c_3 \\ b_1 \perp d_3, b_1 \perp e_3 \end{array} \right\}$$

but if the network is reduced to the two involved ontologies (o_1 and o'_3) only, the closure would be:

$$Cn_{\{o_1, o'_3\}, \{A_{1,3}\}}^\alpha(o_1, o'_3) = \left\{ \begin{array}{l} e_1 \geq f_3, b_1 \geq e_3 \\ c_1 \geq f_3, a_1 \geq f_3 \\ a_1 \geq e_3 \end{array} \right\}$$

It is thus clear that connecting more ontologies provides more information.

According to these definitions, $Cn^\alpha(A) = Cn_{\{\{o,o'\},\{A\}\}}^\alpha(A)$ when $A \in \Lambda(o, o')$. α -consequences of an alignment are defined as the α -consequences of the network made of this alignment and the two ontologies it connects. The α -consequences of a particular alignment are usually larger than the alignment ($\forall A \in \Lambda, A \subseteq Cn^\alpha(A) \subseteq Cn_{\Omega, \Lambda}^\alpha(A)$). If the alignment is not satisfiable, then any correspondence is one of its α -consequences.

Similarly, the ω -consequences of an ontology in a network are formulas that are satisfied in all models of the ontology selected by the network.

Definition 14 (ω -Consequence of an ontology in a network of ontologies). *Given a finite set of ontologies Ω and a finite set of alignments Λ between pairs of ontologies in Ω , a formula δ in the ontology language of $o \in \Omega$ is an ω -consequence of o in (Ω, Λ) (denoted by $o \models_{\Omega, \Lambda} \delta$) if and only if for all models $\langle m, \Delta \rangle$ of (Ω, Λ) , $m_o \models \delta$ (the set of ω -consequences of o is denoted by $Cn_{\Omega, \Lambda}^\omega(o)$).*

The ω -closure of an ontology is the set of its ω -consequences. According to these definitions, $Cn^\omega(o) = Cn_{\{\{o\}, \emptyset\}}^\omega(o)$. These ω -consequences are larger than the classical consequences of the ontology ($\forall o \in \Omega, o \subseteq Cn^\omega(o) \subseteq Cn_{\Omega, \Lambda}^\omega(o)$) because they rely on a smaller set of models.

Example 9 (ω -consequences). *The simple consequences of the ontology o'_3 are:*

$$Cn^\omega(o'_3) = \left\{ \begin{array}{ll} b_3 \sqsubseteq a_3, c_3 \sqsubseteq a_3, & g_3 \sqsubseteq b_3 \\ d_3 \sqsubseteq c_3, e_3 \sqsubseteq c_3, & f_3 \sqsubseteq b_3 \\ f_3 \perp c_3, b_3 \perp c_3 & f_3 \sqsubseteq a_3, \\ d_3 \sqsubseteq a_3, e_3 \sqsubseteq a_3, & g_3 \perp c_3 \\ d_3 \perp b_3, e_3 \perp b_3 & g_3 \sqsubseteq a_3 \end{array} \right\}$$

while within (Ω', Λ) of Example 7, there are even more consequences:

$$Cn_{\Omega', \Lambda}^\omega(o'_3) = \left\{ \begin{array}{ll} b_3 \sqsubseteq a_3, c_3 \sqsubseteq a_3, & g_3 \sqsubseteq b_3 \\ d_3 \sqsubseteq c_3, e_3 \sqsubseteq c_3, & f_3 \sqsubseteq b_3 \\ f_3 \perp c_3, b_3 \perp c_3 & f_3 \sqsubseteq a_3, \\ d_3 \sqsubseteq a_3, e_3 \sqsubseteq a_3, & g_3 \perp c_3, \\ d_3 \perp b_3, e_3 \perp b_3 & g_3 \sqsubseteq a_3 \\ & b_3 \sqsupseteq e_3 \end{array} \right\}$$

From the notion of consequence, we introduce semantic morphism.

Definition 15 (Semantic morphism between networks of ontologies). *Given two networks of ontologies, (Ω, Λ) and (Ω', Λ') , a semantic morphism between (Ω, Λ) and (Ω', Λ') , is $\exists \langle h, k \rangle$, a pair of morphisms: $h : \Omega \rightarrow \Omega'$ and $k : \Lambda \rightarrow \Lambda'$ such that $\forall o \in \Omega, \exists h(o) \in \Omega'$ and $h(o) \models_{\Omega', \Lambda'} o$ and $\forall A \in \Lambda(o, o')$, $\exists k(A) \in \Lambda'(h(o), h(o'))$ and $k(A) \models_{\Omega', \Lambda'} A$.*

We can also define the closure of a network of ontologies by the network of ontologies which replaces each ontology by its ω -closure and each alignment by its α -closure:

$$Cn((\Omega, \Lambda)) = \langle \{Cn_{\Omega, \Lambda}^\omega(o)\}_{o \in \Omega}, \{Cn_{\Omega, \Lambda}^\alpha(o, o')\}_{o, o' \in \Omega} \rangle$$

or alternatively:

$$\langle \{ \{ \delta \mid \forall \langle m, \Delta \rangle \in \mathcal{M}((\Omega, \Lambda)), m_o \models \delta \} \}_{o \in \Omega}, \{ \{ \mu \mid \forall \langle m, \Delta \rangle \in \mathcal{M}((\Omega, \Lambda)), m_o, m_{o'} \models \Delta \mu \} \}_{o, o' \in \Omega} \rangle$$

We also use the notation $Cn_{(\Omega, \Lambda)}^\alpha$ for $Cn_{\Omega, \Lambda}^\alpha$ and $Cn_{(\Omega, \Lambda)}^\omega$ for $Cn_{\Omega, \Lambda}^\omega$.

Example 10 (Full network closure). *Here is the closure of the network of ontologies $\langle \Omega', \Lambda \rangle$ of Example 7 (the first set is the syntactic form corresponding to the alignment or ontology, the second set is what is added by the local closure and the last set what is added by the ω -closure or α -closure):*

$$\begin{aligned}
Cn_{\Omega', \Lambda}^{\omega}(o_1) &= Cn^{\omega}(o_1) = \left\{ \begin{array}{l} b_1 \sqsubseteq a_1, c_1 \sqsubseteq a_1, \\ d_1 \sqsubseteq c_1, e_1 \sqsubseteq c_1 \end{array} \right\} \cup \{ d_1 \sqsubseteq a_1, e_1 \sqsubseteq a_1 \} \\
Cn_{\Omega', \Lambda}^{\omega}(o_2) &= Cn^{\omega}(o_2) = \left\{ \begin{array}{l} b_2 \sqsubseteq a_2, c_2 \sqsubseteq a_2, \\ g_2 \sqsubseteq b_2, f_2 \sqsubseteq b_2, \\ d_2 \sqsubseteq c_2, e_2 \sqsubseteq c_2, \end{array} \right\} \cup \left\{ \begin{array}{l} d_2 \sqsubseteq a_2, e_2 \sqsubseteq a_2, \\ f_2 \sqsubseteq a_2, g_2 \sqsubseteq a_2 \end{array} \right\} \\
Cn_{\Omega', \Lambda}^{\omega}(o'_3) &= \left\{ \begin{array}{l} b_3 \sqsubseteq a_3, c_3 \sqsubseteq a_3, \\ g_3 \sqsubseteq b_3, d_3 \sqsubseteq c_3, \\ e_3 \sqsubseteq c_3, f_3 \sqsubseteq b_3, \\ b_3 \perp c_3 \end{array} \right\} \cup \left\{ \begin{array}{l} f_3 \sqsubseteq a_3, g_3 \sqsubseteq a_3, \\ d_3 \sqsubseteq a_3, e_3 \sqsubseteq a_3, \\ d_3 \perp b_3, e_3 \perp b_3, \\ d_3 \perp f_3, d_3 \perp g_3, \\ e_3 \perp f_3, e_3 \perp g_3, \\ f_3 \perp c_3, g_3 \perp c_3 \end{array} \right\} \cup \{ b_3 \sqsupseteq e_3 \}
\end{aligned}$$

The network does not introduce new assertions in the two first ontologies, but the last one receives a new assertion. Similarly, for alignments, their local closure does not provide new correspondences, but the α -closure becomes larger. These alignment closures are:

$$\begin{aligned}
Cn_{\Omega', \Lambda}^{\alpha}(o_1, o_2) &= \{ b_1 \leq d_2 \} \cup \{ b_1 \leq a_2, b_1 \leq c_2 \} \\
Cn_{\Omega', \Lambda}^{\alpha}(o_2, o'_3) &= \{ c_2 \leq b_3 \} \cup \left\{ \begin{array}{l} c_2 \leq a_3, d_2 \leq a_3, e_2 \leq a_3 \\ d_2 \leq b_3, e_2 \leq b_3, \\ c_2 \perp c_3, c_2 \perp d_3, c_2 \perp e_3 \\ d_2 \perp c_3, d_2 \perp d_3, d_2 \perp e_3 \\ e_2 \perp c_3, e_2 \perp d_3, e_2 \perp e_3 \\ d_2 \geq e_3, c_2 \geq e_3, a_2 \geq e_3 \end{array} \right\} \\
Cn_{\Omega', \Lambda}^{\alpha}(o_1, o'_3) &= \left\{ \begin{array}{l} e_1 \geq f_3, \\ b_1 \geq e_3 \end{array} \right\} \cup \left\{ \begin{array}{l} c_1 \geq f_3, a_1 \geq f_3, a_1 \geq e_3, \\ b_1 \perp c_3, b_1 \perp d_3, b_1 \perp e_3, \\ b_1 \leq b_3, b_1 \leq a_3 \end{array} \right\}
\end{aligned}$$

Such a representation is highly redundant as closures usually are.

The closure of a network of ontologies may introduce non empty alignments between ontologies which were not previously connected or empty. This is possible because constraints do not come locally from the alignment but from the whole network of ontologies. Such a formalism contributes to the definition of the meaning of alignments: it describes what are the consequences of ontologies with alignments, i.e., what can be deduced by an agent.

Property 7. *Cn is a closure operation on normalised networks of ontologies¹.*

This property can rely on a lemma mirroring Property 5.

Lemma 1 (Consequence isotony). *Let $\langle \Omega, \Lambda \rangle$ and $\langle \Omega', \Lambda' \rangle$ be two networks of ontologies,*

$$\langle \Omega, \Lambda \rangle \sqsubseteq \langle \Omega', \Lambda' \rangle \Rightarrow Cn(\langle \Omega, \Lambda \rangle) \sqsubseteq Cn(\langle \Omega', \Lambda' \rangle)$$

¹A closure operation Cn in a set S satisfies three properties: $\forall X, Y \in S : X \sqsubseteq Cn(X)$, $Cn(X) = Cn(Cn(X))$, and $X \sqsubseteq Y \Rightarrow Cn(X) \sqsubseteq Cn(Y)$.

Proof of Lemma 1. From Property 5, we know that there exists $h : \Omega \rightarrow \Omega'$ and that $\forall \langle m', \Delta' \rangle \in \mathcal{M}(\langle \Omega', \Lambda' \rangle)$, $\exists \langle m, \Delta \rangle \in \mathcal{M}(\langle \Omega, \Lambda \rangle)$; $\forall o \in \Omega$, $m_o = m'_{h(o)}$ and $\Delta \ll \Delta'$. Hence, $\exists h : \Omega \rightarrow \Omega'$ such that $\forall o \in \Omega$, each model of $h(o)$ is a model of o , so $\{m_o \mid \langle m, \Delta \rangle \in \mathcal{M}(\langle \Omega, \Lambda \rangle)\} \supseteq \{m'_{h(o)} \mid \langle m', \Delta' \rangle \in \mathcal{M}(\langle \Omega', \Lambda' \rangle)\}$. This entails $Cn_{\Omega, \Lambda}^{\omega}(o) \subseteq Cn_{\Omega', \Lambda'}^{\omega}(h(o))$. This also means that $\forall o, o' \in \Omega$, each pair of models $\langle m'_{h(o)}, m'_{h(o')} \rangle$ of $\langle h(o), h(o') \rangle$ is also a pair of models of $\langle o, o' \rangle$. Considering $\mu \in Cn_{\Omega, \Lambda}^{\alpha}(o, o')$, either $\forall \langle m', \Delta' \rangle \in \mathcal{M}(\langle \Omega', \Lambda' \rangle)$, $m'_{h(o)}, m'_{h(o')} \models_{\Delta'} \mu$ and then $\mu \in Cn_{\Omega', \Lambda'}^{\alpha}(h(o), h(o'))$, or $\exists \langle m', \Delta' \rangle \in \mathcal{M}(\langle \Omega', \Lambda' \rangle)$; $m'_{h(o)}, m'_{h(o')} \not\models_{\Delta'} \mu$. In this latter case, $\exists \langle m, \Delta \rangle \in \mathcal{M}(\langle \Omega, \Lambda \rangle)$, such that $m_o = m'_{h(o)}$. This is a model of $\langle \Omega, \Lambda \rangle$ because this network put less constraints on Δ than $\langle \Omega', \Lambda' \rangle$. But this contradicts the hypothesis that $\mu \in Cn_{\Omega, \Lambda}^{\alpha}(o, o')$. Hence, $Cn_{\Omega, \Lambda}^{\alpha}(o, o') \subseteq Cn_{\Omega', \Lambda'}^{\alpha}(h(o), h(o'))$, so $Cn(\langle \Omega, \Lambda \rangle) \sqsubseteq Cn(\langle \Omega', \Lambda' \rangle)$. \square

Proof of Property 7. $\forall \langle \Omega, \Lambda \rangle$ and $\langle \Omega', \Lambda' \rangle$ normalised networks of ontologies:

- $\langle \Omega, \Lambda \rangle \sqsubseteq Cn(\langle \Omega, \Lambda \rangle)$, because $\forall \langle h, k \rangle$ such that $\forall o, o' \in \Omega$, $h(o) = Cn_{\Omega, \Lambda}^{\omega}(o)$ and $k(\lambda(o, o')) = Cn_{\Omega, \Lambda}^{\alpha}(o, o')$, (i) $o \subseteq Cn_{\Omega, \Lambda}^{\omega}(o)$, because $\forall \delta \in o$, $\forall \langle m, \Delta \rangle \in \mathcal{M}(\langle \Omega, \Lambda \rangle)$, $m_o \models o$, hence $m_o \models \delta$, so $\delta \in Cn_{\Omega, \Lambda}^{\omega}(o)$; and (ii) $\lambda(o, o') \subseteq Cn_{\Omega, \Lambda}^{\alpha}(o, o')$, because $\forall \mu \in \lambda(o, o')$, $\forall \langle m, \Delta \rangle \in \mathcal{M}(\langle \Omega, \Lambda \rangle)$, $m_o, m_{o'} \models_{\Delta} \mu$, hence $\mu \in Cn_{\Omega, \Lambda}^{\alpha}(\lambda(o, o'))$.
- $\forall \langle m, \Delta \rangle \in \mathcal{M}(\langle \Omega, \Lambda \rangle)$, (i) $\forall o \in \Omega$, $m_o \in \mathcal{M}(o)$ and $\forall o, o' \in \Omega$, $m_o, m_{o'} \models_{\Delta} \lambda(o, o')$, (ii) $\forall o \in \Omega$, $m_o \in \mathcal{M}(Cn_{\Omega, \Lambda}^{\omega}(o))$, and (iii) $\forall o, o' \in \Omega$, $m_o, m_{o'} \models_{\Delta} Cn_{\Omega, \Lambda}^{\alpha}(o, o')$ (the two latter assertions because the closure contains elements true in all models). In consequence, $\langle m, \Delta \rangle \in \mathcal{M}(Cn(\langle \Omega, \Lambda \rangle))$, which means that $\mathcal{M}(\langle \Omega, \Lambda \rangle) \subseteq \mathcal{M}(Cn(\langle \Omega, \Lambda \rangle))$, and thus $Cn(\langle \Omega, \Lambda \rangle) \supseteq Cn(Cn(\langle \Omega, \Lambda \rangle))$ (less models means more consequences). By the first clause, $Cn(\langle \Omega, \Lambda \rangle) \sqsubseteq Cn(Cn(\langle \Omega, \Lambda \rangle))$, so, $Cn(\langle \Omega, \Lambda \rangle) \equiv Cn(Cn(\langle \Omega, \Lambda \rangle))$.
- Consequence isotony is proved by Lemma 1

\square

Example 11. Figure 2 displays an extreme example of a network. This network is inconsistent in the interpretation of Example 3, though none of its ontologies nor alignments is inconsistent. The inconsistency manifests itself by starting with the network without one of the correspondences and revising it by this correspondence. It can be only solved by suppressing one of the other correspondences.

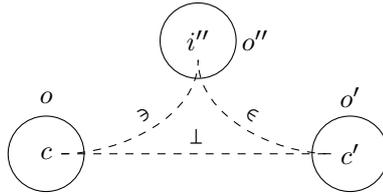


Figure 2: Globally inconsistent alignment pattern.

5 The categories of networks of ontologies

We introduce categories of networks of ontologies. This allows to consider networks of ontologies from a more abstract perspective and to take advantage of general operations, such as pullbacks.

5.1 Syntactic category of networks of ontologies

Networks of ontologies, together with the morphisms previously defined form a category.

Definition 16 (Category \mathcal{NOO}). *Let \mathcal{NOO} be the structure made of:*

- objects are networks of ontologies as in Definition 2;
- morphisms are morphisms between networks of ontologies as in Definition 5;

such that

- the composition of morphisms is simply function composition: $\langle h, k \rangle \circ \langle h', k' \rangle = \langle h \circ h', k \circ k' \rangle$;
- the identity morphism $1_{\langle \Omega, \Lambda \rangle} = \langle 1_\Omega, 1_\Lambda \rangle$ is defined as the morphism associating an ontology to itself and an alignment to itself.

Property 8. \mathcal{NOO} is a category

Proof. The proof directly follows from the definition: Given a morphism $\langle h, k \rangle$ between networks $\langle \Omega, \Lambda \rangle$ and $\langle \Omega', \Lambda' \rangle$ and $\langle h', k' \rangle$ between networks $\langle \Omega', \Lambda' \rangle$ and $\langle \Omega'', \Lambda'' \rangle$, the result of their composition $\langle h' \circ h, k' \circ k \rangle$ is a morphism between $\langle \Omega, \Lambda \rangle$ and $\langle \Omega'', \Lambda'' \rangle$. Indeed, if $\forall o \in \Omega, o \subseteq h(o)$, and $\forall o' \in \Omega', o' \subseteq h'(o')$, then $o \subseteq h(h'(o')) = h' \circ h(o)$. If $\forall A \in \Lambda(o, p), k(A) \in \Lambda'(h(o), h(p))$ and $\forall A' \in \Lambda'(o', p'), k'(A') \in \Lambda''(h'(o'), h'(p'))$ then $\forall A \in \Lambda(o, p), k' \circ k(A) \in \Lambda''(h' \circ h(o), h' \circ h(p))$. If $\forall A \in \Lambda(o, p), A \subseteq k(A)$ and $\forall A' \in \Lambda'(o', p'), A' \subseteq k'(A')$ then $\forall A \in \Lambda(o, p), A \subseteq k(A) \subseteq k'(k(A)) = k' \circ k(A)$.

Composition is associative, i.e., if $f : A \rightarrow B, g : B \rightarrow C, h : C \rightarrow D$, then $h \circ (g \circ f) = (h \circ g) \circ f$, simply because function composition is associative.

The identity morphism is indeed a morphism, because $\forall o \in \Omega, o \subseteq o = 1_\Omega(o)$ and $\forall A \in \Lambda(o, p), 1_\Lambda(A) = A \in \Lambda(o, p) = \Lambda(1_\Omega(o), 1_\Omega(p))$. Moreover, $A \subseteq A = 1_\Lambda(A)$.

Finally, $\forall \langle h, k \rangle$ between networks $\langle \Omega, \Lambda \rangle$ and $\langle \Omega', \Lambda' \rangle, 1_{\langle \Omega, \Lambda \rangle} \circ \langle h, k \rangle = \langle 1_\Omega \circ h, 1_\Lambda \circ k \rangle = \langle h, k \rangle = \langle h \circ 1_{\Omega'}, k \circ 1_{\Lambda'} \rangle = \langle h, k \rangle \circ 1_{\langle \Omega', \Lambda' \rangle}$. \square

A conjunction (meet) operator between networks of ontologies can be introduced generally in a standard way from subsumption, as the greatest common subsumee, but it would not be necessarily unique. However, it is possible to introduce a fibred meet, denoted $\overline{\cap}$, which define the conjunction of a set of networks along a set of isomorphisms to a common generating network.

Definition 17 (Fibred meet of networks of ontologies). *Given a network of ontologies $\langle \Omega, \Lambda \rangle$ and a finite family of networks of ontologies, $\{\langle \Omega_j, \Lambda_j \rangle\}_{j \in J}$, such that $\exists \langle h_j, k_j \rangle_{j \in J}$, pairs of isomorphisms: $h_j : \Omega_j \rightarrow \Omega$ and $k_j : \Lambda_j \rightarrow \Lambda$ with $\forall A \in \Lambda(o, o'), k_j(A) \in \Lambda(h_j(o), h_j(o'))$, the fibred meet of $\{\langle \Omega_j, \Lambda_j \rangle\}_{j \in J}$ with respect to $\langle h_j, k_j \rangle_{j \in J}$ is*

$$\overline{\cap}_{j \in J} \langle \Omega_j, \Lambda_j \rangle = \langle \{ \bigcap_{j \in J} h_j^{-1}(o) \}_{o \in \Omega}, \{ \bigcap_{j \in J} k_j^{-1}(A) \}_{A \in \Lambda} \rangle$$

We may denote $\overline{\cap}_{j \in J} \langle \Omega_j, \Lambda_j \rangle$ as $\langle \overline{\Omega}, \overline{\Lambda} \rangle$.

The generating morphisms must be isomorphisms because, if they were not injective, the π_j cannot be defined (there would be two candidate targets), and if they were not surjective, they would not necessarily cover the whole source network of alignment (see Figure 3).

The degenerated case of a fibred meet is when all ontologies are empty (and hence alignments are empty as well).

The fibred meet is a pullback in the \mathcal{NOO} category.

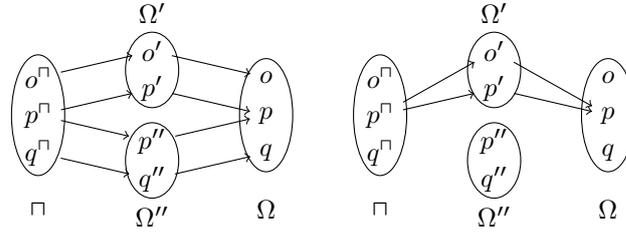


Figure 3: Two cases of non isomorphic generators (ellipsis represents sets of ontologies, letters represent ontologies). On the left, the generating morphisms are not surjective, hence functions generated from \square to Ω' and Ω'' are not morphisms (they do not cover \square). On the right, the generating morphism is not injective, so it is not possible to create a morphism from \square to Ω' .

Property 9. *The fibred meet is a pullback for the $\mathcal{N}\mathcal{O}\mathcal{O}$ category.*

Proof. $\pi_{j'} = \langle h'_{j'}, k'_{j'} \rangle$ such that $h'_{j'}(\bigcap_{j \in J} h_j^{-1}(o)) = h_j^{-1}(o)$, and $k'_{j'}(\bigcap_{j \in J} k_j^{-1}(A)) = k_j^{-1}(A)$. This is a morphism since (1) $\bigcap_{j \in J} h_j^{-1}(o) \subseteq h_j^{-1}(o)$, (2) $\bigcap_{j \in J} k_j^{-1}(A) \subseteq k_j^{-1}(A)$, and (3) if $\bigcap_{j \in J} k_j^{-1}(A) \in \overline{\Lambda}(\bigcap_{j \in J} h_j^{-1}(o), \bigcap_{j \in J} h_j^{-1}(o'))$, then $k'_{j'}(\bigcap_{j \in J} k_j^{-1}(A)) \in \Lambda_{j'}(h_j^{-1}(o), h_j^{-1}(o')) = \Lambda_{j'}(h'_{j'}(\bigcap_{j \in J} h_j^{-1}(o)), h'_{j'}(\bigcap_{j \in J} h_j^{-1}(o'))$.

Moreover, the diagram commutes, i.e., $\forall j', j'' \in J$, $\langle h_{j'}, k_{j'} \rangle \circ \pi_{j'} = \langle h_{j''}, k_{j''} \rangle \circ \pi_{j''}$, because $h'_{j'} \circ h_{j'}(\bigcap_{j \in J} h_j^{-1}(o)) = o = h'_{j''} \circ h_{j''}(\bigcap_{j \in J} h_j^{-1}(o))$ and $k'_{j'} \circ k_{j'}(\bigcap_{j \in J} k_j^{-1}(A)) = A = k'_{j''} \circ k_{j''}(\bigcap_{j \in J} k_j^{-1}(A))$ because $\langle h_j, k_j \rangle$ are pairs of isomorphism.

Finally, $\forall \theta_{j'}(x) = h_j^{-1}(o)$, then u can be defined such that $u(x) = \bigcap_{j \in J} h_j^{-1}(o)$. Similarly, $\forall \theta_{j'}(Y) = k_j^{-1}(A)$, then $u(Y)$ can be defined as $u(Y) = \bigcap_{j \in J} h_j^{-1}(A)$ then $\theta_{j'}(Y) = u \circ \pi_{j'}(Y)$, so the fibred meet is universal. \square

$$\begin{array}{ccc}
 \langle \Omega', \Lambda' \rangle & \overset{u}{\dashrightarrow} & \overline{\prod}_{j \in J} \langle \Omega_j, \Lambda_j \rangle \\
 \searrow \theta_j & & \downarrow \pi_j \\
 & & \langle \Omega_j, \Lambda_j \rangle \xrightarrow{\langle h_j, k_j \rangle} \langle \Omega, \Lambda \rangle
 \end{array}$$

Such a fibred meet is well-defined and unique up to isomorphism for any set of subnetworks of a particular generator network. It does not exist for any diagrams with morphisms to an object, the morphisms have to be isomorphic as presented in Figure 3.

It satisfies the premises of Property 10.

Property 10. *Let $\langle \Omega, \Lambda \rangle$ be a network of ontologies and $\{\langle \Omega_j, \Lambda_j \rangle\}_{j \in J}$ a finite family of networks of ontologies whose fibred meet with respect to $\langle \Omega, \Lambda \rangle$ is defined through morphisms $\langle h_j, k_j \rangle$.*

$$\forall j \in J, \quad \overline{\prod}_{j' \in J} \langle \Omega_{j'}, \Lambda_{j'} \rangle \sqsubseteq \langle \Omega_j, \Lambda_j \rangle \sqsubseteq \langle \Omega, \Lambda \rangle$$

Proof. $\langle \Omega_j, \Lambda_j \rangle \sqsubseteq \langle \Omega, \Lambda \rangle$ is straightforward because the generating pairs of morphism $\langle h_j, k_j \rangle$ are already such that this is true.

$\overline{\prod}_{j' \in J} \langle \Omega_{j'}, \Lambda_{j'} \rangle \sqsubseteq \langle \Omega_j, \Lambda_j \rangle$ because it is possible to consider the pair of morphisms $\langle h'_{j'}, k'_{j'} \rangle$ such that $h'_{j'}(\bigcap_{j' \in J} h_{j'}^{-1}(o)) = h_j^{-1}(o)$ (and obviously, $\bigcap_{j' \in J} h_{j'}^{-1}(o) \subseteq h_j^{-1}(o)$). Similarly, $k'_{j'}(\bigcap_{j' \in J} k_{j'}^{-1}(A)) =$

$k_j^{-1}(A)$ and again, $\bigcap_{j' \in J} k_{j'}^{-1}(A) \subseteq k_j^{-1}(A)$. These are graph morphisms because all $k_{j'}^{-1}(A)$ are alignments between the precursors of the ontologies of A . \square

The fibred meet generated by consistent subnetworks is always consistent since it is subsumed by consistent networks (downward consistency preservation).

Moreover, if the ontologies in the networks are closed, then the fibred meet is closed as well (because it intersects closed networks).

5.2 Category of weighted networks of ontologies

This category can be generalised for taking weights into account [3]. Following the presentation of [8], the relationship between two entities can be assigned a degree of confidence, or weight, taken from a confidence structure.

Definition 18 (Confidence structure). *A confidence structure is an ordered set of degrees $\langle \Xi, \leq \rangle$ for which there exists the greatest element \top and the smallest element \perp .*

The usage of confidence degrees is such that the higher the degree with regard to \leq , the more likely the relation holds. It is convenient to interpret the greatest element as the Boolean true and the least element as the Boolean false. The function $\kappa_A(\mu) : Q_L(o) \times Q_{L'}(o') \times \Theta \rightarrow \Xi$ provides the confidence for a correspondence μ in an alignment A .

Networks of ontologies may be defined on weighted alignments, by simply integrating κ_A in A . Morphisms can be defined by taking into account the confidence (or weight) associated to correspondences.

Definition 19 (Weight-aware morphism between networks of ontologies). *Given two networks of ontologies, $\langle \Omega, \Lambda \rangle$ and $\langle \Omega', \Lambda' \rangle$, a weight-aware morphism between $\langle \Omega, \Lambda \rangle$ and $\langle \Omega', \Lambda' \rangle$, is $\langle h, k \rangle$, a pair of morphisms: $h : \Omega \rightarrow \Omega'$ and $k : \Lambda \rightarrow \Lambda'$ such that $\forall o \in \Omega, \exists h(o) \in \Omega'$ and $o \subseteq h(o)$ and $\forall A \in \Lambda(o, o'), \forall \mu \in A, \exists k(A) \in \Lambda'(h(o), h(o'))$ and $\mu' \in k(A)$ such that $\mu = \mu'$ and $\kappa_A(\mu) \leq \kappa_{k(A)}(\mu')$.*

They can be used for defining a new category:

Definition 20 (Category $\mathcal{NCO}_{\mathcal{W}}$). *Let $\mathcal{NCO}_{\mathcal{W}}$ be the structure made of:*

- objects are networks of ontologies as in Definition 2 with weighted alignments;
- morphisms are weight-aware morphisms between networks of ontologies as in Definition 19;

Obviously this is a category again, we omit the proof.

There is a natural family of functors F_w , with $w \in \Xi$ a threshold value, which are simply quotientation functions.

Definition 21 (F_w function). *The function $F_w : \mathcal{NCO}_{\mathcal{W}} \rightarrow \mathcal{NCO}$ with w within the codomain of κ defined as:*

$$\begin{aligned} F_w(\langle \Omega, \Lambda \rangle) &= \langle \Omega, \{F_w(A) \mid A \in \Lambda\} \rangle \\ F_w(\langle h, k \rangle) &= \langle h, F_w(k) \rangle \\ F_w(A) &= \{\mu \in A \mid \kappa(\mu) \geq w\} \\ F_w(k)(\mu) &= k(\mu) \end{aligned}$$

Property 11. F_w is a functor

Proof. Clearly, $\forall k : \Lambda \rightarrow \Lambda', F_w(k) : F_w(\Lambda) \rightarrow F_w(\Lambda'), F_w(k)(F_w(A)) = \{\mu \in k(A) \mid \kappa(\mu) \geq w\} = F_w(k(A))$. Moreover, if $k(A) \in \Lambda'$, then $F_w(k(A)) \in F_w(\Lambda)$. Finally, $F_w(k)$ is a (weight-blind) morphism because, if $\mu \in F_w(A)$, then $\kappa(\mu) \leq \kappa(k(\mu))$, so, $k(\mu) \in F_w(k)(A)$. \square

These functors are both applying thresholds to the content of alignments and ignoring the weights, it is also possible, with similar arguments to define functors which perform both functions independently (they do not commute: weight-ignorance can only be applied at the end of the chain).

5.3 Further refinements

Further refinements may be introduced such as considering that relations are not independent, e.g., that $\langle e, =, e' \rangle$ entails $\langle e, \sqsubseteq, e' \rangle$ and thus that morphisms should be defined by taking this into account. This is slightly more difficult because several correspondences may be equated to a single correspondence. This approach may be refined by using algebras of relations [7].

Finally, considering semantic morphisms would complete the picture.

6 Related work

Obviously the work presented in (§3) is related to this one focussing on one specific semantics for alignments and eventually for ontologies.

In addition, the work around DOL [1] is a wider effort building from the ground (logic) up to ontologies and alignments through describing the semantics of ontology and alignment languages. Instead, we start from these ontologies and alignments and abstract from their underlying semantics to determine properties on top of which general purpose operations can be built. Hopefully, these two efforts will meet.

7 Conclusion

This report brought two modest contributions:

- a semantics for networks of ontologies which is parameterized by actual semantics and properties independent from the chosen semantics;
- an abstraction of networks of ontologies into categories on which pullbacks can be defined.

These contributions provide general properties that can be used for defining concrete operations. We used them for defining revision of networks of ontologies.

The category of networks have been defined purely syntactically with a disconnection to the semantics. Doing it semantically would be useful as well.

Acknowledgements

I am thankful to Antoine Zimmermann, Manuel Atencia and Armen Inants for fruitful discussions.

References

- [1] The distributed ontology, model and specification language (DOL). Technical Report ad/2014-10-01, Object management group, 2014.
- [2] Grigoris Antoniou, Paul Groth, Frank van Harmelen, and Rinke Hoekstra. *A semantic web primer*. The MIT press, Cambridge (MA US), 3rd edition, 2012.

- [3] Manuel Atencia, Jérôme Euzenat, Giuseppe Pirrò, and Marie-Christine Rousset. Alignment-based trust for resource finding in semantic P2P networks. In *Proc. 10th international semantic web conference (ISWC), Bonn (DE)*, volume 7031 of *Lecture notes in computer science*, pages 51–66, 2011.
- [4] Alexander Borgida and Luciano Serafini. Distributed description logics: Assimilating information from peer sources. *Journal on data semantics*, I:153–184, 2003.
- [5] Bernardo Cuenca Grau, Bijan Parsia, and Evren Sirin. Combining OWL ontologies using \mathcal{E} -connections. *Journal of web semantics*, 4(1):40–59, 2006.
- [6] Jérôme Euzenat. Semantic precision and recall for ontology alignment evaluation. In *Proc. 20th international joint conference on artificial intelligence (IJCAI), Hyderabad (IN)*, pages 348–353, 2007.
- [7] Jérôme Euzenat. Algebras of ontology alignment relations. In *Proc. 7th international semantic web conference (ISWC), Karlsruhe (DE)*, pages 387–402, 2008.
- [8] Jérôme Euzenat and Pavel Shvaiko. *Ontology matching*. Springer, Heidelberg (DE), 2nd edition, 2013.
- [9] Enrico Franconi, Gabriel Kuper, Andrei Lopatenko, and Luciano Serafini. A robust logical and computational characterisation of peer-to-peer database systems. In *Proc. VLDB workshop on databases, information systems and peer-to-peer computing (DBISP2P), Berlin (DE)*, pages 64–76, 2003.
- [10] Chiara Ghidini and Fausto Giunchiglia. Local models semantics, or contextual reasoning = locality + compatibility. *Artificial intelligence*, 127(2):221–259, 2001.
- [11] Chiara Ghidini and Luciano Serafini. Distributed first order logics. In *Proc. 2nd conference on frontiers of combining systems (FroCoS), Amsterdam (NL)*, pages 121–139, 1998.
- [12] Pascal Hitzler, Markus Krötzsch, and Sebastian Rudolph. *Foundations of semantic web technologies*. Chapman & Hall/CRC, 2009.
- [13] Maurizio Lenzerini. Data integration: A theoretical perspective. In *Proc. 21st symposium on principles of database systems (PODS), Madison (WI US)*, pages 233–246, 2002.
- [14] Antoine Zimmermann. *Sémantique des réseaux de connaissances: gestion de l'hétérogénéité fondée sur le principe de médiation*. Thèse d'informatique, Université Joseph Fourier, Grenoble (FR), 2008.
- [15] Antoine Zimmermann. Logical formalisms for agreement technologies. In Sasha Ossowski, editor, *Agreement technologies*, chapter 5, pages 69–82. Springer Verlag, Heidelberg (DE), 2013.
- [16] Antoine Zimmermann and Jérôme Euzenat. Three semantics for distributed systems and their relations with alignment composition. In *Proc. 5th international semantic web conference (ISWC), Athens (GA US)*, volume 4273 of *Lecture notes in computer science*, pages 16–29, 2006.

Contents

1 Motivation	3
2 Alignments and networks of ontologies	3
3 Semantics of networks of ontologies	7
4 Consistency, consequence and closure	10
5 The categories of networks of ontologies	14
5.1 Syntactic category of networks of ontologies	14
5.2 Category of weighted networks of ontologies	16
5.3 Further refinements	17
6 Related work	17
7 Conclusion	17



**RESEARCH CENTRE
GRENOBLE – RHÔNE-ALPES**

Inovallée
655 avenue de l'Europe Montbonnot
38334 Saint Ismier Cedex

Publisher
Inria
Domaine de Voluceau - Rocquencourt
BP 105 - 78153 Le Chesnay Cedex
inria.fr

ISSN 0249-6399