Preliminary Version of a paper published in IGPL

Ontology-Revision Operators Based on Reinterpretation

Carola Eschenbach and Özgür L. Özçep
Department for Informatics (AB WSV),
MIN Faculty, University of Hamburg,
Vogt-Kölln-Str. 30, D-22527 Hamburg, Germany.
E-mail: {eschenbach, oezcep}@informatik.uni-hamburg.de

Abstract

Communication between natural or artificial agents relies on the use of a common vocabulary. Since sharing terms does not necessarily imply that the terms have exactly the same meanings for all agents, integrating (trigger) statements into a formal ontology requires mechanisms for resolving conflicts that are caused by the ambiguity of terms specified in different but similar ontologies.

We define and analyze a family of ontology-revision operators that resolve conflicts by disambiguating concept symbols occurring in both the ontology and the trigger statements. The operators yield bridging axioms relating the different readings of the terms and, by including representations for both readings, preserve the initial ontology as well as the trigger statements. The operators differ regarding which reading of the ambiguous term is assigned to further uses of the common term and regarding the semantic relation assumed between the two readings. The ontology-revision operators are analyzed regarding their adaptability to consistent sequences of trigger statements. One group of operators (type 1) preserves all conflicts with the trigger sequence. Operators from the other group (type 2) can resolve the conflicts, which is demonstrated by showing under which conditions weak type-2 operators yield stabilizing sequences of ontologies. Stronger type-2 operators can result in closer approximations of the terminology underlying the sequence of trigger statements but can also yield non-stabilizing sequences of ontologies.

1 Introduction

Communication between natural or artificial agents relies on the use of common terms with shared meanings. This precondition, however, cannot always be established in advance. While human users of natural language have flexible means for handling situations where different readings of the same term become

obvious, such mechanisms of reinterpretation are not well studied for logic-based agents.

The approach presented in this article aims at handling the communication between agents that hold kindred ontologies for a common domain where terminological conflicts are the exception rather than the rule. Therefore no preprocessing stage of aligning the ontologies is assumed. A sender generates a consistent sequence of statements based on its ontology and the receiver integrates these statements into its ontology. The terminological differences are discovered when the sender presents statements that conflict with the ontology of the receiver.

In the case of observing a conflict, the receiver could request a specification of the critical terms and start an ontology-integration process on this basis [9], [21]. Formal approaches to semantic integration mainly focus on the problem of integrating two ontologies and of establishing a semantic mapping between the name spaces of different (accessible) ontologies.[1] However, we will look at a different strategy to treat the conflicts identified during a communication process. The incoming sequence of statements is incrementally integrated into the initial ontology and thereby the conflicts are resolved one by one. Within the classification of approaches to ontology change by Flouris and colleagues [6], the approach presented in this article resolves terminological conflicts in the process of ontology merging.

The successful integration of conflicting information into a knowledge base is well-studied in the context of theories of belief change. Within this area, operators for belief revision [1] and operators for belief update [12] are formalized. Operators for belief revision resolve conflicts based on the assumption that the knowledge base contains incorrect statements. Operators for belief update resolve conflicts based on the assumption that the knowledge base contains outdated statements. In both cases, the possibility of conflicts based on terminological ambiguity is not considered, the vocabulary of the knowledge base is kept constant, and the elimination of statements conflicting with the new information from the knowledge base is accepted.

In this article, we define operators for ontology revision that combine an ontology and a statement (of restricted syntactic complexity) yielding an ontology combining the information from both sources. Observed conflicts between the ontology and the incoming statements are resolved by assuming the involved concept terms to be ambiguous. According to the observation that different readings of an ambiguous term are in many cases semantically related, the distinction between the different readings will also involve hypotheses on their semantic relatedness. Since the integration of conflicting statements involves the distinction between different readings of one term, we call the underlying strategy *reinterpretation*.

An example of such a knowledge integration scenario is a knowledge-based software agent (R) that holds an ontology O_R and sends a request (e.g., 'List all cheap books on thermodynamics') to a book-selling agent (S). Agent S

[1] Noy [19] presents an overview of approaches to semantic integration.

generates a response using its own ontology O_S, and sends the response as a sequence of statements (e.g., describing the offered books including their price). Agent R integrates the statements into its ontology by successively applying the ontology-revision operator and resolves conflicts that occur due to the difference between O_R and O_S, thereby, e.g., discovering that the term *cheap* has a broader meaning in O_S than in O_R. The receiver can choose to stick to its initial reading of the common term *cheap* or to use the broader reading in the following communication. In the latter case we will say that the terminology of the resulting ontology is adapted to the terminology of the sender.

The meaning of the terms used in communication is based on the ontologies the participating agents hold [19]. For an agent whose ontology is consistent and well-tried, the treatment of terminological conflicts observed in communication should not lead to the loss of (parts of) the initial ontology. Thus, in adapting to the terminology of the book-selling agent, the customer agent needs not give up its own cheap-concept and the relations this concept has to other concepts within the ontology, even though it has learned that the book-selling agent uses a different reading of the term *cheap*. For this reason, belief-revision operators based on the AGM postulates [1] or belief-update operators according to [12] are not directly applicable.

While the loss of (parts of) the initial ontology is not acceptable, the initial ontology cannot (in its initial form) be part of the resulting ontology either, if the resulting ontology contains the conflicting statement (in its initial form). However, the initial ontology can be preserved by shifting it to a slightly different name space within the resulting ontology yielding a semantic mapping. The meaning the initial ontology assigns to the ambiguous term can be preserved by representing it using a new symbol in the resulting ontology, which assigns a new meaning to the common term in accordance with axioms specifying the semantic interrelation between the two readings. The semantic mapping maps the common vocabulary to the symbols representing their initial readings within the resulting ontology. The ontology-revision operator defined along this line is a non-monotonic belief-change operator, since the initial ontology is not preserved in its initial form.

We will present (Section 4) a collection of ontology-revision operators that differ regarding their adaptability to the terminology of the trigger and regarding the hypotheses on the semantic relatedness between the two readings they employ. The operators are evaluated based on several criteria, including the criteria developed for belief revision (Section 5). This evaluation will show the similarities between belief revision and ontology revision as well as the differences. Furthermore, we will show in which sense and to which extent the ontology-revision operators in the case of integrating conflicting statements yield conservative extensions of the initial ontology, i.e. purely terminological changes (Section 6).

Iterated ontology revision results in a sequence of ontologies starting from an initial ontology. If the sequence of triggers to be integrated is consistent, then the resulting ontologies should include more and more of the information from the sequence. If non-monotone belief-change operators are used to integrate

consistent sequences of statements into a knowledge base, statements from the beginning of a sequence need not be contained in the resulting knowledge base after integrating the whole sequence. In addition, within a longer process of integrating statements using an ontology-revision operator, the meaning of a common term can shift more than once, if the hypotheses on the semantic relatedness prove too strong.

The concepts of convergence (for sequences based on infinite sets of triggers) [24] or stability (for sequences based on finite sets of triggers) [13] allow one to formalize that the sequence of statements is integrated on the long run. If a belief-change operator can guarantee that for every initial ontology and every consistent sequence with finitely many trigger statements the sequence of ontologies will become constant at some point, then the operator can be called *stable*. We use the concept of stability to investigate the iterated application of the ontology-revision operators rather than the concept of convergence, since in the setting of communicating agents only finite sets of statements will have to be integrated. Following the investigations on iterated belief revision in the context of learning theory [13], [14], [16], [24], we will investigate how ontology-revision operators behave in iterated application (Section 7). The investigations on iterated ontology revision integrating sequences of statements show the consequences of adapting the terminology based on conflicts and can form the basis for considering iterated ontology revision where in each step an ontology is integrated.

2 Related Work

Ontology revision based on reinterpretation is an approach for integrating conflicting information into an ontology. The initial ontology is recognizable within the resulting ontology due to a semantic mapping between the common terms and symbols representing the reading assigned to the term by the initial ontology. Correspondingly, ontology revision is closely related to belief change and approaches to ontology integration that treat conflicts based on ambiguity.

Alchourrón, Gärdenfors and Makinson (AGM) [1] present a systematic and formal treatment of belief-change operators. Belief-change operators are meant to model the human ability to change beliefs in the presence of trustworthy information conflicting with the current beliefs. AGM [1] consider settings in which the conflict is based on false information and specify principles guiding the (rational) revision of beliefs by rationality postulates. Beliefs are characterized by belief sets, which are sets of formulae that are closed with respect to a logical consequence operator. Belief change is formalized in terms of binary belief-revision functions that map a belief set and a formula (the information triggering the change) onto a new belief set. The rationality postulates are axiom-like specifications of belief-revision functions. Additionally, AGM provide specific belief-revision functions that fulfill the rationality postulates.

Post-AGM work on belief revision considers also the revision of (finite) belief bases, allowing the definition of belief-revision operators based on a finite

representation of the beliefs of an agent. Approaches to belief-base revision in the sense of [11] define belief-revision functions for which the revision results depends on the syntactic structure of the belief bases. Approaches on the revision of knowledge bases ([3], [5]) define syntax-independent operators that are defined on and result in finite representations of belief sets.

In the intended settings of belief revision, conflicts are caused by false information and not by terminological mismatch. Since erasing wrong beliefs is a rational treatment of conflicts, belief revision is concerned with identifying formulae that should not be preserved rather than with the question of how all initial beliefs can be preserved. Nevertheless, the approach to ontology revision discussed below employs techniques developed in the context of belief revision.

Based on the work of Meyer, Lee and Booth [17] on knowledge integration strategies for stratified description-logic knowledge bases, Qi, Liu and Bell [22] define revision operators for description-logic knowledge bases. The two knowledge bases combined have different roles regarding the revision operator. The knowledge base to be integrated will be called the *trigger knowledge base* and the other knowledge base will be called the *initial knowledge base* in the following. Qi, Liu and Bell show that their operators fulfill the main conditions specified for rational belief-revision functions by AGM. But as their operators are applied to finite knowledge bases rather than belief sets, the syntactic structure of the knowledge base can have an effect on the resulting knowledge base. In order to minimize the effect of the syntactic form and to carry over as much information from the initial knowledge base as possible, axioms of the initial knowledge base responsible for a conflict are replaced by weaker axioms that do not yield a conflict, i.e., an axiom β is replaced by an (certain) axiom β_w, which is a consequence of β. The main components of the weakened axioms are exception lists, which can be specified in certain description logics. For example, axioms of the form 'All Cs are Ds' are weakened to axioms of the form 'All Cs except $a_1, \ldots, a_n$ are Ds'. The revision operators of [22] are successful in the sense that the trigger knowledge base is included in the resulting knowledge base, but the initial knowledge base need not be preserved.

The consistency-based approach of Delgrande and Schaub [5] for the revision of propositional belief sets uses language extensions similar to the ontology-revision operators defined in Section 4. The input knowledge base (O) and trigger statement (α) of the belief-change operators are formulated based on a common vocabulary $\mathcal{V}_c$. In the case of conflict, a disjoint vocabulary $\mathcal{V}_p$ and a substitution $(\cdot')$ mapping $\mathcal{V}_c$ to $\mathcal{V}_p$ is introduced, such that every propositional letter $p \in \mathcal{V}_c$ is mapped to a $p' \in \mathcal{V}_p$. Applying this substitution to O results in a renamed variant O'. To semantically interconnect the two sets of symbols and thereby restore propositions from O formulated with $\mathcal{V}_c$, bi-implications of the type $p \leftrightarrow p', p \in \mathcal{V}_c$ are added to O'. The definition of the belief-change operators select (inclusion) maximal sets of such bi-implications (EQ) consistent with the union of the trigger statement and O'. To dispose of the propositions using $\mathcal{V}_p$ while retaining as much content formulated in $\mathcal{V}_c$ as possible, the resulting set $O' \cup \{\alpha\} \cup EQ$ is deductively closed and intersected with $\mathcal{L}(\mathcal{V}_c)$, the propositional language using propositional letters from $\mathcal{V}_c$. Thus, the bi-

implications and the propositions using $\mathcal{V}_p$ are auxiliary means for the revision step, but do not occur in the revision result.

Delgrande and Schaub [5] define two belief-revision operators, a choice-revision operator $\dot{+}_c$, which is based on selecting one maximal set of bi-implications, and a skeptical-revision operator $\dot{+}$. The revision results $O\dot{+}_c\alpha$, $O\dot{+}\alpha$ are knowledge bases over the common vocabulary that do not preserve the initial belief set O. Furthermore, the operators are defined for propositional logic and are not directly applicable to more expressive logical frameworks required to model terminological changes in ontologies.

The permanent extension of the used vocabulary and a terminological shift is the result of the proposal to integrate conflicting description-logic ontologies by Goeb and colleagues [9]. They specify an algorithm that takes as input two ontologies O_R and O_S and yield a new ontology and two semantic mappings that map the symbols of the initial ontologies to the symbols of the resulting ontology. The common vocabulary of O_R and O_S is preserved in the resulting ontology, but the three ontologies can assign three different readings to the symbols of the common vocabulary.

Inconsistencies between O_R and O_S are resolved by applying two substitutions σ_R, σ_S replacing common terms (c) by different symbols ($c\sigma_R, c\sigma_S$) in the two ontologies, yielding compatible ontologies $O\sigma_R$ and $O\sigma_S$. The common symbol (c) is added as a common super-concept (or superordinate role symbol) to the two new symbols. In an additional step, individual axioms of $O\sigma_R$ or $O\sigma_S$ using $c\sigma_R$ or $c\sigma_S$ for which the replacement of the new symbol by the common symbol is compatible with the intermediate ontology are replaced by the version using the common symbol. Consequently, the common terms of the resulting ontology neither represent the receiver's nor the sender's reading. Furthermore, it is not guaranteed that the initial ontologies are (homogeneously) preserved in the sense that $O\sigma_R$ or $O\sigma_S$ are consequences of the integration result [21].

A systematic evaluation of belief-revision functions with respect to incremental integration of consistent sequences of trigger statements is studied by Kelly [13], [14], Martin and Osherson [16], and Zhang and Foo [24]. They investigate the types of belief-revision functions and restrictions on sequences of trigger statements that lead to sequences of epistemic states stabilizing at or converging to a state corresponding to a model of the trigger statements. The goal is to identify belief-revision operators that provide reliable learning methods despite the fact that belief revision is directed at retaining as much of the original beliefs as possible, while learning requires changing beliefs to minimize the difference between the belief set and the facts describing the world. Learning methods are called *reliable* if the incremental integration of a consistent trigger sequences leads to sequences of belief sets that stabilize or converge.

Kelly [13], [14] studied a collection of belief-revision functions with regards to their behavior in iterated application and identified stable operators as well as operators that do not guarantee stable behavior. A result of Zhang and Foo [24] is that revision operators for epistemic states that are not extremely skeptical regarding new statements will converge to a complete knowledge state identifying a model of the trigger statements. We will show that some ontology-

revision operators that adapt to the terminology of the trigger sequences lead to stabilizing sequences of ontologies that include the statements of the trigger sequence. However, ontology-revision operators that use stronger hypotheses regarding the semantic relations between the different readings of the common term can, depending on the structure of the trigger sequence, lead to closer approximations of the source ontology or lead to non-stabilizing integration processes.

3 Basic Definitions: Description Logic and Ontologies

Throughout this article an ontology will be a finite set of formulae over a description-logic (DL) language. Ontologies will be denoted by O and indexed or primed variants. For arbitrary formulae we will use β and indexed or primed variants. A (DL) vocabulary $\mathcal{V}$ includes concept symbols $(K, K', K_i, \dots)$, role symbols (R, R', R_i), and constants $(a, b, c, a', a_i, \dots)$. An *ontology over a vocabulary* $\mathcal{V}$ is a finite set of formulae in which all non-logical symbols are in $\mathcal{V}$. $\mathcal{V}(O)$ is the set of the non-logical symbols occurring in O. For $\mathcal{V}(\{\beta\})$ we write $\mathcal{V}(\beta)$. If $\mathcal{V}$ is a vocabulary, then $\mathcal{L}(\mathcal{V})$ will be used for the set of all formulae that can be formulated over $\mathcal{V}$. $O_{[K/K']}$ is the outcome of uniformly replacing the concept symbol K by K' in O. An interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ of the vocabulary $\mathcal{V}$ is a pair consisting of the nonempty domain $\Delta^{\mathcal{I}}$ and a function $\cdot^{\mathcal{I}}$ assigning to every constant $a \in \mathcal{V}$ an element $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$, to every concept symbol $K \in \mathcal{V}$ a set $K^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$, and to every role symbol $R \in \mathcal{V}$ a relation $R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$.

Concept descriptions $(C, D, C', C_i, \dots)$ are formed based on a vocabulary and concept constructors. The concept constructors used in this article are listed in Table 1.[2] The ontology-revision operators discussed in the following can be represented in the description logic $\mathcal{ALU}$. $\mathcal{ALU}$ is the DL language that adds the concept constructor for concept union (disjunction) to the language $\mathcal{AL}$, which employs the basic inventory of intersection (conjunction), atomic negation, value restriction and limited existential quantification. The examples and discussions use additional concept constructors for general negation, (unqualified) number restriction, and nominals.

Description-logic formulae can be classified as TBox axioms (terminological knowledge) and ABox axioms (world description). In this article, we will use general concept inclusions (GCI), i.e., formulae of the form $C \sqsubseteq D$, and concept equivalences, i.e., formulae of the form $C \doteq D$, as TBox axioms. ABox axioms can have the form $C(a)$ or $R(a, b)$. ABox axioms of the form $K(a)$ (for a concept symbol K) are called *positive literals*, ABox axioms of the form $\neg K(a)$ are *negative literals*, and the union of these sets of formulae is named *literals*. Literals are denoted by α, γ and indexed or primed variants. Finite sequences of literals (A) will be represented as $A = \langle \alpha_1, \alpha_2, \dots, \alpha_n \rangle = \langle \alpha_i \rangle_{i \in \{1,\dots,n\}}$, infinite sequences of literals as $A = \langle \alpha_n \rangle_{n \in \mathbb{N}}$. The set of elements occurring in a sequence

[2] For more details regarding the definitions and the syntax of description logics see [2].

Name	Syntax	Semantics
Top concept	$\top$	$\Delta^{\mathcal{I}}$
Bottom concept	$\bot$	$\emptyset$
Intersection	$C \sqcap D$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
Union	$C \sqcup D$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$
Atomic negation	$\neg K$	$\Delta^{\mathcal{I}} \setminus K^{\mathcal{I}}$
General negation	$\neg C$	$\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$
Value restriction	$\forall R.C$	$\{x \in \Delta^{\mathcal{I}} \mid \{y \in \Delta^{\mathcal{I}} \mid (x,y) \in R^{\mathcal{I}}\} \subseteq C^{\mathcal{I}}\}$
Limited exist. quantification	$\exists R.\top$	$\{x \in \Delta^{\mathcal{I}} \mid \{y \in \Delta^{\mathcal{I}} \mid (x,y) \in R^{\mathcal{I}}\} \neq \emptyset\}$
Unqual. number restriction	$\leq nR$	$\{x \in \Delta^{\mathcal{I}} \mid \lvert\{y \in \Delta^{\mathcal{I}} \mid (x,y) \in R^{\mathcal{I}}\}\rvert \leq n\}$
Unqual. number restriction	$\geq nR$	$\{x \in \Delta^{\mathcal{I}} \mid \lvert\{y \in \Delta^{\mathcal{I}} \mid (x,y) \in R^{\mathcal{I}}\}\rvert \geq n\}$
Nominals	$\{a\}$	$\{a^{\mathcal{I}}\}$

Table 1: DL syntax and semantics for $\mathcal{ALCUNO}$

A is denoted by $\tilde{A}$. If A is a sequence of length n and $i \leq n$ or A is an infinite sequence, then A^i is the prefix of A of length i.

The semantics of TBox axioms and ABox axioms is given by: $\mathcal{I} \models C \sqsubseteq D$ if and only if (iff) $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$; $\mathcal{I} \models C \doteq D$ iff $C^{\mathcal{I}} = D^{\mathcal{I}}$; $\mathcal{I} \models C(a)$ iff $a^{\mathcal{I}} \in C^{\mathcal{I}}$; $\mathcal{I} \models R(a,b)$ iff $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$. If for some formula β we have $\mathcal{I} \models \beta$, then we say that $\mathcal{I}$ *makes* β *true* or $\mathcal{I}$ *is a model of* β. Two formulae are *equivalent* ($\beta \equiv \beta'$) iff they have the same models. An interpretation $\mathcal{I}$ is a *model* of a set of formulae M ($\mathcal{I} \models M$) iff it is a model of every formula of M. A set of formulae is *consistent* iff it has a model, otherwise, M is *inconsistent* ($M \models \bot$). A sequence of literals A is *consistent* iff $\tilde{A}$ is consistent. $\mathsf{Mod}(M)$ is the set of models of M. Two sets of formulae M_1 and M_2 are *equivalent* ($M_1 \equiv M_2$) iff they have the same models ($\mathsf{Mod}(M_1) = \mathsf{Mod}(M_2)$). A formula β is a *consequence* of a set of formulae M ($M \models \beta$) iff every model of M is a model of β, and a set of formulae M' is a *consequence* of M ($M \models M'$) iff $\mathsf{Mod}(M) \subseteq \mathsf{Mod}(M')$. We will also use $\models \beta$ to stand for $\emptyset \models \beta$.

The formal evaluation of the ontology-revision operators will be based on an extended logical language combining description logic and propositional logic. In this language, the description-logic formulae (TBox axioms and ABox axioms) play the role of atomic formulae. Additionally, we use equalities ($a \doteq b$) and inequalities ($a \not\doteq b$) (with the semantics $\mathcal{I} \models (a \doteq b)$ iff $a^{\mathcal{I}} = b^{\mathcal{I}}$ and $\mathcal{I} \models (a \not\doteq b)$ iff $a^{\mathcal{I}} \neq b^{\mathcal{I}}$) as atomic formulae to analyze and express the role of unique name assumptions. However, equalities and inequalities are not assumed to be contained in the sequences of literals representing the trigger statements. The atomic formulae can be combined in the style of propositional logic. Thus, β could be $\neg R(a,b) \vee (R(b,a) \wedge R(a,a))$. Because of their close semantic relatedness, we will not distinguish the symbols for propositional negation and concept negation.

If M is a set of literals over vocabulary $\mathcal{V}$, then the set of inequalities ex-

pressing the *unique name assumption implicit in* M is $\mathsf{una}(M) = \{(a \neq b) \mid K(a), \neg K(b) \in M, \text{ for } K, a, b \in \mathcal{V}\}$. If A is a sequence of literals, then we will write $\mathsf{una}(A)$ instead of $\mathsf{una}(\tilde{A})$. Note that any set of literals M is inconsistent if and only if there is a concept symbol K and a constant a, such that $\{K(a), \neg K(a)\} \subseteq M$. Thus, a set of literals is inconsistent if and only if its implicit unique name assumption is inconsistent.

The (global) strong ontology-revision operators of [20] are defined with reference to the most specific concept assigned by an ontology to a constant. C is *a most specific concept for* a *in the ontology* O iff $O \models C(a)$ and for all C' such that $O \models C'(a)$ also $\models C \sqsubseteq C'$. The existence of a finite representation of a most specific concept depends on the ontology O and the underlying description logic.[3] We assume that there is some systematic way (e.g. an ordering over concept descriptions) to pick out for every constant a a representative of the most specific concept in an ontology O. This representative will be denoted by $\mathsf{msc}_O(a)$.

4 Ontology-Revision Operators: Definitions

Ontology-revision operators are binary operators that map an ontology (O) and a literal (α) to another ontology ($O \circ \alpha$) that represents the integrated information of O and α. As we define ontologies as finite sets of formulae, the ontology-revision operators take as first arguments sets of formulae that are not deductively closed. In this respect our ontology-revision operators are comparable with belief-revision functions operating on belief bases [11] and not on belief sets, which are defined as deductively closed sets of formulae [1]. A formula that occurs as the second argument of an ontology-revision operator is called a *trigger statement* or just *trigger*.

If the trigger α is compatible with O, then it can be added to O to derive the new ontology $O \cup \{\alpha\}$. Correspondingly, the case that $O \cup \{\alpha\}$ is consistent is handled by all belief-change operators and the ontology-revision operators defined in this section in this way. If the trigger α is not compatible with ontology O, then the ontology-revision operators based on reinterpretation defined and analyzed in this article capture the assumption that the incompatibility is caused by an ambiguity in the common vocabulary. If a conflict between O and α derives from an underlying ambiguity of a common term (K) and the goal is to represent both readings in the resulting ontology, then this goal can be achieved by enriching the terminology. Correspondingly, two readings of a term used in O and in α are distinguished in the resulting ontology and a new symbol (K') is introduced to represent one of the readings.

To systematically distinguish between the common symbols and the new symbols, we will assume in the following that the vocabulary used by the receiver can be partitioned into a common vocabulary ($\mathcal{V}_c$) and an internal or private vocabulary ($\mathcal{V}_p$), such that the symbols of $\mathcal{V}_p$ cannot be used by the sender.

[3] [15] describes a family of description logics for which the most specific concept exists and an algorithm for determining the most specific concept.

The symbols introduced in the reinterpretation process are private symbols. We assume that $\mathcal{V}_p$ provides infinitely many symbols not used in the receiver's ontology O. We additionally assume in the following definitions that the choice of a symbol $K' \in \mathcal{V}_p$ in the reinterpretation step is uniquely determined by the symbol K and $\mathcal{V}_p \setminus \mathcal{V}(O)$.

The strong ontology-revision operators $\odot_1$ and $\odot_2$ (Definition 1) differ regarding which reading (the reading represented in O vs. the reading underlying α) is denoted by the new symbol K' after integrating the trigger statement. The type-1 operator $\odot_1$ uses the new symbol to represent the reading underlying the trigger statement and continues to represent the reading specified by the initial ontology by the common symbol. Correspondingly, it does not add the trigger statement α but $\alpha_{[K/K']}$ to the ontology. The type-2 operator $\odot_2$ internalizes the reading assigned by the initial ontology to the common symbol by replacing every occurrence of the common term in the initial ontology with the new symbol. The common symbol is used to represent the reading underlying the trigger statement. Correspondingly, the type-2 operator adds the trigger statement in its initial form to the modified initial ontology. Regarding the common term K, $\odot_1$ preserves the terminology of the ontology O while $\odot_2$ adapts to the terminology of the trigger α. The operators $\odot_1$ and $\odot_2$ are structurally similar in the sense that one operator can be derived from the other by a simple syntactic transformation. In the case of inconsistency, $(O \odot_2 K(a)) = (O \odot_1 K(a))_{[K/K',K'/K]}$ and $(O \odot_1 K(a)) = (O \odot_2 K(a))_{[K/K',K'/K]}$.

If a common term is ambiguous and the different readings are not semantically related, then the statement of the sender does not have any value for the receiver. Unfortunately, the trigger statement does not specify the reading of the common symbols it uses. However, hypotheses regarding the relation of the alternative reading to the terminology specified in the ontology can be added (and in later steps revised, if necessary). Therefore, both operators declare upper and lower bounds for the reading underlying the trigger statement. They implement the assumption that one of the readings of the ambiguous terms is more general than the other (as expressed by $K \sqsubseteq K'$ or $K' \sqsubseteq K$). The observed conflict gives evidence as to which subsumption relation has to be excluded.

The strong operators add additional bounds to minimize the semantic difference between the two readings. They exploit the ontology regarding its specification of the constant that is involved in the conflict. This constant denotes the only known entity that is a witness for the ambiguity of the term and for the difference between the two readings. The second bound based on $\mathsf{msc}_O(a)$ stands for the hypothesis that only objects that are similar to this witness will be further examples of the difference.

Definition 1. *Let O be an ontology over the vocabulary $\mathcal{V}_c \cup \mathcal{V}_p$ with $\mathcal{V}_c \cap \mathcal{V}_p = \emptyset$, $K \in \mathcal{V}_c$ a concept symbol and $a \in \mathcal{V}_c$ a constant for which $\mathsf{msc}_O(a)$ exists. Let $K' \in \mathcal{V}_p$ be a concept symbol not used in O. Then the* strong ontology-revision

operators of type 1 and 2 *($\odot_1$ and $\odot_2$) are defined for literals by*

$$O \odot_1 K(a) = \begin{cases} O \cup \{K(a)\} & \text{if } O \cup \{K(a)\} \text{ is consistent,} \\ O \cup \{K'(a), K \sqsubseteq K', K' \sqsubseteq K \sqcup \mathsf{msc}_O(a)\} & \text{else} \end{cases}$$

$$O \odot_1 \neg K(a) = \begin{cases} O \cup \{\neg K(a)\} & \text{if } O \cup \{\neg K(a)\} \text{ is consistent,} \\ O \cup \{\neg K'(a), K' \sqsubseteq K, K \sqsubseteq K' \sqcup \mathsf{msc}_O(a)\} & \text{else} \end{cases}$$

$$O \odot_2 K(a) = \begin{cases} O \cup \{K(a)\} & \text{if } O \cup \{K(a)\} \text{ is consistent,} \\ O_{[K/K']} \cup \{K(a), K' \sqsubseteq K, K \sqsubseteq K' \sqcup \mathsf{msc}_{O_{[K/K']}}(a)\} & \text{else} \end{cases}$$

$$O \odot_2 \neg K(a) = \begin{cases} O \cup \{\neg K(a)\} & \text{if } O \cup \{\neg K(a)\} \text{ is consistent,} \\ O_{[K/K']} \cup \{\neg K(a), K \sqsubseteq K', K' \sqsubseteq K \sqcup \mathsf{msc}_{O_{[K/K']}}(a)\} & \text{else} \end{cases}$$

Following Grove's idea of so called sphere-based belief revision developed in [10], Wassermann/Fermé [23] defined operations for revising a concept complex[4] by a new piece of information, resulting in a new concept. These ideas were adapted in [20] to define two types of ontology-revision operators in a local and a global variant respectively. The *strong operators* correspond to the global variants.[5] They are called *strong*, because we will define and discuss weaker ontology-revision operators in the following. The local operators are defined with respect to a system of spheres for the concept symbol K, which represent suitable generalizations of K. The use of the most specific concept in the specification of the global operators results as a common generalization of the local operators [20].

The bounds referring to the most specific concept are comparable with the weakenings of Qi, Liu and Bell [22]. For example, the inclusion axiom $K \sqsubseteq K' \sqcup \mathsf{msc}_{O_{[K/K']}}(a)$ occurring in the definition of $O \odot_2 K(a)$ is equivalent to $K \sqcap \neg \mathsf{msc}_{O_{[K/K']}}(a) \sqsubseteq K'$. This formula says that all individuals that instantiate K but do not instantiate the most specific concept of a also instantiate K'. In description logics that provide the concept constructor for nominals, the most specific concept of a in O can (in the context of O) be represented by $\{a\}$. Thus we have $K \sqcap \neg\{a\} \sqsubseteq K'$, which says that all K except a are K'. Such axioms correspond to the form of weakened axioms used in the definitions of the operators $\circ_w, \circ_{rw}$ in [22].

To analyze the difference between the two types of ontology-revision operators, we will investigate iterated applications of the operators based on sequences of trigger statements. This will lead to a more formal explication of the informal notion of *adaptation to the terminology of the trigger*. Since the difference between the type-1 operator and the type-2 operator does not crucially depend

[4] A concept complex according to [23] comprises a concept and descriptions for its prototypical instances.

[5] cf. [20], p. 87. A limitation of the definitions for $\odot_1$ and $\odot_2$ in [20] is that they are restricted to positive literals as trigger statements. To generalize the applicability of the ontology-revision operators, Definition 1 extends the definitions of the operators to deal also with negative literals as triggers. The extension of the definitions to other types of trigger statements needs to handle more than one candidate for reinterpretation and is developed in [21].

on the bound employing the most specific concept, we broaden the discussion to a range of operator pairs.

Definition 2 specifies the weak ontology-revision operators $\otimes_1$ and $\otimes_2$, which does not refer to most specific concepts. Furthermore, it does not employ the constructor for concept union ($\sqcup$) and yields TBox axioms that can be embedded in definitorial TBoxes. The weak ontology-revision operators can therefore be used in the context of any description-logic system that is capable of expressing the trigger sequence and handle definitions.

Definition 2. *Let O be an ontology over the vocabulary $\mathcal{V}_c \cup \mathcal{V}_p$ with $\mathcal{V}_c \cap \mathcal{V}_p = \emptyset$, $K \in \mathcal{V}_c$ a concept symbol and $a \in \mathcal{V}_c$ a constant. Let $K' \in \mathcal{V}_p$ be a new concept symbol. Then the* weak ontology-revision operators of type 1 and 2 *($\otimes_1$ and $\otimes_2$) are defined (for literals) by*

$$
\begin{array}{rcl}
O \otimes_1 K(a) & = & \begin{cases} O \cup \{K(a)\} & \text{if } O \cup \{K(a)\} \text{ is consistent,} \\ O \cup \{K'(a), K \sqsubseteq K'\} & \text{else} \end{cases} \\
O \otimes_1 \neg K(a) & = & \begin{cases} O \cup \{\neg K(a)\} & \text{if } O \cup \{\neg K(a)\} \text{ is consistent,} \\ O \cup \{\neg K'(a), K' \sqsubseteq K\} & \text{else} \end{cases} \\
O \otimes_2 K(a) & = & \begin{cases} O \cup \{K(a)\} & \text{if } O \cup \{K(a)\} \text{ is consistent,} \\ O_{[K/K']} \cup \{K(a), K' \sqsubseteq K\} & \text{else} \end{cases} \\
O \otimes_2 \neg K(a) & = & \begin{cases} O \cup \{\neg K(a)\} & \text{if } O \cup \{\neg K(a)\} \text{ is consistent,} \\ O_{[K/K']} \cup \{\neg K(a), K \sqsubseteq K'\} & \text{else} \end{cases}
\end{array}
$$

We demonstrate the effects of the ontology-revision operators with an example of simple ontologies in a book-trading scenario.

Example 1. *An agent (receiver) using an ontology O_R wants to buy a cheap book on thermodynamics in an online bookshop (sender) that uses the ontology O_S.*

According to O_R something is cheap if and only if it costs less than 5 Euros. That a book cannot have a soft cover and a hard cover at the same time is captured by the axiom $SoftC \sqsubseteq \neg HardC$ in O_R. O_R also specifies that anything that costs less than 5 Euros costs less than 8 Euros. Furthermore, the receiver has some knowledge about four books on thermodynamics. Book th_1 costs between 5 and 8 Euros, the books th_3 and th_4 cost more than 5 Euros, while book th_2 costs less than 5 Euros. th_1 is a hardcover book, th_4 is a softcover book, but for th_2 and th_3 the book type is not known to the receiver.

$$
\begin{array}{rcl}
O_R & = & \{Cheap \doteq CostLt\_5, \\
 & & SoftC \sqsubseteq \neg HardC, CostLt\_5 \sqsubseteq CostLt\_8, \\
 & & \neg CostLt\_5(th_1), CostLt\_8(th_1), HardC(th_1), \\
 & & CostLt\_5(th_2), \neg CostLt\_5(th_3), \neg CostLt\_5(th_4), SoftC(th_4)\}
\end{array}
$$

According to O_S something is cheap if and only if it costs less than 5 Euros or is a hardcover book and costs less than 8 Euros. Thus, O_S and O_R have different cheap-concepts. However, they agree regarding the axioms on book type

and ordering of prices. In the sender's ontology O_S, the book th_1 is specified to have a hard cover and to cost less than 8 Euros. Thus, book th_1 is cheap according to O_S.

To the request to list cheap books on thermodynamics available, the online bookshop answers (α) that the book named th_1 is cheap.

$$\begin{aligned} O_S &= \{Cheap \doteq CostLt\_\,5 \sqcup (HardC \sqcap CostLt\_\,8), \\ &\quad SoftC \sqsubseteq \neg HardC, CostLt\_\,5 \sqsubseteq CostLt\_\,8, \\ &\quad CostLt\_\,8(th_1), HardC(th_1)\} \\ \alpha &= Cheap(th_1) \end{aligned}$$

The different readings of Cheap lead to the inconsistency of $O_R \cup \{\alpha\}$. If the receiver decides to use the weak ontology-revision operator of type 2 for the integration of α, the outcome of the integration is

$$\begin{aligned} O_R \otimes_2 \alpha &= \{Cheap' \doteq CostLt\_\,5, \\ &\quad SoftC \sqsubseteq \neg HardC, CostLt\_\,5 \sqsubseteq CostLt\_\,8, \\ &\quad \neg CostLt\_\,5(th_1), CostLt\_\,8(th_1), HardC(th_1), \\ &\quad CostLt\_\,5(th_2), \neg CostLt\_\,5(th_3), \neg CostLt\_\,5(th_4), SoftC(th_4), \\ &\quad Cheap(th_1), Cheap' \sqsubseteq Cheap\} \end{aligned}$$

The operator $\otimes_2$ reconstructs the relation between the different readings using the subsumption $Cheap' \sqsubseteq Cheap$, which says the sender's cheap-concept, denoted by Cheap, is more general than the receiver's initial cheap-concept, denoted by $Cheap'$. A form of weakness of $\otimes_2$ is demonstrated by the fact that some consequences of the receiver's ontology that are not involved in a conflict are not preserved in their initial form. For example, while the consequence $Cheap(th_2)$ of O_R is a consequence of the revision result $O_R \otimes_2 \alpha$, the literals $\neg Cheap(th_3)$ and $\neg Cheap(th_4)$ are not consequences of $O_R \otimes_2 \alpha$. The asymmetry regarding the preservation of positive and negative literals that contain the concept symbol of the trigger statement will be systematically discussed in the following section (Propositions 3.2, 3.3).

If the receiver uses the strong type-2 operator for the integration, the result can be represented as

$$O_R \odot_2 \alpha \equiv O_R \otimes_2 \alpha \cup \{Cheap \sqsubseteq Cheap' \sqcup (\neg CostLt\_\,5 \sqcap CostLt\_\,8 \sqcap HardC)\}$$

In this case it is assumed that the sender calls something cheap only if it costs less than 5 Euros or costs between 5 and 8 Euros and is a hardcover book. For both O_S and $O_R \odot_2 \alpha$ the formula $Cheap \sqsubseteq CostLt\_\,5 \sqcup (CostLt\_\,8 \sqcap HardC)$ is a consequence. As in the case of the integration with $\otimes_2$, the literal $\neg Cheap(th_3)$ is not a consequence of $O_R \odot_2 \alpha$. However in this case, the literal $\neg Cheap(th_4)$ is preserved, i.e., $O_R \odot_2 \alpha \models \neg Cheap(th_4)$. The difference derives from th_4 being known to be a softcover book and, thus, not similar to th_1, the only known witness of the difference between the two readings of Cheap, whereas the type of book is not known for th_2 and th_3 (see Proposition 3.4).

The scheme underlying the definitions of the ontology-revision operators $\odot_i$ and $\otimes_i$ for ($i \in \{1,2\}$) can be generalized to the definition of a family of operators that differ regarding the specification of the second bound for the new concept. Let sel stand for a selection function that selects upper-bound axioms for the new concept. As possible upper bounds we take concepts that are both more general than the lower bound and will not add new information to the known instance a. Then the selection function sel can be a parameter in the specification of the operators $\oplus_i^{\mathsf{sel}}$.

Definition 3. *Let O be an ontology over the vocabulary $\mathcal{V}_c \cup \mathcal{V}_p$ with $\mathcal{V}_c \cap \mathcal{V}_p = \emptyset$, $K \in \mathcal{V}_c$ a concept symbol, $a \in \mathcal{V}_c$ a constant, and $\alpha = K(a)$ or $\alpha = \neg K(a)$. Let $i \in \{1,2\}$ and $K' \in \mathcal{V}_p$ be the new concept symbol introduced to form $O \otimes_i \alpha$. Then the* upper-bound axioms based on O for α and K' *($\mathsf{ub}_i(O, \alpha, K')$) and the* ontology-revision operators of type 1 and 2 based on sel *($\oplus_1^{\mathsf{sel}}$ and $\oplus_2^{\mathsf{sel}}$) are defined (for literals) by*[6]

$$
\begin{array}{rcl}
\mathsf{ub}_1(O, K(a), K') & = & \{K' \sqsubseteq K \sqcup C \mid O \models C(a)\} \\
\mathsf{ub}_1(O, \neg K(a), K') & = & \{K \sqsubseteq K' \sqcup C \mid O \models C(a)\} \\
\mathsf{ub}_2(O, K(a), K') & = & \{K \sqsubseteq K' \sqcup C \mid O_{[K/K']} \models C(a)\} \\
\mathsf{ub}_2(O, \neg K(a), K') & = & \{K' \sqsubseteq K \sqcup C \mid O_{[K/K']} \models C(a)\} \\
O \oplus_i^{\mathsf{sel}} \alpha & = & \begin{cases} O \cup \{\alpha\} & \text{if } O \cup \{\alpha\} \text{ is consistent,} \\ O \otimes_i \alpha \cup \mathsf{sel}(\mathsf{ub}_i(O, \alpha, K')) & \text{else} \end{cases}
\end{array}
$$

If the selection function sel selects the empty set or another set of tautological formulae, one gets an operator with the same semantic effect as the weak operator $\otimes_i$. If sel returns the complete set or any set that contains the upper bound derived from $\mathsf{msc}_O(a)$, one gets an operator with the same semantic effect as the strong operator $\odot_i$.

One simple form of selection is to (syntactically) evaluate the assertions of the ontology. This idea is captured in the definition of the revision operators $\oplus_i^{\mathsf{expl}}$. The proof of the instability of the stronger ontology-revision operators is based on $\oplus_2^{\mathsf{expl}}$ and Example 5 in Appendix II demonstrates the use of $\oplus_2^{\mathsf{expl}}$. Computing the additional axioms on this basis is rather simple. Other forms of restricting the additional bounds can set restrictions on the syntactic complexity of the bounds (such as restrictions on the embedding of quantifiers).

Definition 4. *Let O be an ontology over the vocabulary $\mathcal{V}_c \cup \mathcal{V}_p$ with $\mathcal{V}_c \cap \mathcal{V}_p = \emptyset$, $K \in \mathcal{V}_c$ a concept symbol, $a \in \mathcal{V}_c$ a constant, and $\alpha = K(a)$ or $\alpha = \neg K(a)$. Let $i \in \{1,2\}$ and $K' \in \mathcal{V}_p$ be the new concept symbol introduced to form $O \otimes_i \alpha$.*

Then the functions $\mathsf{ub}_{i,\mathsf{expl}}$ deriving upper-bound axioms based on explicit concept assertions *and the* ontology-revision operator based on explicit concept

[6] If upper-bound axioms are selected for the type-1 operator in the positive case from $\{K' \sqsubseteq C \mid O \models C(a)$ and $O \models (K \sqsubseteq C)\}$ (and in a corresponding way in the other cases), then the constructor for concept union is not needed. However, in this case the inclusion property Observation 1.1, p. 15, can be violated.

assertions ($\oplus_i^{\mathsf{expl}}$) *are given by*

$$
\begin{array}{rcl}
\mathsf{ub}_{1,\mathsf{expl}}(O, K(a), K') & = & \{K' \sqsubseteq K \sqcup C \mid C(a) \in O\} \\
\mathsf{ub}_{1,\mathsf{expl}}(O, \neg K(a), K') & = & \{K \sqsubseteq K' \sqcup C \mid C(a) \in O\} \\
\mathsf{ub}_{2,\mathsf{expl}}(O, K(a), K') & = & \{K \sqsubseteq K' \sqcup C \mid C(a) \in O_{[K/K']}\} \\
\mathsf{ub}_{2,\mathsf{expl}}(O, \neg K(a), K') & = & \{K' \sqsubseteq K \sqcup C \mid C(a) \in O_{[K/K']}\} \\
O \oplus_i^{\mathsf{expl}} \alpha & = & \begin{cases} O \cup \{\alpha\} & \text{if } O \cup \{\alpha\} \text{ is consistent,} \\ O \otimes_i \alpha \cup \mathsf{ub}_{i,\mathsf{expl}}(O, \alpha, K') & \text{else} \end{cases}
\end{array}
$$

Notation 1. *In the following, the symbol $\circ_1$ will be used as metavariable for ontology-revision operators of type 1, i.e., $\circ_1$ stands for $\odot_1$, $\oplus_1^{\mathsf{sel}}$, $\oplus_1^{\mathsf{expl}}$, or $\otimes_1$. Similarly, $\circ_2$ will be used as metavariable for ontology-revision operators of type 2, i.e., $\circ_2$ stands for $\odot_2$, $\oplus_2^{\mathsf{sel}}$, $\oplus_2^{\mathsf{expl}}$, or $\otimes_2$. The symbol $\circ$ will be used as metavariable for all defined ontology-revision operators. If $\circ$ is an ontology-revision operator, O an ontology, and $A = \langle \alpha_1, \alpha_2, \ldots, \alpha_n \rangle$ a finite sequence of literals, then $O \circ A =_{def} (\ldots(O \circ \alpha_1) \circ \alpha_2) \ldots) \circ \alpha_n$ is the outcome of iterated applications of the operator $\circ$ to the ontology O and the literals of the sequence A. The successive integration of the literals of any sequence A into an ontology O defines a sequence of ontologies. If $\circ$ is an ontology-revision operator, O an ontology, and $A = \langle \alpha_i \rangle_{i \in I}$ a sequence of literals, then the sequence of ontologies resulting from iteratively integrating A is $\langle O \circ A^i \rangle_{i \in I}$.*

5 Basic Properties of the Ontology-Revision Operators

The following observations directly result from the definitions of the operators.

Observation 1. *Let O, O_1, O_2 be ontologies over $\mathcal{V}_c \cup \mathcal{V}_p$ with $\mathcal{V}_c \cap \mathcal{V}_p = \emptyset$, $\mathcal{V}(O_1) \cap \mathcal{V}_p = \mathcal{V}(O_2) \cap \mathcal{V}_p$, α, γ be literals with $\mathcal{V}(\{\alpha, \gamma\}) \subseteq \mathcal{V}_c$, and A a finite sequence of literals over $\mathcal{V}_c$. Let $\circ$ be any ontology-revision operator, $\circ_1$ be an ontology-revision operator of type 1, $\circ_2$ be an ontology-revision operator of type 2, and $i \in \{1, 2\}$.*

1. $O \circ \alpha \cap \mathcal{L}(\mathcal{V}_c \cup \mathcal{V}(O)) \subseteq O \cup \{\alpha\}$ *(inclusion)*
2. $O \circ A \cap \mathcal{L}(\mathcal{V}_c \cup \mathcal{V}(O)) \subseteq O \cup \tilde{A}$ *(inclusion for iterated $\circ$)*
3. $O \circ \alpha = O \cup \{\alpha\}$ *iff* $O \cup \{\alpha\}$ *is consistent.* *(vacuity)*
4. $O \circ A = O \cup \tilde{A}$ *iff* $O \cup \tilde{A}$ *is consistent.* *(vacuity for iterated $\circ$)*
5. *If* $O_1 \equiv O_2$, *then* $(O_1 \otimes_i \alpha) \equiv (O_2 \otimes_i \alpha)$ *(left extensionality for weak operators)*
6. *If* $O_1 \equiv O_2$, *then* $(O_1 \odot_i \alpha) \equiv (O_2 \odot_i \alpha)$ *(left extensionality for strong operators)*

7. *If* $\alpha \equiv \gamma$, *then* $(O \circ \alpha) \equiv (O \circ \gamma)$ *(right extensionality)*

8. $\alpha \in O \circ_2 \alpha$ *(success for* $\circ_2$*)*

9. $O \subseteq O \circ_1 \alpha$ *(monotonicity for* $\circ_1$*)*

10. $O \subseteq O \circ_1 A$ *(monotonicity for iterated* $\circ_1$*)*

11. $O \odot_i \alpha \models O \circ_i \alpha$ *(strength of strong operators)*

12. $O \circ_i \alpha \models O \otimes_i \alpha$ *(strength of weak operators)*

13. $O \circ \alpha$ *is consistent iff* O *is consistent.* *(consistency)*

14. $O \circ A$ *is consistent iff* O *is consistent.* *(consistency for iterated* $\circ$*)*

Observations 1.1, 1.3, 1.5, 1.6, 1.7, and 1.8 are adapted variants of the basic AGM postulates.[7] Observation 1.1 (the part of $O \circ \alpha$ expressed in the common vocabulary and the vocabulary of the initial ontology is included in $O \cup \{\alpha\}$) is weaker than the inclusion postulate for belief bases ($O * \alpha \subseteq O \cup \{\alpha\}$, cf. [11], p. 200). Observation 1.13 is weaker than the corresponding AGM consistency postulate ($O * \alpha$ is inconsistent iff α is inconsistent) as inconsistencies in the ontology O can be preserved by the ontology-revision operators. Similarly, the belief-revision operators defined by Delgrande and Schaub [5] fulfill a weaker consistency postulate corresponding to 1.13 rather than the AGM version.

The slight weakenings of the inclusion and consistency postulates suggest that the ontology-revision operators are not rational belief-revision functions as defined by [1] and [8]. To pay tribute to the difference regarding inclusion and to the fact that type-1 operators do not fulfill success, we call the operators defined in Section 4 *ontology-revision* operators rather than *belief-revision* operators.

Ontology-revision operators take as first arguments finite sets of formulae. In this sense, they are comparable with revision functions for belief bases. But in contrast to operators for belief-base revision that fulfill the inclusion postulate for belief bases ($O * \alpha \subseteq O \cup \{\alpha\}$) weak and strong ontology-revision operators fulfill left extensionality (Observations 1.5, 1.6). Therefore, ontology-revision operators are comparable to revision operators for knowledge bases (like the operators defined by Dalal [3] or Delgrande and Schaub [5]), which fulfill left extensionality but do not fulfill the inclusion postulate for belief bases.

Since conflicts between the ontology O and the trigger statement α are resolved by the ontology-revision operators, the operators cannot be both monotone regarding O and successful regarding α. Type-1 operators are monotone (1.9) and not necessarily successful, while type-2 operators are successful (1.8) but not monotone. As type-1 operators do not fulfill the success postulate, they cannot be classified as belief-expansion operators (cf. [7], p. 49). While the observations 1.1, 1.3, 1.9, and 1.13 can be generalized to statement sequences

[7] Compare the formulation of the postulates for belief sets in [8], [11], and [12]. The two AGM postulates not adapted here deal with the revision of belief sets with complex formulae (conjunction). For proofs of some of the observations consult p. 28 in Appendix I.

(1.2, 1.4, 1.10, and 1.14), such a generalization is not possible for 1.8. This issue will be discussed in Section 7 in the context of the stability of the operators.

Proposition 1 states generalizations of Observation 1.8 and Observation 1.10 that are valid for both types of ontology-revision operators. The initial ontology O is preserved in the resulting ontology $O \circ A$ in the sense that $O\sigma$ is a subset of $O \circ A$ where σ is a substitution function. σ plays the role of a semantic mapping that maps a common symbol s to the symbol $s\sigma$ representing the reading of s relative to O in $O \circ A$. Furthermore, both O and A can be recovered from the integration result by applying another substitution ρ. The substitution ρ is a semantic mapping that maps axioms of $O \circ A$ that originate from O or A to their initial form.

Proposition 1. *For any ontology O over $\mathcal{V}_c \cup \mathcal{V}_p$ with $\mathcal{V}_c \cap \mathcal{V}_p = \emptyset$, finite sequence A of literals over $\mathcal{V}_c$, and ontology-revision operator $\circ$ there are two substitutions of concept symbols (σ and ρ), such that*

1. *$O\sigma \subseteq O \circ A$, and*
2. *$O \cup \tilde{A} \subseteq (O \circ A)\rho$.*

Proof. See p. 28.

6 Conservativity

Type-1 operators preserve the terminology of the initial ontology. Formally, preservation of the terminology means that terminological shifts result in conservative extensions. More specifically, type-1 operators yield conservative extensions of an ontology in cases of triggers that conflict with the ontology.

Definition 5. *A theory O' over vocabulary $\mathcal{V}(O')$ is called a* conservative extension of the theory O *over vocabulary $\mathcal{V}(O) \subseteq \mathcal{V}(O')$ iff for all formulae β in $\mathcal{V}(O)$: $O \models \beta$ iff $O' \models \beta$.*[8]

Proposition 2. *Let O be an ontology, α be a literal, and $\circ_1$ be an ontology-revision operator of type 1. If $O \cup \{\alpha\}$ is inconsistent, then $O \circ_1 \alpha$ is a conservative extension of O.*

Proof. See p. 29. As the receiver wants to enrich his knowledge, he is not generally interested in conservative extensions. If a trigger statement α is compatible with the ontology O, then the sender and the receiver have compatible readings of the common symbols. Correspondingly, if $O \cup \{\alpha\}$ is consistent and $O \not\models \alpha$, then the integration result $O \cup \{\alpha\}$ is not a conservative extension. As a consequence it is not the case that for all finite sequences A the ontology $O \circ_1 A$ is a conservative extension of O. Type-1 operators preserve the terminology of the initial ontology in the sense that the integration of sequences of statements is a combination of conservative extensions and accepting statements. Observation

[8] [18], p. 208.

2 states that for ontology-revision operators of type 1 the effect of integrating sequences of trigger statement can be decomposed into the simple addition of some statements from the sequence and a conservative extension.

Observation 2. *Let O be an ontology over the vocabulary $\mathcal{V}_c \cup \mathcal{V}_p$ with $\mathcal{V}_c \cap \mathcal{V}_p = \emptyset$, A be a finite sequence of statements over $\mathcal{V}_c$, and $\circ_1$ an ontology revision operator of type 1. Then there is a subset $\tilde{A}' \subseteq \tilde{A}$, such that $O \cup \tilde{A}' \subseteq O \circ_1 A$ and $O \circ_1 A$ is a conservative extension of $O \cup \tilde{A}'$.*

A restricted form of conservativity in the case of inconsistency can also be proved for operators of type 2. As type-2 operators adapt to the terminology of the sender, the main question for these operators in the context of conservativity is to describe the parts of the receiver's initial ontology that are preserved in the common language along the integration and the conditions under which the preservation holds.

Proposition 3 states a combination of success and restricted conservativity properties for the type-2 operators in the case of reinterpretation. It generalizes and formalizes the observations on preservation and asymmetry mentioned in Example 1. More precisely, Assertion 3.1 states conservativity for all formulae β that do not contain one of the concept symbols involved in the reinterpretation. Because type-2 operators are successful regarding the trigger statement, the trigger statement constitutes an exception to conservativity. Assertion 3.2 expresses restricted conservativity for those literals in which the reinterpreted symbol occurs with the same negation prefix (negation vs. no negation symbol) as in the trigger. It makes the entity mentioned by the trigger statement the only exception to conservativity for literals of this form. Regarding literals in which the reinterpreted symbol occurs with a complementary negation prefix, the strong and the weak operators differ. Assertion 3.3 expresses that $\otimes_2$ does not preserve any literal in which the reinterpreted symbol K occurs with a complementary negation prefix. In contrast to this, revision with the strong operator $\odot_2$ preserves such literals for those constants that are known to differ from a regarding some property (Assertion 3.4). Given the mixed results regarding the literals, for more complex formulae involving the reinterpreted symbol simple preservation conditions cannot be formulated.

Proposition 3. *Let O be an ontology over the vocabulary $\mathcal{V}_c \cup \mathcal{V}_p$ with $\mathcal{V}_c \cap \mathcal{V}_p = \emptyset$, $K \in \mathcal{V}_c$ be a concept symbol, and $a, c \in \mathcal{V}_c$ be constants, such that $\mathsf{msc}_O(a)$ exists. Let $\alpha = K(a)$ and $\gamma = K(c)$ or $\alpha = \neg K(a)$ and $\gamma = \neg K(c)$. Let $K' \in \mathcal{V}_p \setminus \mathcal{V}(O)$ be the new symbol introduced in $O \circ_2 \alpha$, and β be a formula with $\mathcal{V}(\beta) \subseteq (\mathcal{V}_c \cup \mathcal{V}(O)) \setminus \{K\}$.*

If $O \models \neg\alpha$, then

1. $O \circ_2 \alpha \models \beta$ *iff* $O \models \beta$
2. $O \circ_2 \alpha \models \gamma$ *iff* $O \cup \{a \neq c\} \models \gamma$
3. $O \otimes_2 \alpha \not\models \neg\gamma$
4. $O \odot_2 \alpha \models \neg\gamma$ *iff* $O \models \neg\gamma$ *and* $O \models \neg\mathsf{msc}_O(a)(c)$

Proof. See p. 30.

7 Stability

The monotonicity of $\circ_1$ (Observation 1.10) leads to the preservation of conflicts: If $O \cup \{\alpha\}$ is inconsistent, then $O \circ_1 A \cup \{\alpha\}$ is also inconsistent. Thus, if $O \cup \{\alpha\}$ is inconsistent, even repeated occurrences of α in A cannot result in $O \circ_1 A \models \alpha$. In this sense, the type-1 operators do not lead to an adaptation to the terminology of the trigger sequence but rigidly cling to the terminology of O.

In contrast to type-1 operators, type-2 operators are successful regarding the trigger statement if applied once (Observation 1.8). However, due to their non-monotone behavior regarding the ontology, a literal from a trigger sequence need not be a consequence of the ontology resulting from adding a longer sequence of triggers to the initial ontology.

Example 2. *Consider the ontology* $O = \{K(a)\}$ *and the sequence* $A = \langle K(b), \neg K(a)\rangle$.

Then $O \odot_2 A = \{K'(a), K'(b), \neg K(a), K \sqsubseteq K', K' \sqsubseteq K \sqcup K'\} \not\models K(b)$. *However,* $\tilde{A} \subseteq (O \odot_2 A) \odot_2 A = \{K'(a), K'(b), \neg K(a), K \sqsubseteq K', K' \sqsubseteq K \sqcup K', K(b)\}$, *so the repeated integration of the sequence* A *yields an ontology that includes the information contained in* A.

Since the information from the trigger sequence can get lost, the repetition of statements in the trigger sequence can be helpful to ensure that these statements are included in the final ontology. Therefore, we will study the question, for which operators repetitions in the trigger sequence can guarantee that all statements of the trigger sequence are consequences of the final ontology. As including the same literal twice (three times, four times etc.) in the trigger sequence may still not be enough to guarantee success, we will also consider infinite sequences, in which literals can re-occur infinitely often.

Definition 6. *Let* O *be an ontology,* A *be an infinite sequence of literals, and* $\circ$ *an ontology-revision operator.* $\langle O \circ A^n\rangle_{n\in\mathbb{N}}$ stabilizes (at step i)*, if there is a step* i *from which on the sequence of ontologies is constant, i.e.,*

$$O \circ A^{i+m} = O \circ A^i, \text{ for all } m \in \mathbb{N}$$

Observation 3. *Let* $\tilde{A}_i$ *be the set of literals occurring in* A *after position* i. $\langle O \circ A^n\rangle_{n\in\mathbb{N}}$ *stabilizes at step* i *iff* $\tilde{A}_i \subseteq O \circ A^i$.

Some fundamental incompatibilities between the statement sequence to be integrated and the initial ontology can enforce instability of the derived sequence of ontologies for any ontology-revision operator. For example, if the trigger sequence is inconsistent, then stabilization cannot be expected. However, if the trigger sequence stems from one source ontology and this ontology is consistent, also A is consistent. Consequently, we will evaluate the behavior of the operators mainly with respect to consistent trigger sequences.[9] Furthermore, the sequence

[9] This assumption is also found in the discussion of iterated revision in the context of belief revision [4]. It is also valid in the setting of learning theory, where all trigger statements are generated from a pre-selected model.

of ontologies need not stabilize in cases where the underlying ontologies disagree regarding the identities of the constants' referents. This can be demonstrated by Example 3.

Example 3. *Consider the ontology* $O = \{R(c,a), R(c,b), (\leq 1R)(c)\}$. *It says that* c *is in* R*-relation to* a *and* b *and that there is at most one individual to which* c *is* R*-related. Thus* $O \models (a \doteq b)$. *If* A *is the infinite sequence* $\langle K(a), \neg K(b), K(a), \neg K(b), \dots \rangle$ *with finite* $\tilde{A} = \{K(a), \neg K(b)\}$*, then stabilization cannot occur for operators that resolve conflicts by reinterpreting concept symbols.*

More generally, if according to the ontology of the receiver one object is denoted by different constants a, b but according to the ontology of the sender a, b denote different objects, then this mismatch can lead to non-stabilizing sequences of ontologies for an operator that resolves conflicts by reinterpreting concept symbols. Therefore, we will focus on combinations of sequences and ontologies, where the resulting ontologies are compatible with unique name assumptions implicit in the trigger sequence. This restriction will avoid anomalies as demonstrated in Example 3. Furthermore, we will focus on sequences based on finite sets of literals.

Definition 7. *Ontology-revision operator* $\circ$ *is* stable *iff for any consistent ontology* O *and sequence* A *of literals with finite* $\tilde{A}$ *such that for every* $n \in \mathbb{N}$ *the set* $(O \circ A^n) \cup \mathsf{una}(A^n)$ *is consistent,* $\langle O \circ A^n \rangle_{n \in \mathbb{N}}$ *stabilizes. If an ontology-revision operator is not stable, then we call it* unstable.

Even though we formulated the property of stability for ontology-revision operators in general, it is obvious that type-1 operators are not stable, due to the fact, that type-1 operators are monotone and therefore preserve conflicts. A receiver who uses a type-1 operator does not change its terminology when integrating conflicting trigger information. The initial ontology is included in the resulting ontology and the common symbol involved in the conflict is specified by the resulting ontology in the same way as in the initial ontology. However, the resulting ontology is a proper extension of the initial ontology. The situation is different for the operators of type 2. Type-2 operators are non-monotone and fulfill success in one-step application. Hence type-2 operators might be stable if all conflicts between the initial ontology and the trigger sequence get resolved during the integration process (see Observation 1.4).

7.1 Stability of $\otimes_2$

The main result of this article is the stability of the weak ontology-revision operator of type 2 ($\otimes_2$) and the instability of several stronger ontology-revision operators of type 2, namely $\odot_2$ and $\oplus_2^{\mathsf{expl}}$. To show the stability of $\otimes_2$, we will argue (Corollary 1) that given an ontology O and a sequence of literals $A = \langle \alpha_n \rangle_{n \in I}$:

1. If a conflict resolution for a literal $\alpha_i = K(a)$ is done in step i, then all ontologies derived at later steps are compatible with any $\alpha_j = K(b)$ if they are compatible with $\mathsf{una}(A^j)$.

2. If a conflict resolution for a literal $\alpha_i = \neg K(a)$ is done in step i, then all ontologies derived at later steps are compatible with any $\alpha_j = \neg K(b)$ if they are compatible with $\mathsf{una}(A^j)$.

3. There can be at most two conflict resolutions with respect to the same concept symbol if all ontologies derived are compatible with $\mathsf{una}(A)$.

The first two items show that the conflict resolution of the operator $\otimes_2$ is carried out on the terminological level rather than on the level of statements.

The details of the argument for the stability of $\otimes_2$ are given in Appendix I. In the following, we will provide some useful definitions and sketch the main steps of the complete proof. The proof is based on the observation that the set of literals from the trigger sequence that are involved in conflicts with the receiver's ontology is monotonously reduced during the integration process. Definition 8 provides the basis for identifying the conflicting literals from the sequence.

Definition 8. *Let O be an ontology and A be a sequence of literals with finite $\tilde{A}$.*

1. $\mathcal{C}(O, A) = \{M \subseteq \tilde{A} \mid O \cup \mathsf{una}(\tilde{A}) \cup M \models \bot\}$ *is* the set of all conflict sets of A relative to O.

2. $\mathcal{M}(O, A) = \{M \in \mathcal{C}(O, A) \mid$ *there is no* $M' \in \mathcal{C}(O, A)$ *with* $M' \subset M\}$ *is* the set of all (inclusion) minimal conflict sets of A relative to O.

3. $\mathcal{CL}(O, A) = \bigcup \mathcal{M}(O, A)$ *is* the set of literals of A that are essentially involved in a conflict between O and A.

Lemma 1 describes the effect of the integration of a literal from the trigger sequence into an ontology on the set of conflicting literals using a weak ontology-revision operator of type 2. While each step removes at least the integrated literal from the set of conflicting literals, reinterpretation removes every literal that differs from the integrated literal only regarding the constant. In addition, no literals are added to the set of conflicting literals.

Lemma 1. *Let $\mathcal{V}$ be a vocabulary, $K, K' \in \mathcal{V}$ concept symbols, $a \in \mathcal{V}$ a constant. Let O be an ontology, and A a sequence of literals with finite $\tilde{A}$, $\mathcal{V}(O \cup \tilde{A}) \subseteq \mathcal{V} \setminus \{K'\}$.*

1. *If $\alpha \in \tilde{A}$ and $O_1 = O \cup \{\alpha\}$, then $\mathcal{CL}(O_1, A) \subseteq \mathcal{CL}(O, A) \setminus \{\alpha\}$*

2. *If $K(a) \in \tilde{A}$ and $O_2 = O_{[K/K']} \cup \{K(a), K' \sqsubseteq K\}$, then $\mathcal{CL}(O_2, A) \subseteq \mathcal{CL}(O, A) \setminus \{K(b) \mid$ constant $b \in \mathcal{V}\}$*

3. *If $\neg K(a) \in \tilde{A}$, and $O_3 = O_{[K/K']} \cup \{\neg K(a), K \sqsubseteq K'\}$, then $\mathcal{CL}(O_3, A) \subseteq \mathcal{CL}(O, A) \setminus \{\neg K(b) \mid$ constant $b \in \mathcal{V}\}$*

Proof. See p. 35.

As a consequence of Lemma 1 and Definition 2, the integration of sequences of literals monotonously reduces the set of conflicting literals, thereby removing

any literal that has been integrated at least once. Lemma 2 states that a literal that is integrated once will not appear in the set of conflicting literals at a later step.

Lemma 2. *Let O be an ontology, A a sequence of literals with finite $\tilde{A}$, $\alpha \in \tilde{A}$, and A^n a finite prefix of A.*

1. $\mathcal{CL}(O \otimes_2 \alpha, A) \subseteq \mathcal{CL}(O, A) \setminus \{\alpha\}$
2. $\mathcal{CL}(O \otimes_2 A^n, A) \subseteq \mathcal{CL}(O, A) \setminus \tilde{A^n}$.

Proof. See p. 35.

Furthermore, conflict resolution guarantees that literals based on the same concept are permanently removed from the set of conflicting literals whenever the unique name assumption implicit in the trigger sequence is compatible with the resulting ontology.

Corollary 1. *Let $\mathcal{V}$ be a vocabulary, $K \in \mathcal{V}$ a concept symbol, $a, c \in \mathcal{V}$ constants, O an ontology over $\mathcal{V}$, and A a finite sequence of literals over $\mathcal{V}$.*

1. *If $O \cup \{K(a)\}$ is inconsistent, $O' = (O \otimes_2 K(a)) \otimes_2 A$, and $O' \cup \mathsf{una}(\tilde{A} \cup \{K(a), K(c)\})$ is consistent, then $O' \cup \{K(c)\}$ is consistent.*
2. *If $O \cup \{\neg K(a)\}$ is inconsistent, $O' = (O \otimes_2 \neg K(a)) \otimes_2 A$, and $O' \cup \mathsf{una}(\tilde{A} \cup \{\neg K(a), \neg K(c)\})$ is consistent, then $O' \cup \{\neg K(c)\}$ is consistent.*
3. *If the unique name assumption implicit in a sequence is not violated during the integration of the sequence into an ontology using the weak revision operator of type 2, no concept symbol is reinterpreted more than twice.*

Proof. See p. 36.

The following Corollary 2 expresses a weakening of success in the case of iterated application of the weak ontology-revision operator. It expresses that all conflicts between O and A^n get resolved as $O \otimes_2 A^n$ is compatible with $\tilde{A^n}$.

Corollary 2. *Let O be a consistent ontology over $\mathcal{V}_c \cup \mathcal{V}_p$ with $\mathcal{V}_c \cap \mathcal{V}_p = \emptyset$, and A a sequence of literals over $\mathcal{V}_c$ with finite $\tilde{A}$. Then for all prefixes A^n of A:*
If $(O \otimes_2 A^n) \cup \mathsf{una}(A^n)$ is consistent, then $(O \otimes_2 A^n) \cup \mathsf{una}(A^n) \cup \tilde{A^n}$ is consistent as well.

Proof. According to Lemma 2.2 $\mathcal{CL}(O \otimes_2 A^n, A^n) \subseteq \mathcal{CL}(O, A^n) \setminus \tilde{A^n} = \emptyset$. Since $(O \otimes_2 A^n) \cup \mathsf{una}(A^n)$ is consistent, this means that $(O \otimes_2 A^n) \cup \mathsf{una}(A^n) \cup \tilde{A^n}$ is consistent as well, according to Definition 8. □

As obvious from Lemma 2.2 and Observation 1.4, the repetition of a finite sequence of literals leads to the entailment of the content of the sequence if the weak ontology-revision operator of type 2 is used, as stated in Corollary 3.

Corollary 3. *Let O be a consistent ontology and A a finite sequence of literals, such that $(O \otimes_2 A) \cup \mathsf{una}(A)$ is consistent. Then*

$$(O \otimes_2 A) \otimes_2 A \models \tilde{A}$$

As a further consequence of Lemma 2, the stability of $\otimes_2$ can be proved.

Theorem 1. *The weak ontology-revision operator of type 2 ($\otimes_2$) is stable.*

Proof. See p. 36.

The stability of the weak revision operator of type 2 derives from reducing the set of literals essentially involved in conflicts in each revision step. If the unique name assumption implicit in the sequence is not violated during the integration of the sequence, then any literal that is not essentially involved in a conflict at some step can not become essential for a conflict at a later step.

7.2 Weakness of $\otimes_2$

The stability of the weak ontology-revision operator of type 2 suggests that this operator yields a terminological shift from the initial reading of the common term, as specified in the initial ontology of the receiver, to the reading of the term as specified in the sender's ontology. However, since the receiver gets only the sequence of literals as information about the sender's ontology, it can only adapt to this sequence. In addition, after the second conflict resolution with respect to the same concept symbol, the receiver's reading of the concept symbol is independent from the initial reading it assigned to the concept symbol (represented in the resulting ontology by some other internal concept symbol). This can be illustrated with another book-trading example.

Example 4. *An agent (receiver) using an ontology O_R wants to buy a cheap book on thermodynamics in an online bookshop (sender) that uses the ontology O_S.*

The receiver's price-for-value judgement is based on nothing but the price. According to O_R something is cheap if and only if it costs less than 5 Euros. Furthermore, the receiver knows already that th_1 costs more than 5 Euros (and is not cheap), while th_2 costs less than 5 Euros (and is cheap).

$$O_R \quad = \quad \{Cheap \doteq CostLt\_5, \neg CostLt\_5(th_1), CostLt\_5(th_2)\}$$

The sender has a more refined cheap-concept in which the upper prize limit depends on one of three disjoint book types—hardcover, softcover, or booklet. That the three book types are exclusive and exhaustive is specified, as well as some knowledge on the order of prices. As th_1 is known to be a hardcover book that costs less than 8 Euros, it is classified as cheap. The booklet th_2, which

costs more than 3 Euros, is not cheap.

$$\begin{aligned} O_S \;=\; & \{\mathit{Cheap} \doteq (\mathit{CostLt\_}\ 5 \sqcap \mathit{SoftC}) \sqcup (\mathit{CostLt\_}\ 8 \sqcap \mathit{HardC}) \sqcup \\ & (\mathit{CostLt\_}\ 3 \sqcap \mathit{Booklet}), \\ & \mathit{SoftC} \sqsubseteq \neg \mathit{HardC}, \mathit{Booklet} \doteq \neg(\mathit{SoftC} \sqcup \mathit{HardC}), \\ & \mathit{CostLt\_}\ 3 \sqsubseteq \mathit{CostLt\_}\ 5, \mathit{CostLt\_}\ 5 \sqsubseteq \mathit{CostLt\_}\ 8, \\ & \mathit{HardC}(\mathit{th}_1), \mathit{CostLt\_}\ 8(\mathit{th}_1), \mathit{Booklet}(\mathit{th}_2), \neg \mathit{CostLt\_}\ 3(\mathit{th}_2)\} \end{aligned}$$

Assume that the sequence $A = \langle \mathit{Cheap}(\mathit{th}_1), \neg \mathit{Cheap}(\mathit{th}_2) \rangle$ *stemming from the sender is integrated into the receiver's ontology using* $\otimes_2$*, the weak ontology revision operator of type 2. The integration of the positive literal* $\mathit{Cheap}(\mathit{th}_1)$ *demands a reinterpretation of Cheap. The newly introduced symbol* Cheap' *denotes the receiver's initial cheap-concept.*

$$\begin{aligned} O_R \otimes_2 \mathit{Cheap}(\mathit{th}_1) \;=\; & \{\mathit{Cheap}' \doteq \mathit{CostLt\_}\ 5, \neg \mathit{CostLt\_}\ 5(\mathit{th}_1), \mathit{CostLt\_}\ 5(\mathit{th}_2), \\ & \mathit{Cheap}(\mathit{th}_1), \mathit{Cheap}' \sqsubseteq \mathit{Cheap}\} \end{aligned}$$

Correspondingly, the integration of the second literal $\neg \mathit{Cheap}(\mathit{th}_2)$ *enforces another reinterpretation of Cheap. In the resulting ontology* $O_R \otimes_2 A$*, the new concept symbol* Cheap'' *denotes the receiver's interim cheap-concept.*

$$\begin{aligned} O_R \otimes_2 A \;=\; & \{\mathit{Cheap}' \doteq \mathit{CostLt\_}\ 5, \neg \mathit{CostLt\_}\ 5(\mathit{th}_1), \mathit{CostLt\_}\ 5(\mathit{th}_2), \\ & \mathit{Cheap}''(\mathit{th}_1), \mathit{Cheap}' \sqsubseteq \mathit{Cheap}'', \\ & \neg \mathit{Cheap}(\mathit{th}_2), \mathit{Cheap} \sqsubseteq \mathit{Cheap}''\} \end{aligned}$$

The integration of A into the receiver's ontology O_R *yields ontology* $O_R \otimes_2 A$ *from which no subsumption relation between the receiver's initial cheap-concept, denoted by* Cheap'*, and the new cheap-concept, denoted by Cheap, can be derived.*

The observation formulated in Example 4 can be generalized to show that the interpretation of a concept symbol that was subject to two reinterpretations solely depends on the trigger sequence A and is completely independent of the initial ontology. Theorem 2 says that every model $\mathcal{I}$ of the resulting ontology that conforms to the unique name assumption of the trigger sequence can be modified in such a way that the interpretation of the concept symbol is restricted by nothing but the trigger sequence and the denotations assigned by $\mathcal{I}$ to the constants occurring in the sequence. In this sense, the price to pay for the stability of the ontology-revision operator is the loss of semantic embedding of concept symbols in the resulting ontology.

Theorem 2. *Let* O *be a consistent ontology over vocabulary* $\mathcal{V}_c \cup \mathcal{V}_p$ *with* $\mathcal{V}_c \cap \mathcal{V}_p = \emptyset$*,* A *be a finite, consistent sequence of literals over* $\mathcal{V}_c$*,* $K \in \mathcal{V}_c$ *be any concept symbol that has been reinterpreted twice during the integration of* A *into* O *using the weak revision-operator of type 2, and* $\tilde{A}_K = \{\beta \in \tilde{A} \mid \beta \text{ contains } K\}$ *be the set of literals from* A *using* K*. Let* $\mathcal{I}$ *be a model of* $(O \otimes_2 A) \cup \mathsf{una}(A)$

and $\mathcal{I}_K^A \subseteq \Delta^{\mathcal{I}}$ *be any set, such that* $\{a^{\mathcal{I}} \mid K(a) \in \tilde{A}\} \subseteq \mathcal{I}_K^A$ *and* $\{a^{\mathcal{I}} \mid \neg K(a) \in \tilde{A}\} \cap \mathcal{I}_K^A = \emptyset$.

Define $\mathcal{J}$ *as the modification of* $\mathcal{I}$ *with*

$$\begin{array}{rcll} K^{\mathcal{J}} & = & \mathcal{I}_K^A & \\ (K'')^{\mathcal{J}} & = & (K'')^{\mathcal{I}} \cup \mathcal{I}_K^A & \text{if } K'' \in \mathcal{V}_p, K \sqsubseteq K'' \in O \otimes_2 A \\ (K'')^{\mathcal{J}} & = & (K'')^{\mathcal{I}} \cap \mathcal{I}_K^A & \text{if } K'' \in \mathcal{V}_p, K'' \sqsubseteq K \in O \otimes_2 A \end{array}$$

Then $\mathcal{J}$ *is a model of* $(O \otimes_2 A) \cup \mathsf{una}(A) \cup \tilde{A}_K$.

Proof. See p. 37.

The weak ontology-revision operator of type 2 is stable but also unable to yield useful bridging axioms whenever the different readings of the common term are not related by subsumption. Although the operator can integrate different conflicting statements without introducing a conflict into the ontology, the resulting readings of the common term might not be semantically related to the initial reading. This behavior is due to the fact that the weak operator introduces only one bound relating the common symbol and the symbol introduced in the reinterpretation step. Examples 1 and 5 (in Appendix II) show that the stronger ontology-revision operators of type 2 can yield more useful bridging axioms.

7.3 Instability of $\oplus_2^{\mathsf{expl}}$ and $\odot_2$

The stronger ontology-revision operators use the specification of the critical constant in the ontology to derive additional bridging axioms for the different readings of the common term. Unfortunately, for the stronger operators that introduce a second bound, stability is not guaranteed.

Theorem 3. *The strong ontology-revision operator of type 2 (*$\odot_2$*) and the ontology-revision operator of type 2 using upper-bound axioms based on explicitly introduced concept assertions (*$\oplus_2^{\mathsf{expl}}$*) are not stable.*

Proof. Let $\mathcal{V}_c$ and $\mathcal{V}_p$ be vocabularies such that $\mathcal{V}_c \cap \mathcal{V}_p = \emptyset$, $a, b, c, d \in \mathcal{V}_c$ be constants, and $B, C, D, E \in \mathcal{V}_c$ be concept symbols. Let the ontology O, the finite sequence A, and the infinite sequence A' (the infinite repetition of A) be given by

$$\begin{array}{rcl} O & = & \{\neg B(a), \neg C(a), \neg D(a), \neg E(a), \neg B(b), \neg C(b), \neg B(c), \neg C(c), \neg D(c), \neg E(c), \\ & & \neg B(d), \neg C(d), \neg D(d), \neg E(d)\} \\ A & = & \langle \alpha_i \rangle_{i \in \{1,\ldots,16\}} = \langle C(a), B(a), B(b), B(d), E(b), D(b), D(a), D(c), \neg B(c), \\ & & \neg C(c), \neg C(d), \neg C(b), \neg D(d), \neg E(d), \neg E(a), \neg E(c) \rangle \\ A' & = & \langle \alpha_i \rangle_{i \in \mathbb{N}}, \text{ with } \alpha_i = \alpha_{i+16} \text{ for all } i \in \mathbb{N} \end{array}$$

As O is finite and neither O nor A employ role symbols or concept constructors based on constants (i.e. nominals), for any $k \in \mathbb{N}$: $O \oplus_2^{\mathsf{expl}} (A')^k = O \odot_2 (A')^k$ (s. Lemma 6 and Corollary 4 in Appendix I).

Furthermore, for any literal α with $\mathcal{V}(\alpha) \subseteq \mathcal{V}_c$

$$\begin{array}{rcl} O \oplus_2^{\mathsf{expl}} A \models \alpha & \text{iff} & O \models \alpha \\ O \odot_2 A \models \alpha & \text{iff} & O \models \alpha \end{array}$$

and

$$\begin{array}{rcl} \mathcal{M}(O, A) = \mathcal{M}(O, A') & = & \{\{B(a)\}, \{C(a)\}, \{D(a)\}, \{B(b)\}, \{D(c)\}, \{B(d)\}\} \\ & = & \mathcal{M}(O \oplus_2^{\mathsf{expl}} A, A') = \mathcal{M}(O \odot_2 A, A') \end{array}$$

For details regarding this example see p. 39. □

Although the basic sequence of literals used in the proof is quite long, the underlying structure of the statements is very simple. In particular, the complete example is formulated in a monadic fragment of description logic (without using roles, quantifiers or number restrictions). All concept assertions in the ontology and in the trigger sequence are literals. Thus, the syntactic complexity of the added bridging axioms is minimal. For this construction it is not necessary that the additional bounds are based on computing the most specific concept. Nevertheless, in the given example, the computation of the most specific concept yields exactly the same result as extracting the explicit assertions regarding the given constant.

The combination of Theorem 1, Theorem 2, and Theorem 3 suggests an underlying tradeoff in the sense that the cost of exploiting the ontology to derive hypotheses on the reading of a concept symbol used by a communication partner is the risk to showing unstable behavior regarding repeated input.

8 Conclusion

The definition of ontology-revision operators based on reinterpretation allows one to resolve conflicts between a well-tried ontology and an incoming statement while preserving both the ontology and the conflicting statement due to the establishment of a semantic mapping between the initial ontology and the resulting ontology. The operators introduced differ regarding whether the meaning of the term specified in the initial ontology will be used as the future reading of the common term (type 1), or, whether an adaptation to the terminology of the communication partner should result (type 2).

Independently of the strength of the operator chosen, type-1 operators yield monotone extensions of the initial ontology, where the vocabulary extension in the case of conflicts is conservative. These features seem to be characteristic of communication partners that do not try to learn from solving communication problems. In the case of artificial agents, implementing the process of information integration based on type-1 operators in an incremental fashion does not seem appropriate when the trigger sequence stems from a constant communication partner that holds a consistent ontology. However, if the trigger sequence stems from different communication partners that hold different (conflicting) ontologies, the preservation of one's own terminology might be preferable to a struggle to adapt to the terminology of the trigger sequence.

Type-2 operators, on the other hand, attempt to adapt to the terminology of the communication partner by assigning the readings underlying the trigger statements to the commonly used terms in the case of conflicts. To relate these readings to the initial ontology, the operators implement different hypotheses regarding the semantic relations between the different readings of the common term. Since such hypotheses might turn out to produce new conflicts, a detailed analysis of the different options is required. The results presented in this article show a general conflict between two goals for operators that adapt to the terminology of the trigger statement. On the one hand, stable behavior during the integration of sequences of information can be guaranteed only on the basis of weak hypotheses regarding the bridging axioms relating the different readings of the term. On the other hand, only stronger operators yield useful semantic specifications of the common terms when the semantic relations between the different readings are not just a matter of one term being more general than the other.

The main problem underlying the inability to derive a stable terminology with useful semantic specifications of the different readings of a common term derives from the restricted information available to the operator in each single step. Additionally, the result of the integration process depends strongly on the order within the sequence. This suggests that integration steps should preferably be applied to larger chunks of information. Nevertheless, when we consider extensions of the ontology-revision operators that combine two ontologies [21], then the successive integration of sequences of ontologies into one ontology will yield similar cases of instability as discussed in this article.

Acknowledgements

We thank Christopher Habel for his support during the work on this article and two anonymous reviewers for their detailed and helpful comments.

Appendix I: Proofs

We add further observations and lemmata that will be helpful for the proofs.

Observation 4. *Let O be an ontology and M_1, M_2 be sets of formulae over the vocabulary $\mathcal{V}_c \cup \mathcal{V}_p$ with $\mathcal{V}_c \cap \mathcal{V}_p = \emptyset$. Let $K \in \mathcal{V}_c$, $K' \in \mathcal{V}_p \setminus \mathcal{V}(O \cup M_2)$, $L \in (\mathcal{V}_c \cup \mathcal{V}_p) \setminus \mathcal{V}(O \cup M_1 \cup M_2)$ be concept symbols, $\sigma = [K/L, K'/K]$, and $O' = O_{[K/K']} \cup M_1 \cup M_2$.*

Let $a \in \mathcal{V}_c$ be a constant, $\alpha = K(a)$ or $\alpha = \neg K(a)$, and $\circ_1$ and $\circ_2$ be a pair of corresponding type-1 and type-2 operators.

1. *$O'\sigma = O \cup M_1\sigma \cup (M_2)_{[K/L]}$.*
2. *$O \subseteq O'\sigma$.*
3. *O' has a model iff $O'\sigma$ has one.*

4. *If $O \cup \{\alpha\}$ is inconsistent and K' is the new symbol introduced in $O \circ_2 \alpha$ resp. $O \circ_1 \alpha$, then $O \subseteq (O \circ_1 \alpha)_{[K'/L]} = (O \circ_2 \alpha)\sigma$.*

Proof of Parts of Observation 1 (p. 15).

Proof of 1.1. Let O be an ontology, α be a statement, and $\circ$ an ontology-revision operator as defined above. If $O \cup \{\alpha\}$ is consistent, then all definitions yield $O \circ \alpha = O \cup \{\alpha\}$. If $O \cup \{\alpha\}$ is not consistent, and $\circ$ is a type-1 operator, then $O \circ \alpha = O \cup \{\alpha\}\sigma \cup BA$, where BA is a set of bridging axioms and σ is a substitution mapping all symbols to themselves or to a newly introduced symbol that is not in $\mathcal{V}_c \cup \mathcal{V}(O)$. According to all definitions, each bridging axiom uses the new symbol. Also, $\alpha\sigma$ contains this symbol. Therefore, $(O \circ \alpha) \cap \mathcal{L}(\mathcal{V}_c \cup \mathcal{V}(O)) = O \cap \mathcal{L}(\mathcal{V}_c \cup \mathcal{V}(O)) = O$. If $O \cup \{\alpha\}$ is not consistent, and $\circ$ is a type-2 operator, then $O \circ \alpha = O\sigma \cup \{\alpha\} \cup BA$, where BA is a set of bridging axioms and σ is a substitution restricted as in the first case. According to all definitions, each bridging axiom uses the new symbol. In addition, all formulae in $O\sigma$ that are not in O use the new symbol. Therefore, $(O \circ \alpha) \cap \mathcal{L}(\mathcal{V}_c \cup \mathcal{V}(O)) = (O\sigma \cup \{\alpha\} \cup BA) \cap \mathcal{L}(\mathcal{V}_c \cup \mathcal{V}(O)) = O\sigma \cap \mathcal{L}(\mathcal{V}_c \cup \mathcal{V}(O)) \cup \{\alpha\} \subseteq O \cup \{\alpha\}$.

Proof of 1.5 and 1.6. Let O_1, O_2 be ontologies over $\mathcal{V}_c \cup \mathcal{V}_p$ with $\mathcal{V}_c \cap \mathcal{V}_p = \emptyset$, $\mathcal{V}(O_1) \cap \mathcal{V}_p = \mathcal{V}(O_2) \cap \mathcal{V}_p$, σ be a concept substitution, and M be a set of formulae over $\mathcal{V}_c \cup \mathcal{V}_p$. If $O_1 \equiv O_2$, then $(O_1)\sigma \equiv (O_2)\sigma$, $O_1 \cup M \equiv O_2 \cup M$, and $O_i \models \mathsf{msc}_{O_1}(a) \doteq \mathsf{msc}_{O_2}(a)$ for $i \in \{1, 2\}$. Since in the case of reinterpretation the same substitution is chosen according to the assumption that the choice of a symbol $K' \in \mathcal{V}_p$ in the reinterpretation step is uniquely determined by symbol K and $\mathcal{V}_p \setminus \mathcal{V}(O_1) = \mathcal{V}_p \setminus \mathcal{V}(O_2)$ (see page 10), this results in $(O_1 \circ \alpha) \equiv (O_2 \circ \alpha)$ according to Definitions 1 and 2.

Proof of 1.13. If $O \cup \{\alpha\}$ is consistent, then O is consistent and $O \circ \alpha = O \cup \{\alpha\}$ is consistent. So assume that $O \cup \{\alpha\}$ is inconsistent. If O is inconsistent, then $O \circ_1 \alpha$ is inconsistent according to 9 of this observation, and the inconsistency of $O \circ_2 \alpha$ follows with Observation 4.3. If O is consistent, it has a model $\mathcal{I} \models O$. We prove the consistency of $O \odot_i \alpha$, the consistency of the other operators $\circ_i$ then follows with part 11 of this observation. If $\alpha = K(a)$, then $O \models \neg K(a)$ and $O \odot_1 \alpha = O \cup \{K'(a), K \sqsubseteq K', K' \sqsubseteq K \sqcup \mathsf{msc}_O(a)\}$. Define the modification $\mathcal{J}$ of $\mathcal{I}$ by setting $K'^{\mathcal{J}} = K^{\mathcal{I}} \cup \{a^{\mathcal{I}}\}$. Then clearly $\mathcal{J} \models O \odot_1 \alpha$. If $\alpha = \neg K(a)$, then $O \models K(a)$ and $O \odot_1 \alpha = O \cup \{\neg K'(a), K' \sqsubseteq K, K \sqsubseteq K' \sqcup \mathsf{msc}_O(a)\}$. Now define the modification $\mathcal{J}$ of $\mathcal{I}$ by setting $K'^{\mathcal{J}} = K^{\mathcal{I}} \setminus \{a^{\mathcal{I}}\}$. Again $\mathcal{J} \models O \odot_1 \alpha$ results. With Observation 4.3 the consistency of $O \odot_2 \alpha$ follows. □

Proof of Proposition 1 (p. 17).

We will prove the more elaborate Proposition 4 as parts 1 and 2 can more easily be proved in the context of part 3.

Proposition 4. *For any ontology O over $\mathcal{V}_c \cup \mathcal{V}_p$ with $\mathcal{V}_c \cap \mathcal{V}_p = \emptyset$, finite sequence A of literals over $\mathcal{V}_c$, and ontology-revision operator $\circ$ there are two substitutions of concept symbols (σ and ρ), such that*

1. $O\sigma \subseteq O \circ A$,

2. $O \cup \tilde{A} \subseteq (O \circ A)\rho$,

3. *for any concept symbol* $L \in \mathcal{V}(O) \cup \mathcal{V}_c$, $L = L\rho$

Proof. Let O be an ontology over $\mathcal{V}_c \cup \mathcal{V}_p$ with $\mathcal{V}_c \cap \mathcal{V}_p = \emptyset$. If A is the empty sequence ($O \circ A = O$), the neutral substitution fulfills the conditions for σ and ρ.

Let $A = A^{n+1}$ be a sequence of literals over $\mathcal{V}_c$ of length $n+1$ and the assumption be proved for A^n, such that σ_n and ρ_n are the substitutions fulfilling all conditions of this proposition.

If $O \circ A^n \cup \{\alpha_{n+1}\}$ is consistent, then $O\sigma_n \subseteq O \circ A^n \subseteq O \circ A^n \cup \{\alpha_{n+1}\} = O \circ A^{n+1}$ and $O \cup \tilde{A}^{n+1} = O \cup \tilde{A}^n \cup \{\alpha_{n+1}\} \subseteq (O \circ A^n)\rho_n \cup \{\alpha_{n+1}\} = ((O \circ A^n) \cup \{\alpha_{n+1}\})\rho_n = (O \circ A^{n+1})\rho_n$, since $\mathcal{V}(\alpha_{n+1}) \subseteq \mathcal{V}_c$ and therefore $\alpha_{n+1} = \alpha_{n+1}\rho_n$. Thus, σ_n and ρ_n do the job for A^{n+1} as well as for A^n.

If $O \circ A^n \cup \{\alpha_{n+1}\}$ is inconsistent, then there is a concept symbol $K \in \mathcal{V}(\alpha_{n+1}) \subseteq \mathcal{V}_c$ and a concept symbol $K' \in \mathcal{V}_p$, such that $K' \in \mathcal{V}(O \circ A^{n+1}) \setminus \mathcal{V}(O \circ A^n)$ and K' is introduced to resolve the conflict.

Let ρ_{n+1} be the composition of $[K'/K]$ and ρ_n. It is easy to verify that this choice of ρ_{n+1} fulfills the condition 3 of this proposition, given that ρ_n fulfills this condition.

Since $O \cup \tilde{A}^{n+1} = O \cup \tilde{A}^n \cup \{\alpha_{n+1}\} \subseteq (O \circ A^n)\rho_n \cup \{\alpha_{n+1}\}$, we prove part 2 ($O \cup \tilde{A}^{n+1} \subseteq (O \circ A^{n+1})\rho_{n+1}$) by showing that $O \circ A^n \subseteq (O \circ A^{n+1})_{[K'/K]}$ and $\alpha_{n+1} \in (O \circ A^{n+1})_{[K'/K]}\rho_n$.

If $\circ = \circ_1$ is a type-1 operator, then we have by definition $O \circ_1 A^n \cup \{(\alpha_{n+1})_{[K/K']}\} \subseteq O \circ_1 A^{n+1}$. Since K' does not occur in $O \circ_1 A^n$, we have $O \circ_1 A^n = (O \circ_1 A^n)_{[K'/K]} \subseteq (O \circ_1 A^{n+1})_{[K'/K]}$. Since K' does not occur in α_{n+1}, also $\alpha_{n+1} = (\alpha_{n+1})_{[K/K'][K'/K]} \in (O \circ_1 A^{n+1})_{[K'/K]}$. Since $\mathcal{V}(\alpha_{n+1}) \subseteq \mathcal{V}_c, \alpha_{n+1} = (\alpha_{n+1})\rho_n \in (O \circ_1 A^{n+1})_{[K'/K]}\rho_n$, yielding part 2 of this proposition for type-1 operators.

In this case let σ_{n+1} be the neutral substitution. $O\sigma_{n+1} \subseteq O \circ_1 A^{n+1}$ according to Observation 1.10. Consequently, part 1 holds in the case of type-1 operators.

If $\circ = \circ_2$ is a type-2 operator, and given that K' does not occur in $O \circ_2 A^n$, $O \circ_2 A^n = (O \circ_2 A^n)_{[K/K'][K'/K]} \subseteq ((O \circ_2 A^n)_{[K/K']} \cup \{\alpha_{n+1}\})_{[K'/K]} \subseteq (O \circ_2 A^{n+1})_{[K'/K]}$. In addition, with $\mathcal{V}(\alpha_{n+1}) \subseteq \mathcal{V}_c$ we get $\alpha_{n+1} = (\alpha_{n+1})_{[K'/K]} = (\alpha_{n+1})_{[K'/K]}\rho_n \in (O \circ A^{n+1})_{[K'/K]}\rho_n$. Consequently, part 2 holds in the case of type-2 operators.

In this case let σ_{n+1} be the composition of σ_n and $[K/K']$. We have $O\sigma_{n+1} = (O\sigma_n)_{[K/K']} \subseteq (O \circ_2 A^n)_{[K/K']} \subseteq (O \circ_2 A^n)_{[K/K']} \cup \{\alpha_{n+1}\} \subseteq O \circ_2 A^{n+1}$ by definition, yielding part 1 of this proposition for type-2 operators. □

□

Proof of Proposition 2 (p. 17).

Let O be an ontology and α a literal such that $O \cup \{\alpha\}$ is inconsistent. Let $\circ_1$ be an ontology-revision operator of type 1. We have to show that $O \circ_1 \alpha$ is a conservative extension of O.

If β is a formula with $\mathcal{V}(\beta) \subseteq \mathcal{V}(O)$ and $O \models \beta$, then also $O \circ_1 \alpha \models \beta$, because $O \subseteq O \circ_1 \alpha$. Now suppose that $O \not\models \beta$ for β over $\mathcal{V}(O)$. We have to show that $O \circ_1 \alpha \not\models \beta$. We show this for $\circ_1 = \odot_1$. For $\otimes_1$, $\oplus_1^{\mathsf{expl}}$, and $\oplus_1^{\mathsf{sel}}$ the assertion then follows from Observation 1.11.

We show the proposition for positive literals $\alpha = K(a)$. Let K' be the new concept symbol introduced by the reinterpretation rule and $O \odot_1 K(a) = O \cup \{K \sqsubseteq K', K' \sqsubseteq K \sqcup \mathsf{msc}_O(a), K'(a)\}$. Since $O \not\models \beta$, there is a model $\mathcal{I}$ of $O \cup \neg\beta$. Define $\mathcal{J}$ as the modification of $\mathcal{I}$ with $K'^{\mathcal{J}} = K^{\mathcal{I}} \cup \{a^{\mathcal{I}}\}$. Then $\mathcal{J} \models O \cup \{\neg\beta\}$, since $\mathcal{J}$ is the same as $\mathcal{I}$ for all symbols in $\mathcal{V}(O)$, and $\mathcal{I} \models O \cup \{\neg\beta\}$, $\mathcal{J} \models \{(K \sqsubseteq K'), (K' \sqsubseteq K \sqcup \mathsf{msc}_O(a)), K'(a)\}$ (by construction of $\mathcal{J}$), and therefore $\mathcal{J} \models O \odot_1 K(a) \cup \{\neg\beta\}$.

The proof for negative literals $\alpha = \neg K(a)$ is done similarly by selecting the model $\mathcal{J}$ with $K'^{\mathcal{J}} = K^{\mathcal{I}} \setminus \{a^{\mathcal{I}}\}$. □

Proof of Proposition 3 (p. 18).

Let O be an ontology over the vocabulary $\mathcal{V}_c \cup \mathcal{V}_p$ with $\mathcal{V}_c \cap \mathcal{V}_p = \emptyset$, $K \in \mathcal{V}_c$ be a concept symbol, and $a, c \in \mathcal{V}_c$ be constants, such that $\mathsf{msc}_O(a)$ exists. Let $\alpha = K(a)$ and $\gamma = K(c)$ or $\alpha = \neg K(a)$ and $\gamma = \neg K(c)$, such that $O \models \neg\alpha$. Let $K' \in \mathcal{V}_p \setminus \mathcal{V}(O)$ be the new symbol introduced in $O \circ_2 \alpha$, and β be a formula with $\mathcal{V}(\beta) \subseteq (\mathcal{V}_c \cup \mathcal{V}(O)) \setminus \{K\}$.

In the proofs the substitution $\sigma = [K/L, K'/K]$ with $L \in (\mathcal{V}_c \cup \mathcal{V}_p) \setminus \mathcal{V}(O \circ_2 \alpha)$ will be used. Because of the fact that $O \subseteq (O \circ_2 \alpha)\sigma$ (see Observation 4), the modification of the models in the proofs will be more readable.

Proof of 3.1 $O \circ_2 \alpha \models \beta$ iff $O \models \beta$
As $K, K' \notin \mathcal{V}(\beta)$ we have $\beta\sigma = \beta$.

First assume $O \models \beta$. We have to show $O \circ_2 \alpha \models \beta$. Applying σ this reduces to showing $(O \circ_2 \alpha)\sigma \models \beta$. But this is the case because of $O \subseteq (O \circ_2 \alpha)\sigma$ and the monotonicity of $\models$.

For the other direction, we begin with the case $\alpha = K(a)$ and consider $\odot_2$ in place of $\circ_2$. Thus, assume $O \odot_2 K(a) \models \beta$, applying σ, this leads to $O \cup \{L(a), K \sqsubseteq L, L \sqsubseteq K \sqcup \mathsf{msc}_O(a)\} \models \beta$. Let $\mathcal{I}$ be any model of O. Let $\mathcal{J}$ be the modification of $\mathcal{I}$ with $L^{\mathcal{J}} = K^{\mathcal{I}} \cup \{a^{\mathcal{I}}\}$. Then $\mathcal{J} \models O \cup \{L(a), K \sqsubseteq L, L \sqsubseteq K \sqcup \mathsf{msc}_O(a)\}$ and hence $\mathcal{J} \models \beta$. As L does not occur in β, this means that also $\mathcal{I} \models \beta$. We have shown the assertion that if $O \odot_2 K(a) \models \beta$, then $O \models \beta$. The general assertion for $\circ_2$ follows with Observation 1.11.

The case $\alpha = \neg K(a)$ can be proved in the same fashion, but this time selecting $\mathcal{J}$ based on $\mathcal{I}$ with $L^{\mathcal{J}} = K^{\mathcal{I}} \setminus \{a^{\mathcal{I}}\}$.

Proof of 3.2 $O \circ_2 \alpha \models \gamma$ iff $O \cup \{a \not\doteq c\} \models \gamma$
We begin by considering the case $\alpha = K(a)$ and $\gamma = K(c)$. First assume $O \cup \{a \not\doteq c\} \models K(c)$. Then also $O \circ_2 K(a) \cup \{a \not\doteq c\} \models K'(c)$ and since $K' \sqsubseteq K \in O \circ_2 K(a)$ also $O \circ_2 K(a) \cup \{a \not\doteq c\} \models K(c)$. Now let $\mathcal{I}$ be a model of $O \circ_2 K(a)$. If $a^{\mathcal{I}} \neq c^{\mathcal{I}}$, then $\mathcal{I} \models a \not\doteq c$, and $\mathcal{I} \models K(c)$ follows. If, on the other hand, $a^{\mathcal{I}} = c^{\mathcal{I}}$, then because of $K(a) \in O \circ_2 K(a)$ also $c^{\mathcal{I}} \in K^{\mathcal{I}}$ results, i.e., $\mathcal{I} \models K(c)$.

Now assume $O \cup \{a \not\doteq c\} \not\models K(c)$. Let $\mathcal{I}$ be a model of $O \cup \{a \not\doteq c, \neg K(c)\}$. Consequently $a^{\mathcal{I}} \neq c^{\mathcal{I}}$ and $c^{\mathcal{I}} \notin K^{\mathcal{I}}$. We have to show $O \odot_2 K(a) \not\models K(c)$. The general assertion for $\circ_2$ then follows with Observation 1.11. Applying the substitution σ to both sides of the entailment results in the task to show

$$O \cup \{L(a), K \sqsubseteq L, L \sqsubseteq K \sqcup \mathsf{msc}_O(a)\} \not\models L(c) \tag{1}$$

Let $\mathcal{J}$ be the modification of $\mathcal{I}$ with $L^{\mathcal{J}} = K^{\mathcal{I}} \cup \{a^{\mathcal{I}}\}$. Then $\mathcal{J}$ is a model of $O \cup \{\neg K(c)\}$ and additionally a model of $\{L(a), K \sqsubseteq L, L \sqsubseteq K \sqcup \mathsf{msc}_O(a), \neg L(c)\}$ showing (1).

The proof for the other case ($\alpha = \neg K(a)$ and $\gamma = \neg K(c)$) is similar, using the modification $\mathcal{J}$ of the model $\mathcal{I}$ of $O \cup \{a \not\doteq c, K(c)\}$ with $L^{\mathcal{J}} = K^{\mathcal{I}} \setminus \{a^{\mathcal{I}}\}$.

Proof of 3.3 $O \otimes_2 \alpha \not\models \neg\gamma$
We begin with case $\alpha = K(a)$ and $\gamma = K(c)$. Let $\mathcal{I}$ be a model of $O \otimes_2 K(a)$. Let $\mathcal{J}$ be the modification of $\mathcal{I}$ with $K^{\mathcal{J}} = \Delta^{\mathcal{I}}$. Then $\mathcal{J}$ is a model of $O \otimes_2 K(a)$ and of $K(c)$. (Remember that $K' \sqsubseteq K$ and $K(a)$ are the only formulae of $O \otimes_2 K(a)$ that involve K.)

For the other case let $\mathcal{I}$ be a model of $O \otimes_2 \neg K(a)$ and $\mathcal{J}$ be the modification of $\mathcal{I}$ with $K^{\mathcal{J}} = \emptyset$. Then $\mathcal{J}$ is a model of $O \otimes_2 \neg K(a)$ and of $\neg K(c)$.

Proof of 3.4 $O \odot_2 \alpha \models \neg\gamma$ iff $O \models \neg\gamma$ and $O \models \neg\mathsf{msc}_O(a)(c)$
We begin with case $\alpha = K(a)$ and $\gamma = K(c)$. First assume $O \models \neg K(c)$ and $O \models \neg\mathsf{msc}_O(a)(c)$. Then $O \odot_2 K(a) \models \neg K'(c)$, $O \odot_2 K(a) \models \neg\mathsf{msc}_{O_{[K/K']}}(a)(c)$, and because of $O \odot_2 K(a) \models K \sqsubseteq K' \sqcup \mathsf{msc}_{O_{[K/K']}}(a)$ also $O \odot_2 K(a) \models \neg K(c)$.

Now we want to show, if $O \not\models \neg K(c)$, then $O \odot_2 K(a) \not\models \neg K(c)$ and if $O \not\models \neg\mathsf{msc}_O(a)(c)$, then $O \odot_2 K(a) \not\models \neg K(c)$.

Assume $O \not\models \neg\mathsf{msc}_O(a)(c)$. Let $\mathcal{I}$ be a model of $O \cup \{\mathsf{msc}_O(a)(c)\}$ and construct $\mathcal{J}$ as the modification of $\mathcal{I}$ with $L^{\mathcal{J}} = K^{\mathcal{I}} \cup \{a^{\mathcal{I}}, c^{\mathcal{I}}\}$. Then $c^{\mathcal{J}} \in L^{\mathcal{J}}$ and $\mathcal{J} \models (O \odot_2 K(a))\sigma$ and so also $\mathcal{J} \models (O \odot_2 K(a) \cup \{K(c)\})\sigma$ resulting in $O \odot_2 K(a) \not\models \neg K(c)$.

Assume $O \not\models \neg K(c)$. Let $\mathcal{I}$ be a model of $O \cup \{K(c)\}$. Again use the modification $\mathcal{J}$ of $\mathcal{I}$ with $L^{\mathcal{J}} = K^{\mathcal{I}} \cup \{a^{\mathcal{I}}, c^{\mathcal{I}}\}$. Then as above $\mathcal{J} \models (O \odot_2 K(a) \cup \{K(c)\})\sigma$ and $O \odot_2 K(a) \not\models \neg K(c)$ results.

For the other case ($\alpha = \neg K(a)$ and $\gamma = \neg K(c)$) the proof is similar. In the second part one has to construct the modification $\mathcal{J}$ of interpretation $\mathcal{I}$ such that $\mathcal{I} \models O \cup \{\mathsf{msc}_O(a)(c)\}$ or $\mathcal{I} \models O \cup \{\neg K(c)\}$ with $L^{\mathcal{J}} = K^{\mathcal{I}} \setminus \{a^{\mathcal{I}}, c^{\mathcal{I}}\}$. □

Observation 5. *Let $\mathcal{V}$ be a vocabulary, $K, K' \in \mathcal{V}$ concept symbols, O an ontology, A a sequence of literals with finite $\tilde{A}$ such that $\mathcal{V}(O \cup \tilde{A}) \subseteq \mathcal{V} \setminus \{K'\}$, $\alpha \in \tilde{A}$, and A^n a finite prefix of A.*

1. *If $O \cup \mathsf{una}(A)$ is inconsistent, then $O \cup \{\alpha\} \cup \mathsf{una}(A)$ is inconsistent.*
2. *If $O \cup \mathsf{una}(A)$ is inconsistent, then $O_{[K/K']} \cup \mathsf{una}(A)$ is inconsistent.*
3. *If $(O \otimes_2 \alpha) \cup \mathsf{una}(A)$ is consistent, then $O \cup \mathsf{una}(A)$ is consistent.*

4. *If* $(O \otimes_2 A^n) \cup \mathsf{una}(A)$ *is consistent, then* $O \cup \mathsf{una}(A)$ *is consistent.*

Proof.

1. Trivial.
2. This is a consequence of the fact that K' does not occur in O and K, K' do not occur in $\mathsf{una}(A)$.
3. Direct consequence of Definition 2 and parts 1 and 2 of this observation.
4. Derives from part 3 of this observation by induction on the length of A. □

□

Observation 6. *Let* O *be an ontology,* A *a sequence of literals with finite* $\tilde{A}$*, and* α *a literal.*

1. $\mathcal{M}(O, A) \subseteq \mathcal{C}(O, A) \subseteq 2^{\tilde{A}}$ *are finite and* $\mathcal{CL}(O, A) \subseteq \tilde{A}$ *is finite.*
2. $M \in \mathcal{C}(O, A)$ *iff there is a* $M' \in \mathcal{M}(O, A)$ *such that* $M' \subseteq M \subseteq \tilde{A}$*.*
   $\mathcal{C}(O, A) = \{M \subseteq \tilde{A} \mid$ *there is a* $M' \in \mathcal{M}(O, A)$ *such that* $M' \subseteq M\}$*.*
3. $\mathcal{M}(O, A) = \emptyset$ *iff* $O \cup \tilde{A}$ *is consistent.*
4. $\mathcal{M}(O, A) = \{\emptyset\}$ *iff* $O \cup \mathsf{una}(A)$ *is inconsistent.*
5. $\{\alpha\} \in \mathcal{M}(O, A)$ *iff* $O \cup \mathsf{una}(A)$ *is consistent,* $\alpha \in \tilde{A}$*, and* $O \cup \mathsf{una}(A) \models \neg\alpha$*.*
6. *If* $O \cup \mathsf{una}(A) \models \alpha$*, then* $\alpha \notin \mathcal{CL}(O, A)$*.*

Lemma 3. *Let* O *be an ontology,* A *a sequence of literals with finite* $\tilde{A}$*,* $\alpha \in \tilde{A}$*, and* $O_1 = O \cup \{\alpha\}$*.*

1. $\mathcal{C}(O_1, A) = \{M \subseteq \tilde{A} \mid M \cup \{\alpha\} \in \mathcal{C}(O, A)\}$
2. $\mathcal{M}(O_1, A) \subseteq \{M \setminus \{\alpha\} \mid M \in \mathcal{M}(O, A)\}$

Proof.

1. According to $O_1 = O \cup \{\alpha\}$ and Definition 8.1, $M \in \mathcal{C}(O_1, A)$ iff $M \subseteq \tilde{A}$ and $O \cup \{\alpha\} \cup \mathsf{una}(A) \cup M \models \bot$, which is the same as $M \cup \{\alpha\} \in \mathcal{C}(O, A)$.
2. If $Y \in \mathcal{M}(O_1, A)$ then (Definition 8.2) $Y \in \mathcal{C}(O_1, A)$, thus (part 1 of this lemma) $Y \cup \{\alpha\} \in \mathcal{C}(O, A)$. According to Observation 6.2, there is a $M \in \mathcal{M}(O, A)$, such that $M \subseteq Y \cup \{\alpha\}$ and $M \setminus \{\alpha\} \subseteq Y$. Since $M \in \mathcal{C}(O, A)$, also $M \setminus \{\alpha\} \in \mathcal{C}(O_1, A)$ (part 1 of this lemma), and $M \setminus \{\alpha\} = Y$ according to Definition 8.2. □

□

Definition 9. *Let* $\mathcal{V}$ *be a vocabulary,* A *a sequence of literals over vocabulary* $\mathcal{V}$ *with finite* $\tilde{A}$*,* $K \in \mathcal{V}$ *a concept symbol,* $B = K$ *or* $B = \neg K$*.*

$\mathcal{SL}(A, B) = \{M \subseteq \tilde{A} \mid$ *there is a constant* $b \in \mathcal{V}$ *such that* $B(b) \in M\}$ *is* the set of subsets of $\tilde{A}$ holding a literal based on B.

Lemma 4. *Let $\mathcal{V}$ be a vocabulary, $K, K' \in \mathcal{V}$ concept symbols, $a \in \mathcal{V}$ a constant. Let O be an ontology, and A a sequence of literals with finite $\tilde{A}$, $\mathcal{V}(O \cup \tilde{A}) \subseteq \mathcal{V} \setminus \{K'\}$, $K(a) \in \tilde{A}$, and $O_2 = O_{[K/K']} \cup \{K(a), K' \sqsubseteq K\}$.*

1. $\mathcal{C}(O_2, A) \subseteq \mathcal{C}(O, A)$
2. $\mathcal{C}(O, A) \setminus \mathcal{SL}(A, K) \subseteq \mathcal{C}(O_2, A)$
3. $\mathcal{M}(O, A) \setminus \mathcal{SL}(A, K) \subseteq \mathcal{M}(O_2, A)$
4. $\mathcal{M}(O_2, A) \cap \mathcal{SL}(A, K) = \emptyset$
5. $\mathcal{M}(O_2, A) \subseteq \mathcal{M}(O, A)$
6. $\mathcal{M}(O_2, A) = \mathcal{M}(O, A) \setminus \mathcal{SL}(A, K)$

Proof. Let $L \in \mathcal{V} \setminus \mathcal{V}(O \cup \tilde{A})$ be a concept symbol.

1. We prove $\mathcal{C}(O_2, A) \subseteq \mathcal{C}(O, A)$ by showing that if $M \subseteq \tilde{A}$ and $O \cup \mathsf{una}(A) \cup M$ has a model, then $O_2 \cup \mathsf{una}(A) \cup M$ has a model as well.

   Let $\mathcal{I}$ be a model of $O \cup \mathsf{una}(A) \cup M$. Let $\mathcal{J}$ be the modification of $\mathcal{I}$ with $L^{\mathcal{J}} = K^{\mathcal{I}} \cup \{a^{\mathcal{I}}\}$. By construction, $\mathcal{J} \models O \cup \{L(a), K \sqsubseteq L\} \cup \mathsf{una}(A)$ is obvious. We will show that also $\mathcal{J} \models M_{[K/L]}$.

   If $\gamma \in M_{[K/L]}$ and L does not occur in γ, then $\gamma \in M$, and $\mathcal{J} \models \gamma$ derives from $\mathcal{I} \models \gamma$ and the construction of $\mathcal{J}$.

   If $L(b) \in M_{[K/L]}$, then $K(b) \in M$, $\mathcal{I} \models K(b)$ and $K^{\mathcal{I}} \subseteq L^{\mathcal{J}}$ yields $\mathcal{J} \models L(b)$.

   If $\neg L(b) \in M_{[K/L]}$, then $\neg K(b) \in M \subseteq \tilde{A}$, $\mathcal{I} \models \neg K(b)$, $a^{\mathcal{I}} \neq b^{\mathcal{I}}$ (as $K(a), \neg K(b) \in \tilde{A}$ and $\mathcal{I} \models \mathsf{una}(A)$) and $L^{\mathcal{J}} = K^{\mathcal{I}} \cup \{a^{\mathcal{I}}\}$ yields $\mathcal{J} \models \neg L(b)$.

   Consequently, $\mathcal{J} \models (O_2 \cup \mathsf{una}(\tilde{A}) \cup M)_{[K/L, K'/K]}$, therefore (Observation 4.3) $O_2 \cup \mathsf{una}(A) \cup M$ has a model as well.

2. We prove $\mathcal{C}(O, A) \setminus \mathcal{SL}(A, K) \subseteq \mathcal{C}(O_2, A)$ by showing that if $M \subseteq \tilde{A}$ does not contain a literal of the form $K(b)$ and $M \notin \mathcal{C}(O_2, A)$, then $M \notin \mathcal{C}(O, A)$.

   Let $M \subseteq \tilde{A}$, $M \notin \mathcal{C}(O_2, A)$, such that M does not contain a literal of the form $K(b)$. Then (Observation 4.3) there is a model $\mathcal{I}$ of $O \cup \{L(a), K \sqsubseteq L\} \cup \mathsf{una}(A) \cup M_{[K/L]}$. We will show that $\mathcal{I} \models M$.

   If $\gamma \in M$ and K does not occur in γ, then $\gamma \in M_{[K/L]}$, and $\mathcal{I} \models \gamma$ derives from $\mathcal{I} \models M_{[K/L]}$.

   If $\gamma \in M$ and K occurs in γ, then $\gamma = \neg K(b)$ for some constant b, since literals of the form $K(b)$ do not occur in M. Then $\neg L(b) \in M_{[K/L]}$, $\mathcal{I} \models \neg L(b)$, therefore $b^{\mathcal{I}} \notin L^{\mathcal{I}} \supseteq K^{\mathcal{I}}$ (because of $\mathcal{I} \models K \sqsubseteq L$) and therefore $\mathcal{I} \models \neg K(b)$.

   Consequently, $\mathcal{I} \models O \cup \mathsf{una}(A) \cup M$, thus $M \notin \mathcal{C}(O, A)$.

3. Assume $M \in \mathcal{M}(O, A) \setminus \mathcal{SL}(A, K)$ and $X \subset M$. According to part 2 of this lemma, $M \in \mathcal{C}(O_2, A)$. We will show that $X \notin \mathcal{C}(O_2, A)$, yielding $M \in \mathcal{M}(O_2, A)$.

   $M \in \mathcal{M}(O, A)$ means that $O \cup \mathsf{una}(A) \cup M \models \bot$ and that M is (inclusion) minimal in this respect. With $X \subset M$ this yields that $O \cup \mathsf{una}(A) \cup X$ is consistent.

   Let $\mathcal{I}$ be a model of $O \cup \mathsf{una}(A) \cup X$. Let $\mathcal{J}$ be the modification of $\mathcal{I}$ with $L^{\mathcal{J}} = K^{\mathcal{I}} \cup \{a^{\mathcal{I}}\}$. By construction we get $\mathcal{J} \models O \cup \{L(a), K \sqsubseteq L\} \cup \mathsf{una}(A)$. We will show that $\mathcal{J} \models X_{[K/L]}$.

   If $\gamma \in X_{[K/L]}$ and L does not occur in γ, then $\gamma \in X$ and $\mathcal{J} \models \gamma$ derives from $\mathcal{I} \models \gamma$ and the construction of $\mathcal{J}$.

   If $\gamma \in X_{[K/L]}$ and L occurs in γ, then $\gamma = \neg L(b)$ for some constant $b \in \mathcal{V}$, since literals of the form $L(b)$ do not occur in $X_{[K/L]}$. In this case, $\neg K(b) \in X \subseteq \tilde{A}$, $\mathcal{I} \models \neg K(b)$, $a^{\mathcal{I}} \neq b^{\mathcal{I}}$ (since $K(a), \neg K(b) \in \tilde{A}$ and $\mathcal{I} \models \mathsf{una}(A)$) and $L^{\mathcal{J}} = K^{\mathcal{I}} \cup \{a^{\mathcal{I}}\}$ yields $\mathcal{J} \models \neg L(b)$.

   Thus, $(O_2 \cup \mathsf{una}(A) \cup X)_{[K/L, K'/K]}$ is consistent, therefore (Observation 4.3) $O_2 \cup \mathsf{una}(A) \cup X$ is consistent, which means $X \notin \mathcal{C}(O_2, A)$.

4. Assume $K(b) \in \tilde{A}$ for some constant $b \in \mathcal{V}$, and $K(b) \notin M \subseteq \tilde{A}$, such that $M \notin \mathcal{C}(O_2, A)$. We will show that $M \cup \{K(b)\} \notin \mathcal{C}(O_2, A) \supseteq \mathcal{M}(O_2, A)$.

   Let $\mathcal{I}$ be a model of $O_2 \cup \mathsf{una}(A) \cup M$. Let $\mathcal{J}$ be the modification of $\mathcal{I}$ with $K^{\mathcal{J}} = K^{\mathcal{I}} \cup \{b^{\mathcal{I}}\}$. We will show that $\mathcal{J} \models O_2 \cup \mathsf{una}(A) \cup M \cup \{K(b)\}$.

   Remember $O_2 = O_{[K/K']} \cup \{K(a), K' \sqsubseteq K\}$.

   Since $\mathcal{I} \models O_{[K/K']}$ and K does not occur in $O_{[K/K']}$, $\mathcal{J} \models O_{[K/K']}$. Obviously, $\mathcal{J} \models \{K(a), K' \sqsubseteq K\}$, thus $\mathcal{J} \models O_2$. Also $\mathcal{J} \models \mathsf{una}(A)$, since $\mathcal{I} \models \mathsf{una}(A)$. Regarding the question whether $\mathcal{J} \models M \cup \{K(b)\}$, we have to consider only those literals from M that are of the form $\neg K(c)$ for some constant $c \in \mathcal{V}$, because $\mathcal{I} \models M$ and $K^{\mathcal{I}} \subseteq K^{\mathcal{J}}$, $\mathcal{J} \models K(b)$ by construction of $\mathcal{J}$.

   Let $\neg K(c) \in M$. Since $K(b), \neg K(c) \in \tilde{A}$ and $\mathcal{I} \models \mathsf{una}(A)$, $b^{\mathcal{I}} \neq c^{\mathcal{I}}$. Since $\neg K(c) \in M$ and $\mathcal{I} \models M$, $\mathcal{I} \models \neg K(c)$. By construction of $\mathcal{J}$, this means that $\mathcal{J} \models \neg K(c)$ as well.

   Thus, $\mathcal{J} \models O_2 \cup \mathsf{una}(A) \cup M \cup \{K(b)\}$, and $M \cup \{K(b)\} \notin \mathcal{C}(O_2, A)$.

5. $\mathcal{M}(O_2, A) \subseteq \mathcal{C}(O, A)$ is given by part 1 of this lemma. Assume $X \subset M \in \mathcal{M}(O_2, A)$. We will show $X \notin \mathcal{C}(O, A)$. Therefore $M \in \mathcal{M}(O, A)$.

   Since $X \notin \mathcal{C}(O_2, A)$, $O_2 \cup \mathsf{una}(A) \cup X$ has a model, therefore (Observation 4.3) $O \cup \{L(a), K \sqsubseteq L\} \cup \mathsf{una}(A) \cup X_{[K/L]}$ has a model.

   Let $\mathcal{I}$ be a model of $O \cup \{L(a), K \sqsubseteq L\} \cup \mathsf{una}(A) \cup X_{[K/L]}$. We will show $\mathcal{I} \models X$. For this, we have to consider literals from X based on K only (since $\mathcal{I} \models X_{[K/L]}$). However, a literal of the form $K(b)$ does not occur in $X \subset M$ (according to part 4 of this lemma).

If $\neg K(b) \in X$ then $\mathcal{I} \models \neg L(b)$ and, since $\mathcal{I} \models K \sqsubseteq L$, also $\mathcal{I} \models \neg K(b)$. Consequently, $\mathcal{I} \models O \cup \mathsf{una}(A) \cup X$ and $X \notin \mathcal{C}(O, A)$.

6. $\mathcal{M}(O_2, A) = \mathcal{M}(O, A) \setminus \mathcal{SL}(A, K)$ derives as a combination of parts 3, 4, and 5 of this lemma. □

□

Lemma 5. *Let $\mathcal{V}$ be a vocabulary, $K, K' \in \mathcal{V}$ concept symbols, $a \in \mathcal{V}$ a constant. Let O be an ontology, and A a sequence of literals with finite $\tilde{A}$, such that $\mathcal{V}(O \cup \tilde{A}) \subseteq \mathcal{V} \setminus \{K'\}$, $\neg K(a) \in \tilde{A}$, and $O_3 = O_{[K/K']} \cup \{\neg K(a), K \sqsubseteq K'\}$.*

1. $\mathcal{C}(O_3, A) \subseteq \mathcal{C}(O, A)$
2. $\mathcal{C}(O, A) \setminus \mathcal{SL}(A, \neg K) \subseteq \mathcal{C}(O_3, A)$
3. $\mathcal{M}(O, A) \setminus \mathcal{SL}(A, \neg K) \subseteq \mathcal{M}(O_3, A)$
4. $\mathcal{M}(O_3, A) \cap \mathcal{SL}(A, \neg K) = \emptyset$
5. $\mathcal{M}(O_3, A) \subseteq \mathcal{M}(O, A)$
6. $\mathcal{M}(O_3, A) = \mathcal{M}(O, A) \setminus \mathcal{SL}(A, \neg K)$

Proof. Analogous to the proof of 4 in the obvious way. □

Proof of Lemma 1 (p. 21).

Let $\mathcal{V}$ be a vocabulary, $K, K' \in \mathcal{V}$ concept symbols, $a \in \mathcal{V}$ a constant. Let O be an ontology, and A a sequence of literals with finite $\tilde{A}$, $\mathcal{V}(O \cup \tilde{A}) \subseteq \mathcal{V} \setminus \{K'\}$, and $\alpha \in \tilde{A}$.

1. $\mathcal{CL}(O \cup \{\alpha\}, A) \subseteq \mathcal{CL}(O, A) \setminus \{\alpha\}$ results from Definition 8.3 and Lemma 3.2.
2. If $K(a) \in \tilde{A}$, then
   $\mathcal{CL}(O_{[K/K']} \cup \{K(a), K' \sqsubseteq K\}, A) \subseteq \mathcal{CL}(O, A) \setminus \{K(b) \mid \text{constant } b \in \mathcal{V}\}$
   is a direct consequence of Lemma 4.6 and Definition 8.3.
3. If $\neg K(a) \in \tilde{A}$, then
   $\mathcal{CL}(O_{[K/K']} \cup \{\neg K(a), K \sqsubseteq K'\}, A) \subseteq \mathcal{CL}(O, A) \setminus \{\neg K(b) \mid \text{constant } b \in \mathcal{V}\}$
   is a direct consequence of Lemma 5.6 and Definition 8.3. □

Proof of Lemma 2 (p. 22). Let O be an ontology, A a sequence of literals with finite $\tilde{A}$, $\alpha \in \tilde{A}$, and A^n a finite prefix of A.

1. $\mathcal{CL}(O \otimes_2 \alpha, A) \subseteq \mathcal{CL}(O, A) \setminus \{\alpha\}$
   If $O \cup \alpha$ is consistent, then $O \otimes_2 \alpha = O \cup \{\alpha\}$ and the assertion derives from Lemma 1.1.

   If $O \cup \alpha$ is inconsistent, then the assertion derives from Lemma 1.2 and 1.3.

2. $\mathcal{CL}(O \otimes_2 A^n, A) \subseteq \mathcal{CL}(O, A) \setminus \tilde{A}^n$
   If $O \otimes_2 A^n \cup \mathsf{una}(A)$ is inconsistent, then $\mathcal{CL}(O \otimes_2 A^n, A) = \emptyset$ according to Observation 6.4 and Definition 8.3.

   If $O \otimes_2 A^n \cup \mathsf{una}(A)$ is consistent, then the assertion derives from Observation 5.4 and part 1 of this lemma by induction on n.

□

Proof of Corollary 1 (p. 22). Let $\mathcal{V}$ be a vocabulary, $K \in \mathcal{V}$ a concept symbol, $a, c \in \mathcal{V}$ constants, $A = \langle \alpha_i \rangle_{i \in \{1,\ldots,n\}}$ a finite sequence of literals over $\mathcal{V}$, O an ontology over $\mathcal{V}$.

1. Let $O' = (O \otimes_2 K(a)) \otimes_2 A$. Assume $O \cup \{K(a)\}$ is inconsistent and $O' \cup \mathsf{una}(\tilde{A} \cup \{K(a), K(c)\})$ is consistent.

   Let $A' = \langle \alpha_i \rangle_{i \in \{1,\ldots,n+2\}}$ be the extension of A with $\alpha_{n+1} = K(c), \alpha_{n+2} = K(a)$. Thus, $A = (A')^n$. According to Definition 2, $O_2 = (O \otimes_2 K(a)) = O_{[K/K']} \cup \{K(a), K' \sqsubseteq K\}$ (for some K' not occurring in O) and according to Lemma 1.2, $\mathcal{CL}(O_2, A') \subseteq \mathcal{CL}(O, A') \setminus \{K(b) \mid \text{constant } b \in \mathcal{V}\}$. Therefore, $K(c) \notin \mathcal{CL}(O_2, A')$.

   By assumption, $O' = O_2 \otimes_2 A$. According to Lemma 2.2 we get $\mathcal{CL}(O', A') \subseteq \mathcal{CL}(O_2, A') \setminus \tilde{A}$. Therefore $K(c) \notin \mathcal{CL}(O', A')$. According to Definition 8.3, $\{K(c)\} \notin \mathcal{M}(O', A')$. As $O' \cup \mathsf{una}(\tilde{A} \cup \{K(a), K(c)\})$ is consistent, $\emptyset \notin \mathcal{C}(O', A')$. Therefore (by Definition 8.2), $\{K(c)\} \notin \mathcal{C}(O', A')$, which means that $O' \cup \{K(c)\}$ is consistent.

2. Corresponding to part 1 of this proof considering $\neg K(a)$ and $\neg K(c)$ instead of $K(a)$ and $K(c)$.

3. Let $1 \leq j \leq k \leq n$, such that $\alpha_j = K(a)$ and $\alpha_k = K(c)$ or $\alpha_j = \neg K(a)$ and $\alpha_k = \neg K(c)$. Let $O' = O \otimes_2 A$. Assume $O' \cup \mathsf{una}(A)$ is consistent and $O \otimes_2 A^{j-1} \cup \{\alpha_j\}$ is inconsistent.

   As $O' \cup \mathsf{una}(A)$ is consistent, $O \otimes_2 A^{k-1} \cup \mathsf{una}(A)$ is consistent as well according to Observation 5.4. Parts 1 and 2 of this corollary yield that $O \otimes_2 A^{k-1} \cup \{\alpha_k\}$ is consistent. □

Proof of Corollary 2 (p. 22).
Derives from Lemma 2.2, Definition 1.3, Observations 6.3 and 6.4. □

Proof of Theorem 1 (p. 23).

Let O be a consistent ontology, A a sequence of literals with finite $\tilde{A}$ such that for every $n \in \mathbb{N}$ the set $O \otimes_2 A^n \cup \mathsf{una}(A^n)$ is consistent.

Let A^k be a prefix of A with $\tilde{A} = \tilde{A}^k$. Let $i \geq k$ be such that all literals of $\tilde{A}$ that occur at least once after k occur between k and i. Because of Lemma 2.2 and $\tilde{A} = \tilde{A}^k$, $\mathcal{CL}(O \otimes_2 A^k, A) = \emptyset$. According to Observation 6.4 this means that $O \otimes_2 A^k \cup \tilde{A}$ is consistent. Thus, after step k only expansions can occur. Because of Observation 1.4, $O \otimes_2 A^i = O \otimes_2 A^k \cup \{\alpha_{k+1}, \ldots, \alpha_i\}$. Because of

the choice of i it identifies an upper bound for the step at which the sequence of ontologies stabilizes. □

Definition 10. *For a set of formulae M and a concept symbol K let $M_K = \{\beta \in M \mid \beta \text{ contains } K\}$ be* the subset of formulae of M that syntactically contain K.

Observation 7. *Let O be an ontology over vocabulary $\mathcal{V}_c \cup \mathcal{V}_p$ with $\mathcal{V}_c \cap \mathcal{V}_p = \emptyset$ and A a finite sequence of literals over $\mathcal{V}_c$, such that $(O \otimes_2 A) \cup \mathsf{una}(A)$ is consistent. Let $K \in \mathcal{V}_c$ be any concept symbol that has been reinterpreted during the integration of A into O using the weak revision operator of type 2.*

1. *There is a concept symbol $K^\# \in \mathcal{V}_p$, such that*
   - $(O \otimes_2 A)_K \subseteq \{K \sqsubseteq K^\#\} \cup \tilde{A}$ *or*
   - $(O \otimes_2 A)_K \subseteq \{K^\# \sqsubseteq K\} \cup \tilde{A}$.
2. *If $K \in \mathcal{V}_c$ has been reinterpreted twice, then there is another concept symbol $K' \in \mathcal{V}_p \setminus \{K^\#\}$, such that*
   - $(O \otimes_2 A)_{K^\#} \subseteq \{K \sqsubseteq K^\#, K' \sqsubseteq K^\#\} \cup \tilde{A}_{[K/K^\#]}$ *or*
   - $(O \otimes_2 A)_{K^\#} \subseteq \{K^\# \sqsubseteq K, K^\# \sqsubseteq K'\} \cup \tilde{A}_{[K/K^\#]}$

Proof.

1. Consequence of Definition 2.
2. Consequence of Definition 2 in combination with Corollary 1. □

□

Proof of Theorem 2 (p. 24).

Let $K \in \mathcal{V}_c$ be any concept symbol that has been reinterpreted twice during the integration of A into O using the weak revision operator of type 2. Let $K'' \in \mathcal{V}_p$ be the concept symbol, such that (according to Observation 7.1) $(O \otimes_2 A)_K \subseteq \{K \sqsubseteq K'', K'' \sqsubseteq K\} \cup \tilde{A}$.

Since $\mathcal{I} \models (O \otimes_2 A) \cup \mathsf{una}(A)$ and $\mathcal{J}$ is the same as $\mathcal{I}$ for all symbols except K and K'', $\mathcal{J} \models \mathsf{una}(A)$ and $\mathcal{J} \models \beta$ for any $\beta \in O \otimes_2 A$ that does not contain K or K''.

$\mathcal{J} \models \tilde{A}_K$ derives from $K^\mathcal{J} = \mathcal{I}_K^A$, $\{a^\mathcal{I} \mid K(a) \in \tilde{A}\} \subseteq \mathcal{I}_K^A$, and $\{a^\mathcal{I} \mid \neg K(a) \in \tilde{A}\} \cap \mathcal{I}_K^A = \emptyset$.

According to Observation 7.1, $(O \otimes_2 A)_K \subseteq \{K \sqsubseteq K''\} \cup \tilde{A}$ or $(O \otimes_2 A)_K \subseteq \{K'' \sqsubseteq K\} \cup \tilde{A}$. The construction yields $\mathcal{J} \models (O \otimes_2 A)_K$ in either case.

Since K has been reinterpreted twice (according to Observation 7.2) there is a concept symbol K' such that, $(O \otimes_2 A)_{K''} \subseteq \{K \sqsubseteq K'', K' \sqsubseteq K''\} \cup \tilde{A}_{[K/K'']}$ or $(O \otimes_2 A)_{K''} \subseteq \{K'' \sqsubseteq K, K'' \sqsubseteq K'\} \cup \tilde{A}_{[K/K'']}$. Since $\mathcal{I} \models (O \otimes_2 A)_{K''}$, the construction also guarantees that $\mathcal{J} \models (O \otimes_2 A)_{K''}$ in either case. □

Lemma 6. *Let $\mathcal{ML}(\mathcal{V})$ be a propositionally complete description logic without nominals, i.e. a description logic that provides concept constructors for intersection, union, and negation, but does not provide any concept constructor based on constants. Let the vocabulary $\mathcal{V}$ provide constants and atomic concepts but no role symbols. Correspondingly, no concept description based on $\mathcal{ML}(\mathcal{V})$ does employ a concept constructor based on roles or constants.*

Let O be a description-logic ontology based on $\mathcal{ML}(\mathcal{V})$ with finite ABox. Let $\mathsf{expl}_O(x) =_{def} \sqcap\{D \mid D(x) \in O\}$ be a conjunction of all concepts explicitly asserted for x in O.

Then for any concept description C based on $\mathcal{ML}(\mathcal{V})$, constant $x \in \mathcal{V}$, and sequence A of literals over $\mathcal{V}$,

1. *$O \models C(x)$ iff $O \models \mathsf{expl}_O(x) \sqsubseteq C$.*
2. *If $\mathsf{msc}_O(x)$ exists, then $O \models \mathsf{expl}_O(x) \doteq \mathsf{msc}_O(x)$.*
3. *$O \cup \mathsf{una}(A)$ is consistent iff O is consistent and $\mathsf{una}(A)$ is consistent.*

Proof.

1. The definition of expl yields $O \models \mathsf{expl}_O(x)(x)$. Therefore, if $O \models \mathsf{expl}_O(x) \sqsubseteq C$, then also $O \models C(x)$.

   Now assume $O \models C(x)$, an interpretation $\mathcal{I}$, such that $\mathcal{I} \models O$, and $d \in \mathsf{expl}_O(x)^{\mathcal{I}}$. We will show that $d \in C^{\mathcal{I}}$.

   Let $\mathcal{J}$ be the modification of $\mathcal{I}$, such that $\mathcal{J}(x) = d$. We first want to show that $\mathcal{J} \models O$. As $\mathcal{J}$ differs from $\mathcal{I}$ only regarding x, we just have to consider those $\alpha \in O$ that use x. According to the restrictions assumed on the syntactic structure of O, the constant x can occur in O only in ABox axioms of the form $D(x)$, where D is a description that does not contain x. Let $\alpha = D(x) \in O$. Then $\models \mathsf{expl}_O(x) \sqsubseteq D$ by definition of expl. $\mathcal{J} \models D(x)$ iff $x^{\mathcal{J}} = d \in D^{\mathcal{J}} = D^{\mathcal{I}}$. As $d \in \mathsf{expl}_O(x)^{\mathcal{I}} \subseteq D^{\mathcal{I}}$, this is the case.

   Thus, $\mathcal{J} \models O$. As $O \models C(x)$, this means $\mathcal{J} \models C(x)$, i.e., $d = x^{\mathcal{J}} \in C^{\mathcal{J}} = C^{\mathcal{I}}$, since x does not occur in C. Consequently $O \models C(x)$ iff $O \models \mathsf{expl}_O(x) \sqsubseteq C$.

2. As $O \models \mathsf{msc}_O(x)(x)$, we get $O \models \mathsf{expl}_O(x) \sqsubseteq \mathsf{msc}_O(x)$ from part 1 of this lemma. As $O \models \mathsf{expl}_O(x)(x)$, we derive $\models \mathsf{msc}_O(x) \sqsubseteq \mathsf{expl}_O(x)$ from the definition of msc. Therefore $O \models \mathsf{expl}_O(x) \doteq \mathsf{msc}_O(x)$.

3. It is obvious that both O and $\mathsf{una}(A)$ are consistent, if $O \cup \mathsf{una}(A)$ is consistent.

   Assume that both O and $\mathsf{una}(A)$ are consistent and let $\mathcal{J}$ be an interpretation, such that $\mathcal{J} \models O$. Let $\mathcal{I}$ be the Herbrand interpretation defined as follows: $\Delta^{\mathcal{I}}$ is the set of all constants of $\mathcal{V}$, for all constants x: $x^{\mathcal{I}} = x$ and for all concept symbols K: $K^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid x^{\mathcal{J}} \in K^{\mathcal{J}}\}$.

As $\mathsf{una}(A)$ is consistent and contains statements of the form $x \neq y$ only, $\mathcal{I} \models \mathsf{una}(A)$ is a direct consequence of the definition of $\mathcal{I}$. By induction on the formation of complex concepts, one can show that for all concepts C based on $\mathcal{ML}(\mathcal{V})$, the condition $C^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid x^{\mathcal{J}} \in C^{\mathcal{J}}\}$ holds.

On this basis, the proof of $\mathcal{I} \models O$ is straight forward.

□

Corollary 4. *Let $\mathcal{V}$ be a vocabulary without role symbols and O, O_1, O_2 be ontologies based on $\mathcal{ML}(\mathcal{V})$, and let α be a literal over $\mathcal{V}$. Then*

1. $O \oplus_i^{\mathsf{expl}} \alpha \equiv O \odot_i \alpha$.
2. *If $O_1 \equiv O_2$, then $O_1 \oplus_i^{\mathsf{expl}} \alpha \equiv O_2 \oplus_i^{\mathsf{expl}} \alpha$.*

Proof of Theorem 3 (p. 25).

Let $\mathcal{V}_c$ and $\mathcal{V}_p$ be vocabularies such that $\mathcal{V}_c \cap \mathcal{V}_p = \emptyset$, $a, b, c, d \in \mathcal{V}_c$ be constants, and $B, C, D, E \in \mathcal{V}_c$ be concept symbols. Let the ontology O, the finite sequence A, and the infinite sequence A' (the infinite repetition of A) be given by

$$\begin{aligned}
O &= \{\neg B(a), \neg C(a), \neg D(a), \neg E(a), \neg B(b), \neg C(b), \neg B(c), \neg C(c), \neg D(c), \neg E(c),\\
&\quad \neg B(d), \neg C(d), \neg D(d), \neg E(d)\}\\
A &= \langle \alpha_i \rangle_{i \in \{1,\ldots,16\}} = \langle C(a), B(a), B(b), B(d), E(b), D(b), D(a), D(c), \neg B(c),\\
&\quad \neg C(c), \neg C(d), \neg C(b), \neg D(d), \neg E(d), \neg E(a), \neg E(c)\rangle\\
A' &= \langle \alpha_i \rangle_{i \in \mathbb{N}}, \text{ with } \alpha_i = \alpha_{i+16} \text{ for all } i \in \mathbb{N}
\end{aligned}$$

In this case, and for any literal α with $\mathcal{V}(\alpha) \subseteq \mathcal{V}_c$

$$\begin{aligned}
O \oplus_2^{\mathsf{expl}} A \models \alpha \quad &\text{iff} \quad O \models \alpha\\
O \odot_2 A \models \alpha \quad &\text{iff} \quad O \models \alpha
\end{aligned}$$

and

$$\begin{aligned}
\mathcal{M}(O \odot_2 A, A') &=\\
\mathcal{M}(O \oplus_2^{\mathsf{expl}} A, A') &= \mathcal{M}(O, A) = \mathcal{M}(O, A')\\
&= \{\{B(a)\}, \{C(a)\}, \{D(a)\}, \{B(b)\}, \{D(c)\}, \{B(d)\}\}
\end{aligned}$$

The (infinite) sequence A' is the systematic repetition of the sequence A. After each round of 16 steps, the same set of literals over $\mathcal{V}_c$ are consequences of the resulting ontology and the same set of conflicting literals is re-established. Therefore, the sequence of ontologies generated during the integration process does not stabilize.

As O and A conform to the restrictions described in Corollary 4, the sequence of ontologies generated by $\odot_2$ and $\oplus_2^{\mathsf{expl}}$ consist of equivalent ontologies. We spare the readers the lengthy listing of determining $O \oplus_2^{\mathsf{expl}} A$. Nevertheless, we discuss the basic steps to motivate the resulting structure.

The set of assertions concerning a in O is $\{\neg B(a), \neg C(a), \neg D(a), \neg E(a)\} \subseteq O$. Correspondingly, the first statement $C(a)$ from A conflicts with O and the integration result is

$O^1 = O_{[C/C']} \cup \{C(a), C' \sqsubseteq C, C \sqsubseteq C' \sqcup \neg B, C \sqsubseteq C' \sqcup \neg C', C \sqsubseteq C' \sqcup \neg D, C \sqsubseteq C' \sqcup \neg E\}$.

To show the content of the derived ontologies in a more compact form, we make use of the facts that an axiom like $X \sqsubseteq Y \sqcup Z$ is equivalent to the axiom $X \sqcap \neg Y \sqsubseteq Z$ and that the set of axioms $\{X \sqsubseteq Y, X \sqsubseteq Z\}$ is equivalent to $\{X \sqsubseteq Y \sqcap Z\}$. Correspondingly, $O^1 \equiv O_{[C/C']} \cup \{C(a), C' \sqsubseteq C, (C \sqcap \neg C') \sqsubseteq (\neg B \sqcap \neg D \sqcap \neg E)\}$.

The second statement of A is $B(a)$. This conflicts with O^1 as $\neg B(a) \in O^1$. The result of integrating $B(a)$ is
$O^2 \equiv O^1_{[B/B']} \cup \{B(a), B' \sqsubseteq B, (B \sqcap \neg B') \sqsubseteq (\neg C' \sqcap \neg D \sqcap \neg E \sqcap C)\}$
As $O^1_{[B/B']} \models (C \sqcap \neg C') \sqsubseteq (\neg D \sqcap \neg E)$
$O^2 \equiv O^1_{[B/B']} \cup \{B(a), B' \sqsubseteq B, (B \sqcap \neg B') \sqsubseteq (C \sqcap \neg C')\}$.

The axiom $(B \sqcap \neg B') \sqsubseteq (C \sqcap \neg C')$ expresses that the difference between B and B' is subsumed by the difference between C and C'. Consequently,
$O^2 \equiv O_{[C/C'][B/B']} \cup \{C(a), B(a), C' \sqsubseteq C, B' \sqsubseteq B, (C \sqcap \neg C') \sqsubseteq (\neg B' \sqcap \neg D \sqcap \neg E), (B \sqcap \neg B') \sqsubseteq (C \sqcap \neg C')\}$.

The third and fourth statements from the trigger sequence $(B(b), B(d))$ do not conflict with O^2 (resp. O^3) and
$O^4 = O^2 \cup \{B(b), B(d)\} \equiv O_{[C/C'][B/B']} \cup \{C(a), B(a), B(b), B(d), C' \sqsubseteq C,$
$B' \sqsubseteq B, (C \sqcap \neg C') \sqsubseteq (\neg B' \sqcap \neg D \sqcap \neg E), (B \sqcap \neg B') \sqsubseteq (C \sqcap \neg C')\}$.

All of the following steps follow these two patterns. Either there is no conflict (steps 3, 4, 7, 8, 11, 12, 15, 16) and a simple expansion occurs, or a conflict occurs (steps 1, 2, 5, 6, 9, 10, 13, 14) where the difference between the two readings is subsumed by the difference between the readings of other ambiguous terms disambiguated in an earlier step. This ordering of differences is established since no ontology derived during the integration sequence can prove that the individuals named by the constant can be distinguished based on the concepts involved.

The resulting ontology $O^{16} = O^{16}_A \cup O^{16}_T$ consists of an ABox O^{16}_A and a TBox O^{16}_T. The ABox consists of entries from O and A in their initial form or a variant derived by substitution. While all entries from O appear in O^{16}_A in the variant using simple primes (derived in the first reinterpretation of the concept symbol), the positive items from A appear in O^{16}_A in the variant using double primes (derived in the second reinterpretation of the concept symbol).
$O^{16}_A = O_{[C/C',B/B',D/D',E/E']} \cup \{C''(a), B''(a), B''(b), B''(d), E''(b), D''(b), D''(a), D''(c)\} \cup \{\neg B(c), \neg C(c), \neg C(d), \neg C(b), \neg D(d), \neg E(d), \neg E(a), \neg E(c)\}$

The resulting TBox $O^{16}_T = O^{16}_{T1} \cup O^{16}_{T2}$ can be decomposed into the part (O^{16}_{T1}) with the axioms relating only two readings of one term and the part (O^{16}_{T2}) with the axioms relating the two readings and another concept.
$O^{16}_{T1} = \{B' \sqsubseteq B'', B \sqsubseteq B'', C' \sqsubseteq C'', C \sqsubseteq C'', D' \sqsubseteq D'', D \sqsubseteq D'', E' \sqsubseteq E'', E \sqsubseteq E''\}$.

The axioms of O^{16}_{T2} can be mapped to an equivalent set of axioms O^{16}_{T3} as described above.
$O^{16}_{T2} \equiv O^{16}_{T3} = \{(E'' \sqcap \neg E) \sqsubseteq (D'' \sqcap \neg D), (D'' \sqcap \neg D) \sqsubseteq (C'' \sqcap \neg C),$
$(C'' \sqcap \neg C) \sqsubseteq (B'' \sqcap \neg B), (B'' \sqcap \neg B) \sqsubseteq (D'' \sqcap \neg D'), (D'' \sqcap \neg D') \sqsubseteq (E'' \sqcap \neg E'),$

$(E'' \sqcap \neg E') \sqsubseteq (B'' \sqcap \neg B'), (B'' \sqcap \neg B') \sqsubseteq (C'' \sqcap \neg C'), (C'' \sqcap \neg C') \sqsubseteq (\neg B' \sqcap \neg E' \sqcap \neg D')\}$

Let for each $X \in \{B, C, D, E\}, X^*$ stand for $\neg X \sqcap \neg X' \sqcap X''$ and $Y = B^* \sqcap C^* \sqcap D^* \sqcap E^*$. Then $O^{16}_{T2} \models (E'' \sqcap \neg E) \sqsubseteq Y$. As $\neg D'(a), D''(a), \neg E(a), \neg D'(c), D''(c), \neg E(c) \in O^{16}_A$, we can derive $O^{16} \models Y(a)$ and $O^{16} \models Y(c)$. As $\neg B'(d), B''(d), \neg C(d), \neg E(d) \in O^{16}_A$, we can also derive $O^{16} \models Y(d)$. As $\neg B'(b), B''(b), \neg C(b) \in O^{16}_A$, we can derive $O^{16} \models (E'' \sqcap \neg E' \sqcap D'' \sqcap \neg D' \sqcap B^* \sqcap C^*)(b)$, but $O^{16} \cup \{D(b)\} \cup \mathsf{una}(A) \not\models \neg E(b)$, $O^{16} \not\models E(b)$, and $O^{16} \not\models D(b)$.

This proves that for any literal α with $\mathcal{V}(\alpha) \subseteq \mathcal{V}_c$

$$O \oplus_2^{\mathsf{expl}} A \models \alpha \quad \text{iff} \quad O \models \alpha$$

and

$$\mathcal{M}(O \oplus_2^{\mathsf{expl}} A, A') \quad = \quad \mathcal{M}(O, A')$$

As $\mathcal{V}(O) \subseteq \mathcal{V}_c$ and O contains only literals, this means that $O^{16} \models O$. The difference between O and O^{16} derives from the richer private vocabulary used by O^{16}. However, as the constants used in A are not distinguishable by O^{16} regarding this vocabulary, further iterations of integrating A will yield exactly the same sequence of conflicts and similar sequences of solution.

Appendix II: Example

Example 5. *An agent (receiver) using an ontology O_R wants to buy a cheap book on thermodynamics in an online bookshop (sender) that uses the ontology O_S.*

In this case, the receiver does not have any information on the announced books beforehand but just general terminological specifications. Both ontologies agree regarding notions such as costs-less-than-n-Euros*. However, they disagree on the specification of Cheap, i.e., in their judgements on value for money.*

A sequence A of literals stemming from the sender is integrated into the receiver's ontology using $\oplus_2^{\mathsf{expl}}$. The sender gives details on the type and price of the books (th_1 is a hardcover book that costs between 5 and 8 Euros, th_2 is a booklet with a price between 3 and 5 Euros) as well as its own value-for-money

judgement (th_1 is cheap, but th_2 is not cheap).

$$
\begin{aligned}
O_R &= \{\mathit{Cheap} \doteqdot \mathit{CostLt\_5}, \\
&\quad \mathit{CostLt\_3} \sqsubseteq \mathit{CostLt\_5}, \mathit{CostLt\_5} \sqsubseteq \mathit{CostLt\_8}\} \\
O_S &= \{\mathit{Cheap} \doteqdot (\mathit{CostLt\_5} \sqcap \mathit{SoftC}) \sqcup (\mathit{CostLt\_8} \sqcap \mathit{HardC}) \sqcup \\
&\quad (\mathit{CostLt\_3} \sqcap \mathit{Booklet}), \\
&\quad \mathit{CostLt\_3} \sqsubseteq \mathit{CostLt\_5}, \mathit{CostLt\_5} \sqsubseteq \mathit{CostLt\_8}, \\
&\quad \mathit{SoftC} \sqsubseteq \neg\mathit{HardC}, \mathit{Booklet} \doteqdot \neg(\mathit{SoftC} \sqcup \mathit{HardC}), \\
&\quad \mathit{HardC}(th_1), \neg\mathit{CostLt\_5}(th_1), \mathit{CostLt\_8}(th_1), \\
&\quad \mathit{Booklet}(th_2), \neg\mathit{CostLt\_3}(th_2), \mathit{CostLt\_5}(th_2)\} \\
A &= \langle \mathit{HardC}(th_1), \neg\mathit{CostLt\_5}(th_1), \mathit{CostLt\_8}(th_1), \\
&\quad \mathit{Booklet}(th_2), \neg\mathit{CostLt\_3}(th_2), \mathit{CostLt\_5}(th_2), \\
&\quad \mathit{Cheap}(th_1), \neg\mathit{Cheap}(th_2)\rangle
\end{aligned}
$$

Integrating this sequence using the operator $\oplus_2^{\mathsf{expl}}$ needs two reinterpretation steps involving the concept symbol Cheap.

$$
\begin{aligned}
O_R \oplus_2^{\mathsf{expl}} A &= \{\mathit{Cheap}' \doteqdot \mathit{CostLt\_5}, \\
&\quad \mathit{CostLt\_3} \sqsubseteq \mathit{CostLt\_5}, \mathit{CostLt\_5} \sqsubseteq \mathit{CostLt\_8}, \\
&\quad \mathit{HardC}(th_1), \neg\mathit{CostLt\_5}(th_1), \mathit{CostLt\_8}(th_1), \\
&\quad \mathit{Booklet}(th_2), \neg\mathit{CostLt\_3}(th_2), \mathit{CostLt\_5}(th_2), \\
&\quad \mathit{Cheap}''(th_1), \mathit{Cheap}' \sqsubseteq \mathit{Cheap}'', \mathit{Cheap}'' \sqsubseteq \mathit{Cheap}' \sqcup \neg\mathit{CostLt\_5}, \\
&\quad \mathit{Cheap}'' \sqsubseteq \mathit{Cheap}' \sqcup \mathit{HardC}, \mathit{Cheap}'' \sqsubseteq \mathit{Cheap}' \sqcup \mathit{CostLt\_8}, \\
&\quad \neg\mathit{Cheap}(th_2), \mathit{Cheap} \sqsubseteq \mathit{Cheap}'', \mathit{Cheap}'' \sqsubseteq \mathit{Cheap} \sqcup \mathit{CostLt\_5}, \\
&\quad \mathit{Cheap}'' \sqsubseteq \mathit{Cheap} \sqcup \mathit{Booklet}, \mathit{Cheap}'' \sqsubseteq \mathit{Cheap} \sqcup \neg\mathit{CostLt\_3}\}
\end{aligned}
$$

The resulting ontology $O_R \oplus_2^{\mathsf{expl}} A$ allows the derivation of $\neg\mathit{Cheap}(th_2)$ and of $\mathit{Cheap}(th_1)$ (using $\mathit{Cheap}''(th_1)$, $\neg\mathit{CostLt\_5}(th_1)$, and $\mathit{Cheap}'' \sqsubseteq \mathit{Cheap} \sqcup \mathit{CostLt\_5}$). Thus, it agrees with O_S regarding the value-for-money judgements. Furthermore, the additional bridging axioms introduced to relate the different readings of the term Cheap result in an approximation of the meaning of the common symbol within the sender's ontology, as both O_S and $O_R \oplus_2^{\mathsf{expl}} A$ have the following statements as consequences: books that cost less than 3 Euros are cheap ($\mathit{CostLt\_3} \sqsubseteq \mathit{Cheap}$), cheap books cost less than 8 Euros ($\mathit{Cheap} \sqsubseteq \mathit{CostLt\_8}$), books under 5 Euros that are not booklets are cheap ($\mathit{CostLt\_5} \sqcap \neg\mathit{Booklet} \sqsubseteq \mathit{Cheap}$), and cheap books that are not hardcover books cost less then 5 Euros $\mathit{Cheap} \sqcap \neg\mathit{HardC} \sqsubseteq \mathit{CostLt\_5}$.

References

[1] Carlos E. Alchourrón, Peter Gärdenfors, and David Makinson. On the logic of theory change: partial meet contraction and revision functions.

Journal of Symbolic Logic, 50:510–530, 1985.

[2] Franz Baader. Description logic terminology. In F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P. F. Patel-Schneider, editors, *The Description Logic Handbook*, pages 495–505. Cambridge University Press, 2002.

[3] Mukesh Dalal. Investigations into a theory of knowledge base revision: Preliminary report. In *Proceedings of the Seventh National Conference on Artificial Intelligence (AAAI-88)*, pages 475–479, St. Paul, Minnesota, August 1988. AAAI Press.

[4] James P. Delgrande, Didier Dubois, and Jérôme Lang. Iterated revision as prioritized merging. In P. Doherty, J. Mylopoulos, and C.A. Welty, editors, *Proceedings of the Tenth International Conference on Principles of Knowledge Representation and Reasoning (KR-2006)*, pages 210–220. AAAI Press, 2006.

[5] James P. Delgrande and Torsten Schaub. A consistency-based approach for belief change. *Artificial Intelligence*, 151(1–2):1–41, 2003.

[6] Giorgos Flouris, Dimitris Manakanatas, Haridimos Kondylakis, Dimitris Plexousakis, and Grigoris Antoniou. Ontology change: classification and survey. *The Knowledge Engineering Review*, 23(2):117–152, 2008.

[7] Peter Gärdenfors. *Knowledge in Flux: Modeling the Dynamics of Epistemic States*. MIT Press, Bradford Books, Cambridge, MA, 1988.

[8] Peter Gärdenfors. Belief revision: A vade-mecum. In A. Pettorossi, editor, *Proceedings of the Third International Workshop on Meta-Programming in Logic (META-92)*, volume 649 of *Lecture Notes in Computer Science*, pages 1–10. Springer, 1992.

[9] Moritz Goeb, Peter Reiss, Bernhard Schiemann, and Ulf Schreiber. Dynamic T-box-handling in agent-agent-communication. In Ch. Beierle and G. Kern-Isberner, editors, *Dynamics of Knowledge and Belief. Proceedings of the Workshop at the 30th Annual German Conference on Artificial Intelligence (KI-2007)*, pages 100–117. Fernuniversität in Hagen, September 2007.

[10] Adam Grove. Two modellings for theory change. *Journal of Philosophical Logic*, 17:157–170, 1988.

[11] Sven Ove Hansson. *A Textbook of Belief Dynamics*. Kluwer Academic Publishers, 1999.

[12] Hirofumi Katsuno and Alberto Mendelzon. On the difference between updating a knowledge base and revising it. In J.F. Allen, R. Fikes, and E. Sandewall, editors, *Proceedings of the Second International Conference on Principles of Knowledge Representation and Reasoning (KR-1991)*, pages 387–394, San Mateo, California, 1991. Morgan Kaufmann.

[13] Kevin T. Kelly. Iterated belief revision, reliability, and inductive amnesia. *Erkenntnis*, 50:11–58, 1998.

[14] Kevin T. Kelly. The learning power of belief revision. In I. Gilboa, editor, *Proceedings of the Seventh Conference on Theoretical Aspects of Rationality and Knowledge (TARK-98)*, pages 111–124. Morgan Kaufmann, 1998.

[15] Ralf Küsters and Ralf Molitor. Computing most specific concepts in description logics with existential restrictions. LTCS-Report 00-05, LuFG Theoretical Computer Science, RWTH Aachen, Germany, 2000.

[16] Eric Martin and Daniel Osherson. Scientific discovery based on belief revision. *Journal of Symbolic Logic*, 62:1352–1370, 1997.

[17] Thomas Meyer, Kevin Lee, and Richard Booth. Knowledge integration for description logics. In M.M. Veloso and S. Kambhampati, editors, *Proceedings of the Twentieth National Conference on Artificial Intelligence (AAAI-2005)*, pages 645–650. AAAI Press / The MIT Press, 2005.

[18] Donald J. Monk. *Mathematical Logic*. Springer, 1976.

[19] Natalya Fridman Noy. Semantic integration: A survey of ontology-based approaches. *SIGMOD Record*, 33(4):65–70, 2004.

[20] Özgür L. Özçep. Ontology revision through concept contraction. In S. Artemov and R. Parikh, editors, *Proceedings of the Workshop on Rationality and Knowledge, 18th European Summerschool in Logic, Language, and Information, Universidad de Malaga*, pages 79–90, 2006.

[21] Özgür L. Özçep. Towards principles for ontology integration. In C. Eschenbach and M. Grüninger, editors, *Proceedings of the Fifth International Conference on Formal Ontology in Information Systems (FOIS-2008)*, pages 137–150, 2008.

[22] Guilin Qi, Weiru Liu, and David A. Bell. A revision-based approach to handling inconsistency in description logics. In *Proceedings of the 11th International Workshop on Non-Monotonic Reasoning (NMR06)*, pages 124–132, 2006.

[23] Renata Wassermann and Eduardo Fermé. A note on prototype revision, 1999. *Spinning Ideas. (Electronic Essays dedicated to Peter Gärdenfors on his 50th Birthday).* http://www.lucs.lu.se/spinning/.

[24] Dongmo Zhang and Norman Y. Foo. Convergency of learning process. In B. McKay and J.K. Slaney, editors, *Proceedings of the Fifteenth Australian Joint Conference on Artificial Intelligence*, volume 2667 of *Lecture Notes in Computer Science*, pages 547–556. Springer, 2002.