Wright State University

CORE Scholar

Computer Science and Engineering Faculty Publications

Computer Science & Engineering

6-1-2011

Local Closed-World Reasoning with Description Logics under the Well-Founded Semantics

Matthias Knorr

Jose Julio Alferes

Pascal Hitzler
pascal.hitzler@wright.edu

Follow this and additional works at: https://corescholar.libraries.wright.edu/cse

Part of the Bioinformatics Commons, Communication Technology and New Media Commons, Databases and Information Systems Commons, OS and Networks Commons, and the Science and Technology Studies Commons

Local Closed World Reasoning with Description Logics under the Well-Founded Semantics

Matthias Knorr[a,*], José Júlio Alferes[a,1], Pascal Hitzler[b,2]

[a]*CENTRIA, Departamento de Informática, FCT/UNL, Quinta da Torre, 2829–516 Caparica, Portugal*
[b]*Kno.e.sis Center, Dept of Computer Science & Engineering, Wright State University, 3640 Colonel Glenn Hwy, Dayton, OH 45435, USA*

Abstract

An important question for the upcoming Semantic Web is how to best combine open world ontology languages, such as the OWL-based ones, with closed world rule-based languages. One of the most mature proposals for this combination is known as hybrid MKNF knowledge bases [52], and it is based on an adaptation of the Stable Model Semantics to knowledge bases consisting of ontology axioms and rules. In this paper we propose a well-founded semantics for nondisjunctive hybrid MKNF knowledge bases that promises to provide better efficiency of reasoning, and that is compatible with both the OWL-based semantics and the traditional Well-Founded Semantics for logic programs. Moreover, our proposal allows for the detection of inconsistencies, possibly occurring in tightly integrated ontology axioms and rules, with only little additional effort. We also identify tractable fragments of the resulting language.

Key words: Knowledge Representation, Description Logics and Ontologies, Non-monotonic Reasoning, Logic Programming, Semantic Web

1. Introduction and Motivation

The Semantic Web has recently become a major source of inspiration for Knowledge Representation and Reasoning (KRR). The underlying idea of the Semantic Web is to use KRR techniques to enhance data in the World Wide Web with knowledge bases, making this data available for processing by intelligent systems. Semantic Web has become a mature field of research, and industrial applications of Semantic Web technologies are on the way. Semantic Web is a topic that is clearly here to stay.

---

[*]Corresponding author, mknorr@di.fct.unl.pt, Tel. (+351) 21 294 8536, FAX (+351) 21 294 8541
[1]jja@di.fct.unl.pt
[2]pascal@pascal-hitzler.de

However, we believe that the KRR formalisms used in the Semantic Web are not adequate for several application areas within the Semantic Web. We therefore motivate in this section why KRR formalisms combining open and closed world reasoning are sometimes preferable over fragments of classical first-order logics, and we present application scenarios illustrating the requirement for that combination. Then, we show the limitations of already existing approaches, and we state the main contributions of our proposal.

1.1. Open vs. Closed World Reasoning

The most prominent expressive KRR approach employed in Semantic Web research is based on Description Logics [3, 27]. In particular, the Web Ontology Language OWL [26] is based on the description logic $\mathcal{SROIQ}(D)$, and it is a recommended standard by the World Wide Web Consortium (W3C) for modelling Semantic Web knowledge bases (commonly known as *ontologies*).

Description Logics (DLs), in turn, bear a first-order predicate logic semantics. DLs are monotonic and adhere to the Open World Assumption (OWA). This means that (negative) conclusions drawn from a knowledge base must be based on information explicitly present in the knowledge base. Being based on classical first-order logic, DLs differ from other KRR formalisms, e.g., those studied in the nonmonotonic reasoning field, that usually apply the Closed World Assumption (CWA). Under that assumption, all non-provable expressions are assumed to be false.

The decision to rely on the OWA appears to be a natural one in light of the envisioned applications related to the World Wide Web: the absence of a piece of knowledge should not generally be taken as an indication that this piece of knowledge is false. However, there are also application scenarios where the CWA, or at least the partial closure of the knowledge base, is a more natural choice. Such scenarios can occur, e.g., if ontology-based reasoning is done in conjunction with data stored in a database. Database data is usually considered to be complete, and so statements not in the database should be taken as false.

As an example where a combination of open and closed world assumption is desired, consider the large case study described in [54], containing millions of assertions about matching patient records with clinical trials criteria. In this clinical domain, open world reasoning is needed in radiology and laboratory data. For example, unless a lab test asserts a negative finding, no arbitrary assumptions about the results of the test can be made. That is, we can only be certain that some patient does not have a specific kind of cancer if the corresponding test has a negative result. However, as observed in [54], the closed world assumption can and should be used with data about medical treatment to infer that a patient is not on a medication unless otherwise stated. The work of [54] applies only open world reasoning but claims that the usage of closed world reasoning in data about medical treatment would be highly desirable and that the combination of OWA and CWA is an open problem in their work. Similar situations occur, e.g., in matchmaking using Semantic Web Services (cf. [22]), and in other scenarios in the medical domain.

In fact, life sciences, including medicine, is a prominently studied application area for OWL. Several large-scale ontologies have been developed in this area that are being used in practice, such as GALEN[3] and SNOMED.[4] These ontologies provide unified medical terminologies for the management and exchange of clinical information. The knowledge bases typically consist of information about anatomy, diseases, procedures, drugs, etc., and their applications range from medical record management to diagnostics support. SNOMED is used, for example, in the case study described above. All of these applications use ontology reasoning based on the OWA. But it is not difficult to foresee situations in these domains that would benefit from local closed world reasoning. Consider, for example, that such a medical knowledge base is used to decide whether a certain anaesthetic should be applied before surgery, depending on whether the patient is allergic to the anaesthetic or not. This information might not be available, and it should be modelled using the CWA: in an emergency situation, unless we know explicitly about an allergy, we assume that the patient is not allergic, and we apply the anaesthetic. Other examples can be found if we were to model exceptions in anatomical terminology; e.g., the existence of persons whose heart is actually on the right-hand side. Exception modelling is not directly possible in classical first-order logic (this is a problem usually known in Artificial Intelligence as the *specification problem*) and so also not possible in OWL using only the OWA.

All of these examples demonstrate why application developers frequently voice that it would be favourable to have *local* closed world modelling as an additional feature for ontology-based systems. More precisely, it would be desirable to have a KRR formalism that allows us to interpret some parts of the knowledge base under the CWA, and others under the OWA. Such capabilities would considerably enhance the usability of OWL.

1.2. Combining Rules and Ontologies

Ontologies are a standard OWA formalism while rules usually apply the CWA. A combination of ontologies and rules would clearly yield a combination of the OWA and the CWA. However, combining rules and ontologies is a nontrivial task, since a naive combination of ontologies and OWA-based rules is already undecidable [32]. In fact, formalisms for rules and formalisms for ontologies differ substantially on how decidability is achieved. For ontologies, decidability is achieved by specific syntactic restrictions on the available first-order predicates, and by restricting the way these predicates can be related. Rules do not have such syntactic restrictions, but are usually limited in their applicability to the finitely many different objects explicitly appearing in the knowledge base. An immediate effect of these differences is that some expressive features of one of the approaches are not available in the other approach. Namely, rules

---

[3]http://www.opengalen.org/
[4]http://www.ihtsdo.org/snomed-ct/

make it possible to express: non-treeshape-like relationships [63][5] such as "an uncle is the brother of one's father"; integrity constraints [57] to state, e.g., that a certain piece of information is explicitly present in the database; and closed world reasoning and specification of exceptions, as discussed above. Ontologies, on the contrary, make it possible to express open world reasoning, reason with unbounded or infinite domains, and they are thus well-suited to represent many types of incomplete information and schema knowledge. For example, in rule-based formalisms one typically cannot say that "every person has a father and a mother who are both persons" without listing all the parents explicitly. Our stance is that a combination of rules and ontologies is not only of interest for current applications in the web, but also as a highly sophisticated means of knowledge representation in general.

As argued in [52], a hybrid formalism combining rules and DL ontologies should satisfy certain criteria:

- Faithfulness: The integration of DLs and rules should preserve the semantics of both formalisms – that is, the semantics of a hybrid knowledge base in which one component is empty should be the same as the semantics of the other component. In other words, the addition of rules to a DL should not change the semantics of the DL and vice versa.

- Tightness: Rules should not be layered on top of a DL or vice versa; rather, the integration between a DL and rules should be tight in the sense that both the DL and the rule component should be able to contribute to the consequences of the other component.

- Flexibility: The hybrid formalism should be flexible and allow one to view the same predicate under both open and closed world interpretation. This allows us to enrich a DL with nonmonotonic consequences from rules, and to enrich the rules with the capabilities of ontology reasoning described by a DL.

- Decidability: To obtain a useful formalism that can be used in applications such as the Semantic Web, the hybrid formalism should be at least decidable, and preferably of low worst-case complexity.

1.3. Hybrid MKNF and Stable Models vs. Well-Founded Semantics

As shown in [52], among the various proposals for combining rules and ontologies (e.g. [8, 12, 14, 16, 25, 36, 39–42, 48, 52, 59]) the only one satisfying all four criteria above are Hybrid MKNF knowledge bases [52], which build on the logics of Minimal Knowledge and Negation as Failure (MKNF) [46]. A detailed discussion about the importance of Hybrid MKNF knowledge bases for modelling knowledge in the Semantic Web can be found in [31], and [22, 23]

---

[5]The DL SROIQ [30] also provides role composition axioms, which can be used to address some, but by no means all, use cases.

provide arguments for the usefulness of epistemic reasoning in the way it is done in MKNF logics. The proposal by Motik and Rosati [52] seamlessly integrates arbitrary decidable description logics with essentially (disjunctive) logic programming rules, making it possible to reason over a combination of monotonic open world knowledge and nonmonotonic closed world knowledge within a single (hybrid) framework.

Several reasoning algorithms are presented in [52] for Hybrid MKNF knowledge bases, and it is shown that the data complexity of reasoning within this framework is in many cases not higher than reasoning in the corresponding fragment of logic programming. Thus, adding an ontology to rules does not in general increase the data complexity when compared to rules alone. But the same cannot be said about adding rules to ontologies. E.g., we have at least a data complexity of coNP for a combination of normal logic programming rules with ontologies even if the data complexity of the Description Logics fragment is in the complexity class P. Indeed, although the approach of Hybrid MKNF knowledge bases is powerful, whenever we add rules with arbitrary nonmonotonic negation to an ontology, we in general loose tractability. Only a specific limited use of nonmonotonic negation, i.e. stratified rules, admits to maintain tractability (see [52]). However, we claim that robustness w.r.t. updates and the combination of different sources of information is an important property of a combination of rules and ontologies. Since it cannot be guarranteed that this property is maintained in such cases, we obtain a higher computational complexity in general.

The reason for that increase in the complexity lies in the fact that, as shown in [46], rules are interpreted in a similar way as in the Stable Model Semantics (SMS) [19] for logic programs, whose reasoning algorithms are NP-hard. So, if a semantics based on the SMS is adopted, then any improvements on the complexity of the combination of rules and ontologies are bound by NP-hardness.

The other major semantics for Logic Programming (LP) – the Well-Founded Semantics (WFS) [62] – seems to offer a solution. WFS is a three-valued semantics, where propositions can be 'true', 'false' or 'undefined' (while in SMS propositions can only be 'true' or 'false'), and WFS assigns a single model – the well-founded model – to every nondisjunctive logic program. The WFS is sound with respect to the SMS, in that whenever a proposition is true (resp. false) under the WFS, then it is also true (resp. false) in all stable models. Though the WFS is semantically weaker than SMS (in terms of the derivable true and false consequences), reasoning in the WFS has a lower computational complexity than in SMS – for normal programs the data complexity is P for the WFS instead of coNP for SMS [7]. Our stance is that the lower complexity bound makes WFS more promising than SMS as a basis for the semantics of hybrid knowledge bases. This is even more the case in application areas such as the one mentioned above [54] where huge amounts of data are involved.

Additionally, reasoning in SMS requires one to obtain the entire model of a knowledge base (just like [52] for combinations of rules and ontologies), while the WFS is amenable to top-down, query-driven reasoning, in which only the part of the knowledge base "relevant" to a specific query is accessed [6]. This

makes a WFS based approach all the more suitable for large scale applications.

1.4. Contribution

In this paper, we define a new semantics for Hybrid MKNF knowledge bases, restricted to nondisjunctive rules, that soundly approximates the semantics of [52] but is, in some important cases, in a strictly lower complexity class. In particular, when dealing with a tractable description logic, our combined approach remains tractable w.r.t. data complexity. We achieve this by extending the two-valued MKNF semantics from [52] to three truth values where each two-valued model from [52] corresponds to a total three-valued model of our approach (and vice versa) and where the least (w.r.t. derivable knowledge) three-valued MKNF model is the well-founded MKNF model. Our proposal straightforwardly satisfies the four criteria presented above for the combination of rules and ontologies. Moreover, the proposed semantics also guarantees the following properties:

- The well-founded MKNF model is faithful w.r.t. the two-valued MKNF models of [52], i.e. each query that is true (resp. false) in the well-founded MKNF model is also true (resp. false), in each two-valued MKNF model.

- Our proposal coincides with the original DL-semantics when no rules are present, and the original WFS of logic programs if the DL component is empty.

- If the knowledge base is consistent, then the approach is coherent in the sense of [55], i.e. if a formula φ is first-order false in the ontology, then the nonmonotonic interpretation of φ in the rules is enforced to be false as well.

- If the knowledge base is inconsistent, then our approach allows us to detect inconsistencies without any substantial additional computational effort apart from the consistency check of the ontology alone.

- The computational data complexity of our approach depends on the computational complexity of the applied DL, but if the considered DL is of polynomial data complexity, then the combination with rules remains polynomial.

1.5. Outline and Running Example

The rest of the paper is structured as follows. We first recall, in Section 2, preliminaries on Description Logics, the logics of Hybrid MKNF, and Hybrid MKNF knowledge bases. Then, in Section 3, we present our semantic framework that extends MKNF semantics to three truth values, based on which the well-founded MKNF model is defined. In Section 4 we show how to construct the well-founded MKNF model and also how the inconsistency detection works. A comparison to related work is presented in Section 5, before we conclude in Section 6.[6]

---

[6]A preliminary, and much shorter version of this work was presented in [37].

Throughout the paper, we make use of the following toy example.

Example 1. Consider an online store selling audio CDs. In order to attract more clients and raise sales, the store manager decided to introduce more sophisticated tools for recommending and searching CDs.

For that purpose, an ontology is used for structuring and maintaining the database of CDs. Each CD is associated with a unique identifier, a publisher, a release date, and the pieces of music the CD contains. Each piece of music has at least one track, and it is possible that a piece has several tracks (as is common for classical music). Additionally, each piece has a unique identifier and can be associated with the artist, composer, genre, origin of the piece.

Moreover, the system should be able to express guidelines for recommendations, either based on general criteria or based on customer specifications. For example, the store may want to automatically recommend to all customers CDs that are on offer or top sellers. Or some customer may want to get recommendations for CDs that he does not already own and that, according to some preference criteria, he probably likes. Whereas the first guideline can be represented in the ontology, the second one requires the closed world assumption (e.g., for inferring "by default", i.e., in the absence of evidence to the contrary, that the customer does not have the CD) and can be represented by a nonmonotonic rule. □

2. Preliminaries

In this section we recall preliminary notions that are needed in the rest of the paper. In detail, we present general notions for Description Logics, the logics of minimal knowledge and negation as failure, and Hybrid MKNF knowledge bases.

2.1. Description Logics

We focus the presentation in this subsection on the description logic $\mathcal{ALC}$, a foundational description logic for the research around OWL. However, our approach is basically independent of the underlying description logic, and the reader familiar with description logics will have no difficulty in applying our approach to more expressive description logics such as $\mathcal{SHOIN}$ or $\mathcal{SROIQ}$, that underlie OWL, resp. OWL 2.[7] We also recall some standard extensions appearing, for example, in lightweight description logics such as $\mathcal{EL}^{++}$ [2]. For further background on description logics we refer to [3, 27].

The basic elements to represent knowledge in DLs are: *individuals*, which represent objects in a domain of discourse; *concepts*, which group together individuals with common properties; and *roles*, which relate individuals. Based on the sets N_I, N_C, and N_R of individual names, concept names, and role names,

---

[7]See, e.g., [27] for the definitions of these and other decription logics and their relationships to OWL and OWL 2.

respectively, the syntactic elements of $\mathcal{ALC}$ are constructed according to the following grammar (in which $A \in N_C$ denotes an atomic concept, $C_{(i)}$ denote complex concepts, $r \in N_R$ denotes a role, and $a_i \in N_I$ denote individuals):

$$C_{(i)} \longrightarrow \bot \mid \top \mid A \mid \neg C \mid C_1 \sqcap C_2 \mid C_1 \sqcup C_2 \mid \exists r.C \mid \forall r.C$$

The semantics of the syntactic elements of $\mathcal{ALC}$ is defined in terms of *interpretations*. An interpretation $I = (\Delta^I, \cdot^I)$ consists of a non-empty set Δ^I – the *domain* of I – and an *interpretation function* $\cdot^I$ that maps each individual $a \in N_I$ to a distinct element $a^I \in \Delta^I$, each concept name $A \in N_C$ to a set $A^I \subseteq \Delta^I$, and each role name $r \in N_R$ to a relation $r^I \subseteq \Delta^I \times \Delta^I$. An interpretation can be extended to complex concepts as follows:

$$\begin{aligned}
\top^I &= \Delta^I \\
\bot^I &= \emptyset \\
(C_1 \sqcap C_2)^I &= C_1^I \cap C_2^I \\
(C_1 \sqcup C_2)^I &= C_1^I \cup C_2^I \\
(\neg C)^I &= \Delta^I \setminus C^I \\
(\forall r.C)^I &= \{x \in \Delta^I \mid \forall y.(x,y) \in r^I \text{ implies } y \in C^I\} \\
(\exists r.C)^I &= \{x \in \Delta^I \mid \exists y.(x,y) \in r^I \text{ and } y \in C^I\}
\end{aligned}$$

An $\mathcal{ALC}$ knowledge base $\mathcal{O}$ is a finite set of *axioms* formed using concepts, roles, and individuals. A *concept assertion* is an axiom of the form $C(a)$ that assigns an individual a to a concept C. A *role assertion* is an axiom of the form $r(a_1, a_2)$ that relates two individuals a_1, a_2 by the role r. Concept and role assertions form the ABox. A *concept inclusion* is an axiom of the form $C_1 \sqsubseteq C_2$ that states the subsumption of the concept C_1 by the concept C_2. A *concept equivalence* axiom $C_1 \equiv C_2$ is a shortcut for two inclusions $C_1 \sqsubseteq C_2$ and $C_2 \sqsubseteq C_1$. Concept inclusions and concept equivalences form the TBox, and the TBox and the ABox form the knowledge base $\mathcal{O}$. An interpretation I satisfies a concept assertion $C(a)$ if $a^I \in C^I$, a role assertion $r(a_1, a_2)$ if $(a_1^I, a_2^I) \in r^I$, a concept inclusion $C_1 \sqsubseteq C_2$ if $C_1^I \subseteq C_2^I$, and a concept equivalence $C_1 \equiv C_2$ if $C_1^I = C_2^I$. An interpretation that satisfies all axioms of a knowledge base $\mathcal{O}$ is called a *model* of $\mathcal{O}$. A concept C is called *satisfiable with respect to $\mathcal{O}$* if $\mathcal{O}$ has a model I in which $C^I \neq \emptyset$. $\mathcal{ALC}$ is a decidable logic, and reasoning under $\mathcal{ALC}$ is ExpTime-complete [3].

$\mathcal{ALC}$ can be extended in several ways, and one common extension is the addition of role inclusion axioms in the TBox. Since we use role inclusions in the running example, we recall them here as well. A *role inclusion* is an axiom of the form $r \sqsubseteq s$ that states that role r is subsumed by the role s. A *role composition axiom* is of the form $r \circ s \sqsubseteq t$, and it states that role t subsumes the composition of roles r and s. An interpretation I satisfies a role inclusion $r \sqsubseteq s$ if $r^I \sqsubseteq s^I$ and a role composition axiom $r \circ s \sqsubseteq t$ if $\forall a_1, a_2, a_3 \in \Delta^I : (a_1, a_2) \in r^I \wedge (a_2, a_3) \in s^I \rightarrow (a_1, a_3) \in t^I$. Note that role compositions can be used to express transitivity of roles and left-and right-identity roles.

Example 2. Consider the online store scenario of Example 1. The following axioms and assertions could be part of the ontology that the store uses:[8]

$$
\begin{align}
\text{CD} &\sqsubseteq \exists\text{HasPiece.Piece} \tag{1}\\
\text{Piece} &\sqsubseteq \exists\text{HasArtist.Artist} \tag{2}\\
\text{HasPiece} \circ \text{HasArtist} &\sqsubseteq \text{HasArtist} \tag{3}\\
\text{TopSeller} \sqcup \text{OnOffer} &\sqsubseteq \text{Recommend} \tag{4}\\
\text{HasPiece(BNAW, BlueTrain)} & \tag{5}\\
\text{HasArtist(BlueTrain, JohnColtrane)} & \tag{6}
\end{align}
$$

Axiom (1) states that each CD consists of at least one piece, and axiom (2) expresses that each piece of music has an artist. The role composition axiom (3) states that if x is related to y by HasPiece and y is related to z by HasArtist, then x is related to z by HasArtist, i.e. HasArtist is a left-identity role. (1)–(3) alone allow us to derive, e.g., that the artist of a piece on a certain CD is an artist of that CD. Note that this conclusion can be drawn without any present CDs, artists or pieces of music, as intended when reasoning with schema knowledge in an infinite domain. Of course, once specific information is available (assertions 5 and 6), we are able to derive, e.g., that John Coltrane is an artist of the album BNAW, and likewise for all the other artists on that CD not explicitly mentioned in the example.

Axiom (4) expresses one general guideline for recommendations: CDs that are on offer or top sellers are automatically recommended to the customers. □

2.2. Logics of Minimal Knowledge and Negation as Failure

The logic of minimal knowledge and negation as failure (MKNF) [46] extends first-order logic with two modal operators **K** and **not** that inspect the knowledge base: intuitively, given a first-order formula φ, **K** φ asks whether φ is known while **not** φ is used to check whether φ is not known. The two modal operators permit local closed world reasoning. In particular, the operator **not** allows one to draw conclusions from the absence of information, in a way similar to that of default negation in Logic Programming. We present below the syntax and the semantics of MKNF as introduced in [51, 52].

Let $\Sigma = (\Sigma_c, \Sigma_f, \Sigma_p)$ be a first-order signature, where Σ_c is a set of constants, Σ_f is a set of function symbols, and Σ_p is a set of predicates containing the binary equality predicate $\approx$. The syntax of MKNF formulas over Σ is defined as follows. A *first-order atom* $P(t_1, \ldots, t_n)$ is an MKNF formula where P is a predicate

---

[8]In these axioms, and throughout the paper, we adopt the convention that names starting with a capital letter represent concepts and roles (termed DL-atoms as defined later in the paper), while names starting with a lower case letter represent variables and predicates not appearing in the ontology (that will be called non-DL-atoms as defined later in the paper). In general, names of individuals/objects also start with a lower case letter. We only make an exception for the names in our running example as these are usually proper names.

and t_i are first-order terms. If φ is an MKNF formula, then $\neg\varphi$, $\exists x : \varphi$, $\mathbf{K}\,\varphi$, and $\mathbf{not}\,\varphi$ are MKNF formulas and likewise $\varphi_1 \wedge \varphi_2$ for MKNF formulas φ_1, φ_2. Moreover, $\varphi_1 \vee \varphi_2$, $\varphi_1 \supset \varphi_2$, $\varphi_1 \equiv \varphi_2$, $\forall x : \varphi$, $\mathbf{t}$, $\mathbf{f}$, $t_1 \approx t_2$, and $t_1 \not\approx t_2$ are abbreviations, respectively, for $\neg(\neg\varphi_1 \wedge \neg\varphi_2)$, $\neg\varphi_1 \vee \varphi_2$, $(\varphi_1 \supset \varphi_2) \wedge (\varphi_2 \supset \varphi_1)$, $\neg(\exists x : \neg\varphi)$, $a \vee \neg a$, $a \wedge \neg a$, $\approx (t_1, t_2)$, and $\neg(t_1 \approx t_2)$. First-order atoms of the form $t_1 \approx t_2$ (resp. $t_1 \not\approx t_2$) are called *equalities* (resp. *inequalities*), and $\varphi[t_1/x_1, \ldots, t_n/x_n]$ denotes the formula obtained by substituting the free variables x_i in φ (i.e. the variables that are not in the scope of any quantifier) by the terms t_i. Given a (first-order) formula φ, $\mathbf{K}\,\varphi$ is called a *modal* $\mathbf{K}$-*atom* and $\mathbf{not}\,\varphi$ a *modal* $\mathbf{not}$-*atom*; modal $\mathbf{K}$-atoms and $\mathbf{not}$-atoms are *modal atoms*. An MKNF formula φ is called *strict*, if there is no modal atom in φ that occurs in the scope of a modal operator. An MKNF formula φ without any free variables is *closed*, and an MKNF formula φ is *ground* if φ does not contain variables at all. An MKNF formula φ is *modally closed* if all modal operators ($\mathbf{K}$ and $\mathbf{not}$) are applied in φ only to closed subformulas, and φ is *positive* if φ does not contain the operator $\mathbf{not}$. An MKNF formula φ is *subjective* if all first-order atoms of φ occur within the scope of a modal operator, and φ is *flat* if φ is subjective and all occurrences of modal atoms in φ are strict.

Let Σ be a signature and Δ a universe. A first-order interpretation I over Σ and Δ assigns an object $a^I \in \Delta$ to each constant $a \in \Sigma_c$, a function $f^I : \Delta^n \to \Delta$ to each n-ary function symbol $f \in \Sigma_f$, and a relation $P^I \subseteq \Delta^n$ to each n-ary predicate $P \in \Sigma_p$. Furthermore, I interprets the predicate $\approx$ as equality – i.e., for $\alpha, \beta \in \Delta$, we have $(\alpha, \beta) \in \approx^I$ iff $\alpha = \beta$. Unlike in standard first-order logic, for each element $\alpha \in \Sigma$, the signature Σ is required to contain a special constant n_α – called a *name* – such that $n_\alpha^I = \alpha$. The interpretation of a variable-free term $t = f(s_1, \ldots, s_n)$ is defined recursively as $t^I = f^I(s_1^I, \ldots, s_n^I)$.

The semantics of an MKNF formula over a signature Σ (henceforth considered implicit in all definitions) is defined as follows. An *MKNF structure* is a triple (I, M, N) where I is a first-order interpretation over Δ and Σ, and M and N are nonempty sets of first-order interpretations over Δ and Σ. Given an MKNF structure (I, M, N), satisfiability of closed MKNF formulas is defined as follows:

$$
\begin{aligned}
(I, M, N) &\models P(t_1, \ldots, t_n) &&\text{iff } (t_1^I, \ldots, t_n^I) \in P^I \\
(I, M, N) &\models \neg\varphi &&\text{iff } (I, M, N) \not\models \varphi \\
(I, M, N) &\models \varphi_1 \wedge \varphi_2 &&\text{iff } (I, M, N) \models \varphi_1 \text{ and } (I, M, N) \models \varphi_2 \\
(I, M, N) &\models \exists x : \varphi &&\text{iff } (I, M, N) \models \varphi[n_\alpha/x] \text{ for some } \alpha \in \Delta \\
(I, M, N) &\models \mathbf{K}\,\varphi &&\text{iff } (J, M, N) \models \varphi \text{ for all } J \in M \\
(I, M, N) &\models \mathbf{not}\,\varphi &&\text{iff } (J, M, N) \not\models \varphi \text{ for some } J \in N
\end{aligned}
$$

Note that the evaluation of $\mathbf{K}$ and $\mathbf{not}$ are kept separate in this definition of satisfiability. The relation between these operators is established in the notion of a two-valued MKNF model defined below.

An *MKNF interpretation* M over a universe Δ is a nonempty set of first-order interpretations. For a closed MKNF formula φ, we say that M *satisfies* φ, written $M \models \varphi$, if $(I, M, M) \models \varphi$ for each $I \in M$. The notion of a two-valued

MKNF model of a closed MKNF formula φ is based on a preference relation on MKNF interpretations that satisfy φ. More precisely, an MKNF interpretation M over Δ is a *two-valued MKNF model* of a closed MKNF formula φ if (1) M satisfies φ, and (2) for each MKNF interpretation M' such that $M' \supset M$ we have $(I', M', M) \not\models \varphi$ for some $I' \in M'$. An MKNF formula φ is *MKNF satisfiable* if a two-valued MKNF model of φ exists; otherwise φ is *MKNF unsatisfiable*. Furthermore, φ *MKNF entails* ψ, written $\varphi \models_{MKNF} \psi$, if $M \models \psi$ for each two-valued MKNF model M of φ.

Note that this definition of model is asymmetric in the treatment of the modal operators **K** and **not**. In fact, the maximisation of M in (2) is only done in the component of the structure used for evaluating the operator **K**. This results in a minimisation of the derivable **K**-atoms in any two-valued MKNF model of a given formula φ.

Example 3. Though $M = \{\{p\}\}$ satisfies both **K** p and $\neg$**not** p, M is only a two-valued MKNF model of the first formula. M is not a two-valued MKNF model of the second one since $(I', M', M) \models \neg$**not** p holds for any M' with $M' \supset M$. $\qquad\square$

The MKNF semantics, as originally defined in [46], shows certain undesirable properties such as counterintuitive semantics caused by the usage of arbitrary universes and the differing interpretation of constants in different interpretations. To overcome these problems, [52] additionally applies the standard name assumption to hybrid MKNF knowledge bases. We briefly recall two such problems from [52] and the notion of standard names assumption introduced to overcome them, and we refer for the complete discussion to [52].

One problem when using MKNF as in [46] for the integration of rules and ontologies is the usage of arbitrary universes. Consider the MKNF formula $\varphi = \varphi_1 \wedge \varphi_2$, where $\varphi_1 = \mathbf{K}\, P(a)$ and $\varphi_2 = \mathbf{not}\, P(b) \supset \mathbf{f}$. Intuitively, one would not expect that φ is satisfiable since there is no indication that $P(b)$ should be true. However, if the universe contains only one element, then a and b are interpreted as the same object, and φ is satisfied. In this case one unintendedly derives that $\varphi \models a \approx b$ holds.

Another problem is caused by constants that are interpreted differently in different interpretations. Consider $\varphi_1 = \mathbf{K}\, P(a)$ and $\varphi_2 = \exists x : \mathbf{K}\, P(x)$. In this case one would expect that $\varphi_1 \models \varphi_2$, that is, every two-valued MKNF model of φ_1 is also a two-valued MKNF model of φ_2. However, let M be an MKNF interpretation containing two elements I_1 and I_2 where I_1 is a first-order interpretation in which a is interpreted as a name α_1 and $I_1 \models P(\alpha_1)$, and I_2 is a first-order interpretation in which a is interpreted as some other name α_2 and $I_2 \models P(\alpha_2)$. We thus have that $M \models \varphi_1$ but not $M \models \varphi_2$ since this would require to have an x in the domain such that $P(x)$ is true in all $I \in M$.

To avoid such unintended behaviour, the standard name assumption is imposed on top of MKNF.

Definition 1. (Standard Name Assumption [52]). A first-order interpretation I over a signature Σ employs the *standard name assumption* if

(1) the universe Δ of I contains all constants of Σ and a countably infinite number of additional constants called *parameters*;

(2) $t^I = t$ for each ground term t constructed using the function symbols from Σ and the constants from Δ; and

(3) the predicate $\approx$ is interpreted in I as a congruence relation – that is, $\approx$ is reflexive, symmetric, transitive, and allows for the replacement of equals by equals [18].

Consequences of first-order formulas under the standard first-order semantics and the standard name assumption cannot be distinguished [52]. Thus, in the rest of the paper we use the standard name assumption for first-order inferences.

2.3. Hybrid MKNF Knowledge Bases

Hybrid MKNF knowledge bases as introduced in [51, 52][9] essentially are MKNF formulas restricted to a certain form. They consist of two components: a decidable description logic knowledge base translatable into first-order logic and a finite set of rules of modal atoms.

More precisely, the approach of hybrid MKNF knowledge bases is applicable to any first-order fragment $\mathcal{DL}$ satisfying the following conditions: (i) each knowledge base $\mathcal{O} \in \mathcal{DL}$ can be translated[10] into a formula $\pi(\mathcal{O})$ of function-free first-order logic with equality; (ii) $\mathcal{DL}$ supports *A-Box*-assertions of the form $P(a_1, \ldots, a_n)$, where P is a predicate and each a_i a constant of $\mathcal{DL}$; and (iii) satisfiability checking and instance checking (i.e. checking entailments of the form $\mathcal{O} \models P(a_1, \ldots, a_n)$) are decidable. In particular, description logics around OWL satisfy these conditions. Note that we limit ourselves to function-free first-order logic since otherwise decidability would not be possible. Thus for the rest of the paper, we will not allow function symbols in hybrid MKNF knowledge bases.

We recall MKNF rules and hybrid MKNF knowledge bases from [51].

Definition 2. Let $\mathcal{O}$ be a DL knowledge base. A function-free first-order atom $P(t_1, \ldots, t_n)$ over Σ such that P is $\approx$ or occurs in $\mathcal{O}$ is called a *DL-atom*; all other atoms are called *non-DL-atoms*. An MKNF rule r has the following form where H_i, A_i, and B_i are function free first-order atoms:

$$\mathbf{K}\,H_1 \vee \ldots \vee \mathbf{K}\,H_l \leftarrow \mathbf{K}\,A_1, \ldots, \mathbf{K}\,A_n, \mathbf{not}\,B_1, \ldots, \mathbf{not}\,B_m \qquad (7)$$

The sets $\{\mathbf{K}\,H_i\}$, $\{\mathbf{K}\,A_i\}$, and $\{\mathbf{not}\,B_i\}$ are called the *rule head*, the *positive body*, and the *negative body*, respectively. A rule r is *nondisjunctive* if $l = 1$; r is *positive* if $m = 0$; r is a *fact* if $n = m = 0$. A *program* $\mathcal{P}$ is a finite set of MKNF rules. A *hybrid MKNF knowledge base* $\mathcal{K}$ is a pair $(\mathcal{O}, \mathcal{P})$ and $\mathcal{K}$ is *nondisjunctive* if all rules in $\mathcal{P}$ are nondisjunctive.

---

[9]We focus here on the presentation as in [51], and we thus omit classical negation and arbitrary first-order formulas in rules as presented in [52].

[10]See [3] for standard translations of Description Logic axioms.

Hybrid MKNF knowledge bases rules[11], as defined above, do not coincide syntactically with any MKNF formula. For interpreting hybrid MKNF knowledge bases in terms of MKNF logic, the transformation π that transforms a DL ontology into first-order formulas is extended to knowledge bases as follows:

Definition 3. Let $\mathcal{K} = (\mathcal{O}, \mathcal{P})$ be a hybrid MKNF knowledge base. We extend π to rules r of the form (7), $\mathcal{P}$, and $\mathcal{K}$ as follows, where $\vec{x}$ is the vector of the free variables of r.

$$\pi(r) = \forall \vec{x} : (\mathbf{K} H_1 \vee \ldots \vee \mathbf{K} H_l \subset \mathbf{K} A_1 \wedge \ldots \wedge K A_n \wedge \mathbf{not}\, B_1 \wedge \ldots \wedge \mathbf{not}\, B_m)$$

$$\pi(\mathcal{P}) = \bigwedge_{r \in \mathcal{P}} \pi(r) \qquad \pi(\mathcal{K}) = \mathbf{K}\, \pi(\mathcal{O}) \wedge \pi(\mathcal{P})$$

To simplify the presentation, we will abuse notation in the rest of the paper and identify $\mathcal{K}$ with $\pi(\mathcal{K})$. It will be obvious from the context when $\mathcal{K}$ represents its first-order transformation $\pi(\mathcal{K})$.

Hybrid MKNF knowledge bases, even without function symbols, are in general undecidable, unless they are restricted in some way. The reason for that is that rules can be applied to all the objects in the infinite domain. The basic idea to make reasoning with hybrid MKNF knowledge bases decidable is to apply rules only to the individuals that appear in the knowledge base. This restriction is achieved by DL-safety.

Definition 4. An MKNF rule r is *DL-safe* if every variable in r occurs in at least one non-DL-atom $\mathbf{K}\, B$ occurring in the body of r. A hybrid MKNF knowledge base $\mathcal{K}$ is *DL-safe* if all the rules in $\mathbf{K}$ are DL-safe.

In the rest of the paper, unless otherwise stated, we only consider DL-safe knowledge bases.

Grounding the knowledge base, as defined below, ensures that rules apply to all the individuals appearing in the knowledge base, whereas DL-safety guarantees that no other individual can be used.

Definition 5. Given a hybrid MKNF knowledge base $\mathcal{K} = (\mathcal{O}, \mathcal{P})$, the *ground instantiation of $\mathcal{K}$* is the KB $\mathcal{K}_G = (\mathcal{O}, \mathcal{P}_G)$ where $\mathcal{P}_G$ is obtained from $\mathcal{P}$ by replacing each rule r of $\mathcal{P}$ with a set of rules substituting each variable in r with constants from $\mathcal{K}$ in all possible ways.

It was shown in [52] that, for a DL-safe hybrid knowledge base $\mathcal{K}$, the two-valued MKNF models of $\mathcal{K}$ and $\mathcal{K}_G$ coincide.

Example 4. Consider again the scenario of Example 1, together with the axioms and assertions of Example 2. These axioms can be part of an ontology $\mathcal{O}$

---

[11]As mentioned before, our main results only apply to nondisjunctive rules, i.e. $l = 1$ for each rule of the form (7). However, to keep to the original definition of [52], we present the definition of the more general form of MKNF rules.

of a hybrid MKNF knowledge base $\mathcal{K} = (\mathcal{O}, \mathcal{P})$. In $\mathcal{P}$ we can encode further recommendation guidelines, in particular those that require closed world reasoning. For example, imagine that we want to give customers recommendations for interesting CDs they do not own and that do not have a low evaluation. This can be encoded with the rules shown below:[12]

$$
\begin{aligned}
\mathbf{K}\,\mathrm{Recommend(x)} \quad &\leftarrow \quad \mathbf{K}\,\mathrm{CD(x)}, \mathbf{not}\,\mathrm{owns}(x), \mathbf{not}\,\mathrm{LowEval}(x), \\
& \qquad \mathbf{K}\,\mathrm{interesting}(x). \tag{8} \\
\mathbf{K}\,\mathrm{interesting}(x) \quad &\leftarrow \quad \mathbf{K}\,\mathrm{CD}(x), \mathbf{K}\,\mathrm{CD}(y), \mathbf{K}\,\mathrm{owns(y)}, \mathbf{not}\,\mathrm{owns}(x), \\
& \qquad \mathbf{K}\,\mathrm{similar}(x, y). \tag{9} \\
\mathbf{K}\,\mathrm{similar}(x, y) \quad &\leftarrow \quad \mathbf{K}\,\mathrm{CD(x)}, \mathbf{K}\,\mathrm{CD(y)}, \mathbf{K}\,\mathrm{Artist(z)}, \\
& \qquad \mathbf{K}\,\mathrm{HasArtist}(x, z), \mathbf{K}\,\mathrm{HasArtist}(y, z). \tag{10}
\end{aligned}
$$

$$
\begin{aligned}
\mathbf{K}\,\mathrm{owns(EnConcert)} \quad &\leftarrow \tag{11} \\
\mathbf{K}\,\mathrm{HasArtist(EnConcert,JackJohnson)} \quad &\leftarrow \tag{12} \\
\mathbf{K}\,\mathrm{HasArtist(ToTheSea,JackJohnson)} \quad &\leftarrow \tag{13} \\
\mathbf{K}\,\mathrm{OnOffer(BNAW)} \quad &\leftarrow \tag{14}
\end{aligned}
$$

Note that closed world reasoning is used for owns and lowEval. In the case of predicate owns, it is reasonable to assume that the knowledge about owned CDs is fully available. So, if there is no fact stating that a given CD is owned, one should assume that the CD is not owned. In the case of predicate lowEval it might happen that there is no evaluation yet available, and we want the recommendation anyway: a CD is not considered for recommendation only when there actually is a (known) low evaluation for the CD. Such an evaluation could be taken from other customers of the store or from a web page of professional reviews. Here, for simplicity, we keep this part of the reasoning process implicit.

Moreover, in the rules above, a CD is interesting if the customer owns another CD which is similar (9), and two CDs are similar if they have a common artist (10). Note that the predicate CD is used to ensure DL-safety, and we assume that the instances of that predicate relevant to any drawn conclusion are always appropriately defined.

If we now add facts (11)–(14), then we can derive Recommend(ToTheSea), since no low evaluation is known for ToTheSea, and Recommend(BNAW) since BNAW is on offer.

This example illustrates that hybrid MKNF knowledge bases allow us to obtain consequences for predicates that are 'defined' both in the ontology and

---

[12]In this encoding, for simplicity, we consider that the program part of the knowledge base is specific to each customer. This avoids an explicit representation of several customers, of the relation stating which CDs are owned by each customer, etc.

in the rules. The result may then be further applied to derive subsequent consequences either in rules or in the ontology. Note that the facts (12) and (13) are here explicitly added, representing the implicit consequences derivable from the appropriate ontology alone, similar to HasArtist(BNAW,JohnColtrane) in Example 2. □

In [52], several reasoning algorithms were provided for combinations of arbitrary description logic fragments and rules of differing expressivity. Table 1 presents the data complexity of instance checking for combinations of nondisjunctive rules (with arbitrary or stratified[13] negation or without **not** in the rules) with description logics fragments of differing computational complexity. We point out that allowing arbitrary nonmonotonic negation increases data complexity drastically and in particular beyond tractability. This is not the case of the rules in Example 4, since these are stratified. However, this is just an initial set of rules for our running example, which is further elaborated below and becomes non-stratified (e.g., the addition of rule 17 in Example 8 renders the set of rules non-stratified). With the proposal we present in the following sections these sets of non-stratified rules do not constitute a problem regarding complexity.

rules	$\mathcal{DL} = \emptyset$	$\mathcal{DL} \in \mathrm{P}$	$\mathcal{DL} \in \mathrm{coNP}$
definite	P	P	coNP
stratified	P	P	Δ_2^p
normal	coNP	coNP	Π_2^p

Table 1: Data complexity of instance checking in MKNF

3. Three-valued MKNF Semantics

In this section we introduce a three-valued semantics for hybrid MKNF knowledge bases. The rationale and the main goal behind this three-valued semantics is to define a semantics that is closely related to the well-founded semantics of logic programs. This is done in order to take advantage of the (data) complexity of the WFS that is lower than the (data) complexity of the corresponding two-valued semantics. Nevertheless, the DL-part of a hybrid MKNF knowledge base is still interpreted under the two-valued semantics. Thus, we achieve a faithful integration, in the sense that without rules the meaning of the knowledge base exactly coincides with the usual semantics from DLs.

The definition of the three-valued semantics presented in this section applies equally with or without the standard name assumption. However, since we want to achieve a semantics that is faithful w.r.t. the two-valued MKNF semantics as presented in Section 2, we assume standard name assumption.

---

[13]Essentially, rules can be separated into strata that can be evaluated separately – see Section 4.4 in [52].

The two-valued hybrid MKNF semantics [52] is closely related (cf. [46]) to the stable models semantics [19]. In both of them, the meaning of a knowledge base is determined by a set of models. In fact, an MKNF formula such as $\varphi = ((\mathbf{not}\, p \supset \mathbf{K}\, q \wedge (\mathbf{not}\, q \supset \mathbf{K}\, p))$ (and the corresponding set of rules) has two models – one model in which p is true and q is false, and another one in which p is false and q is true. Moreover, these two-valued models are, in general, obtained by a guess and check process, thus having a high computational complexity.

The well-founded semantics of logic programs [62] generalises the two-valued models of the stable model semantics to a three-valued setting. In this way, it is possible to determine the meaning of a knowledge base solely on the basis of a single (minimal) model that is obtained with a lower computational complexity. Intuitively, a third truth value $\mathbf{u}$, denoting undefined, is introduced as an alternative to the values $\mathbf{t}$ and $\mathbf{f}$, enabling one to delay the evaluation to any of the two latter values until further information is available. We want to follow this idea when defining a three-valued MKNF semantics. There is however one more problem to be taken into account: since we are interested in applying the semantics to hybrid MKNF knowledge bases containing two-valued ontologies, which we want to integrate faithfully, we are going to define the semantics in such a way that an MKNF formula corresponding to a DL fragment is ensured to be just two-valued.

We therefore define a three-valued MKNF semantics that extends the two-valued semantics of [52], but remains two-valued for the case of MKNF formulas without modal operators. We start by defining MKNF structures for this three-valued setting.

Definition 6. A *three-valued (partial) MKNF structure* $(I, \mathcal{M}, \mathcal{N})$ consists of a first-order interpretation I and two pairs $\mathcal{M} = \langle M, M_1 \rangle$ and $\mathcal{N} = \langle N, N_1 \rangle$ of sets of first-order interpretations where $M_1 \subseteq M$ and $N_1 \subseteq N$. An MKNF structure is called *total* if $\mathcal{M} = \langle M, M \rangle$ and $\mathcal{N} = \langle N, N \rangle$.

In the two-valued semantics, an MKNF structure (I, M, N) contains sets of interpretations M and N for evaluating a modal atom $\mathbf{K}\,\varphi$, respectively $\mathbf{not}\,\varphi$, to $\mathbf{t}$ or $\mathbf{f}$, depending on whether φ is contained in all elements of M, respectively N. This clearly leaves no space for an extension to a third truth value $\mathbf{u}$. So, we turn sets of interpretations into pairs of sets of interpretations. Then, as we show below, a modal atom $\mathbf{K}\,\varphi$ is true w.r.t. $\langle M, M_1 \rangle$ if φ is true in all elements of M; a modal atom $\mathbf{K}\,\varphi$ is false if φ is not true in all elements of M_1; a modal atom $\mathbf{K}\,\varphi$ is undefined otherwise (i.e. if φ is true in all elements of M_1). The additional restrictions, saying that $M_1 \subseteq M$ and $N_1 \subseteq N$, are needed to ensure that no modal atom can be both true and false at the same time, and it can easily be shown via induction that the same holds for any MKNF formula φ. In this way, we guarantee that no fourth truth value 'both' is needed. Nevertheless, given an MKNF formula φ, three-valued MKNF structures may evaluate $\mathbf{K}\,\varphi$ and $\mathbf{not}\,\varphi$ to true at the same time, just like in the two-valued case, and we show

below how to prevent this from happening when defining MKNF interpretation pairs.

We now define the evaluation of closed MKNF formulas in such three-valued MKNF structures.

Definition 7. Let $(I, \mathcal{M}, \mathcal{N})$ be a three-valued MKNF structure and $\{\mathbf{t}, \mathbf{u}, \mathbf{f}\}$ the set of truth values with the order $\mathbf{f} < \mathbf{u} < \mathbf{t}$, where the operator max (resp. min) chooses the greatest (resp. least) element with respect to this ordering. We define:

- $(I, \mathcal{M}, \mathcal{N})(P(t_1, \ldots, t_n)) = \begin{cases} \mathbf{t} & \text{iff } (t_1^I, \ldots, t_n^I) \in P^I \\ \mathbf{f} & \text{iff } (t_1^I, \ldots, t_n^I) \notin P^I \end{cases}$

- $(I, \mathcal{M}, \mathcal{N})(\neg\varphi) = \begin{cases} \mathbf{t} & \text{iff } (I, \mathcal{M}, \mathcal{N})(\varphi) = \mathbf{f} \\ \mathbf{u} & \text{iff } (I, \mathcal{M}, \mathcal{N})(\varphi) = \mathbf{u} \\ \mathbf{f} & \text{iff } (I, \mathcal{M}, \mathcal{N})(\varphi) = \mathbf{t} \end{cases}$

- $(I, \mathcal{M}, \mathcal{N})(\varphi_1 \wedge \varphi_2) = \min\{(I, \mathcal{M}, \mathcal{N})(\varphi_1), (I, \mathcal{M}, \mathcal{N})(\varphi_2)\}$

- $(I, \mathcal{M}, \mathcal{N})(\varphi_1 \supset \varphi_2) = \mathbf{t}$ iff $(I, \mathcal{M}, \mathcal{N})(\varphi_2) \geq (I, \mathcal{M}, \mathcal{N})(\varphi_1)$ and $\mathbf{f}$ otherwise

- $(I, \mathcal{M}, \mathcal{N})(\exists x : \varphi) = \max\{(I, \mathcal{M}, \mathcal{N})(\varphi[\alpha/x]) \mid \alpha \in \Delta\}$

- $(I, \mathcal{M}, \mathcal{N})(\mathbf{K}\,\varphi) = \begin{cases} \mathbf{t} & \text{iff } (J, \langle M, M_1 \rangle, \mathcal{N})(\varphi) = \mathbf{t} \text{ for all } J \in M \\ \mathbf{f} & \text{iff } (J, \langle M, M_1 \rangle, \mathcal{N})(\varphi) = \mathbf{f} \text{ for some } J \in M_1 \\ \mathbf{u} & \text{otherwise} \end{cases}$

- $(I, \mathcal{M}, \mathcal{N})(\mathbf{not}\,\varphi) = \begin{cases} \mathbf{t} & \text{iff } (J, \mathcal{M}, \langle N, N_1 \rangle)(\varphi) = \mathbf{f} \text{ for some } J \in N_1 \\ \mathbf{f} & \text{iff } (J, \mathcal{M}, \langle N, N_1 \rangle)(\varphi) = \mathbf{t} \text{ for all } J \in N \\ \mathbf{u} & \text{otherwise} \end{cases}$

As intended, this evaluation is not a purely three-valued one, since first-order atoms are evaluated as in the two-valued case. In fact, an MKNF formula φ without modal operators (and thus also a pure description logic knowledge base) is only two-valued. It can easily be seen that such a φ is evaluated in exactly the same way as in the scheme presented in Section 2. This is desired in particular when the knowledge base consists just of the DL part. So, the third truth value only affects MKNF formulas containing modal atoms, which in the case of hybrid MKNF knowledge bases can only occur in the rules. These rules, corresponding to implications, are, however, no longer interpreted in a way one would expect from a boolean perspective: $\mathbf{u} \supset \mathbf{u}$ is true in the evaluation defined above, while $\mathbf{u} \vee \neg\mathbf{u}$ is actually undefined. The reason for this change is that, in this way rules can only be true or false, similarly to what happens in logic programming, even when they contain undefined modal atoms. Intuitively, the advantage for hybrid MKNF knowledge bases is that we can leave single modal

atoms undefined, thus not necessarily having to create several models, while the entire knowledge base is only true or false.

We point out that the evaluation of **not** w.r.t. $\langle N, N_1 \rangle$ is symmetrical to the evaluation of **K** w.r.t. $\langle M, M_1 \rangle$, only that the conditions are switched. E.g., the condition for true modal **K**-atoms w.r.t. M yields false modal **not**-atoms w.r.t. N. In case of $M = N$ and $M_1 = N_1$ this corresponds to the two-valued (monotonic) evaluation in Section 2.

3.2. Three-valued MKNF Models

Given the above definition of evaluation of MKNF formulas, we are now ready to extend (two-valued) MKNF interpretations and MKNF models to three truth values. For that purpose, we have to generalize (two-valued) MKNF interpretations M to pairs of MKNF interpretations (M, N), since otherwise no formula could ever be undefined.

Definition 8. An *MKNF interpretation pair* (M, N) consists of two MKNF interpretations M, N with $\emptyset \subset N \subseteq M$. An MKNF interpretation pair *satisfies* a closed MKNF formula φ, written $(M, N) \models \varphi$, if and only if

$$(I, \langle M, N \rangle, \langle M, N \rangle)(\varphi) = \mathbf{t}$$

for each $I \in M$. If $M = N$, then the MKNF interpretation pair (M, N) is called *total*. If there exists an MKNF interpretation pair satisfying φ, then φ is *consistent*.

The set M contains all interpretations that model only truth, while N models everything that is true or undefined. Evidently, just as in the two-valued case, anything not being modelled in N is false. The subset relation between M and N ensures that MKNF interpretation pairs are defined in accordance with the three-valued MKNF structures, so that each formula is evaluated to exactly one truth value. Note the striking similarity compared to MKNF interpretations in the two-valued case by using the MKNF interpretation pair (M, N) to evaluate both **K** and **not** simultaneously.

We now define the preference relation on MKNF interpretation pairs that is required for the notion of (nonmonotonic) three-valued MKNF models, following an approach similar to the one in the two-valued case – i.e. by minimising non-falsity (truth or undefinedness, in this case) of formulas w.r.t. **K**.

Definition 9. Any MKNF interpretation pair (M, N) is a *three-valued MKNF model* for a given closed MKNF formula φ if

(1) (M, N) satisfies φ and

(2) for each MKNF interpretation pair (M', N') with $M \subseteq M'$ and $N \subseteq N'$, where at least one of the inclusions is proper and $M' = N'$ if $M = N$, there is $I' \in M'$ such that $(I', \langle M', N' \rangle, \langle M, N \rangle)(\varphi) \neq \mathbf{t}$.

Condition (1) checks whether (M, N) evaluates φ to $\mathbf{t}$ while the second condition verifies that (M, N) contains only knowledge necessary to obtain this evaluation to $\mathbf{t}$. This is achieved by generalising the corresponding notion in the two-valued MKNF semantics to the three-valued case: for each MKNF interpretation pair (M', N') that properly subsumes (M, N), it is checked that φ does not evaluate to $\mathbf{t}$ for all $I' \in M'$, where (M', N') is used to evaluate $\mathbf{K}$ while (M, N) evaluates $\mathbf{not}$. Intuitively, one may consider an MKNF interpretation pair as a guess for the true evaluation of the considered formula, and condition (2) checks, having fixed the evaluation of modal $\mathbf{not}$-atoms, whether the evaluation of modal $\mathbf{K}$-atoms is actually minimal w.r.t. to the order $\mathbf{f} < \mathbf{u} < \mathbf{t}$ of truth values. We illustrate in the example below how this minimisation is achieved.

Example 5. Consider the MKNF formula φ (corresponding to two rules):

$$(\mathbf{not}\, p \supset \mathbf{K}\, q) \wedge (\mathbf{not}\, q \supset \mathbf{K}\, p)$$

An MKNF interpretation pair (M, N) that satisfies condition (1) of Definition 9 has to evaluate both conjuncts to true. The MKNF interpretation pair $(\{\{p\}, \{p, q\}\}, \{\{p, q\}\})$ that evaluates $\mathbf{K}\, p$ to $\mathbf{t}$ and $\mathbf{K}\, q$ to $\mathbf{u}$ satisfies the first condition but is not a three-valued MKNF model of φ since, e.g., $(M', N') = (\{\emptyset, \{p\}, \{q\}, \{p, q\}\}, \{\{p, q\}\})$ violates condition (2). In fact, this MKNF interpretation pair (M', N') is a three-valued MKNF model. The operator $\mathbf{not}$ is always evaluated w.r.t. the MKNF interpretation pair (M, N), even when considering condition (2) of Definition 9, so, for $N = \{\{p, q\}\}$, the two implications are true anyway, and M has to be the set of all possible interpretations $\{\emptyset, \{p\}, \{q\}, \{p, q\}\}$ to satisfy condition (2). Thus, we obtain the MKNF interpretation pair that evaluates $\mathbf{K}\, p$ and $\mathbf{K}\, q$ to $\mathbf{u}$. In other words, the initial MKNF interpretation pair was not minimal w.r.t. the evaluation of modal $\mathbf{K}$-atoms. Similar to the minimisation of the evaluation of $\mathbf{K}\, p$ from $\mathbf{t}$ to $\mathbf{u}$, changes from $\mathbf{u}$ to $\mathbf{f}$ are possible: maintain the original $M = \{\{p\}, \{p, q\}\}$ and set $N = M$. Now the evaluation of $\mathbf{K}\, q$ is minimised from $\mathbf{u}$ to $\mathbf{f}$, and it is easy to verify that the resulting MKNF interpretation pair is in fact a three-valued MKNF model of φ. $\square$

It should be pointed out that the larger the set M or N is, the less true or undefined knowledge is inferred. So, minimisation is achieved by increasing the sets in consideration. Note that $N' \subseteq M'$ for MKNF interpretation pairs (M', N') ensures that we only check reasonable candidates for augmenting (M, N).[14]

We now adapt some notions needed in the rest of the paper from the two-valued MKNF semantics to the three-valued setting.

---

[14] In comparison to [37] the definition has been slightly altered to simplify proofs and computation: in case of a total MKNF interpretation pair (M, M), it is sufficient to check that no other total MKNF interpretation pair (M', M') actually yields a true evaluation for all $I' \in M'$. This simplification is also justified by the intuition of enlarging N' separately: there is no undefinedness in a total MKNF interpretation pair, and minimisation of undefinedness is thus not necessary.

Definition 10. If there is a three-valued MKNF model for a given closed MKNF formula φ, then φ is called *MKNF-consistent*, otherwise φ is called *MKNF-inconsistent*. If $(I, \langle M, N \rangle, \langle M, N \rangle)(\psi) = \mathbf{t}$ for all three-valued MKNF models (M, N) of φ, then φ *entails* ψ, written $\varphi \models^3_{MKNF} \psi$.

Note that MKNF-inconsistent MKNF formulas do not necessarily evaluate to $\mathbf{f}$. For example, $\varphi = \mathbf{K} u \wedge \mathbf{not}\, u$ evaluates to $\mathbf{u}$ for the MKNF interpretation pair $(\{\{u\}, \emptyset\}, \{\{u\}\})$. In fact, an MKNF-inconsistent formula can even evaluate to $\mathbf{t}$ and that was already the case for the two-valued MKNF semantics of [52]. E.g., $\varphi = \neg \mathbf{not}\, p$ is MKNF-inconsistent, and it evaluates to $\mathbf{t}$ in some MKNF interpretation pairs (and also in some MKNF interpretations of [52]). This does not constitute a problem since it does not affect the definitions of MKNF models or MKNF-consistency.

Though the notions of inconsistency and unsatisfiability are usually applied in the same technical sense, we want to distinguish between MKNF-satisfiability in the two-valued case and MKNF-consistency for three-valued MKNF models. Likewise, we distinguish between the two-valued notion 'MKNF entails' and the three-valued 'entails'.

In spite of keeping the notions separate, two- and three-valued MKNF models are closely related: we now show that any two-valued MKNF model M corresponds exactly to a (total) three-valued one and vice versa. For that purpose, we first prove that evaluation in an MKNF structure (I, M, N) and evaluation in a total three-valued structure $(I, \langle M, M \rangle, \langle N, N \rangle)$ are identical. Intuitively, this holds because nothing can be undefined in a total three-valued structure.

Lemma 1. *Given a closed MKNF formula φ, $(I, M, N) \models \varphi$ if and only if $(I, \langle M, M \rangle, \langle N, N \rangle)(\varphi) = \mathbf{t}$.*

PROOF. The proof is done by induction on the formula φ.

Let φ be $P(t_1, \ldots, t_n)$. We have $(I, M, N) \models P(t_1, \ldots, t_n)$ iff $(t_1^I, \ldots, t_n^I) \in P^I$ iff $(I, \langle M, M \rangle, \langle N, N \rangle)(P(t_1, \ldots, t_n)) = \mathbf{t}$.

Assume that the lemma holds for φ_1. We show the induction steps for $\neg$ and $\mathbf{K}$, all the other cases follow analogously.

Let φ be $\neg \varphi_1$. We have that $(I, M, N) \models \neg \varphi_1$ iff $(I, M, N) \not\models \varphi_1$ iff, by the induction hypothesis, $(I, \langle M, M \rangle, \langle N, N \rangle)(\varphi_1) = \mathbf{f}$ iff by definition of evaluation in partial structures $(I, \langle M, M \rangle, \langle N, N \rangle)(\neg \varphi_1) = \mathbf{t}$.

Let φ be $\mathbf{K} \varphi_1$. We have $(I, M, N) \models \mathbf{K} \varphi_1$ iff $(I, M, N) \models \varphi_1$ holds for each $I \in M$ iff $(I, \langle M, M \rangle, \langle N, N \rangle)(\varphi_1) = \mathbf{t}$ for all $I \in M$ by the induction hypothesis iff $(I, \langle M, M \rangle, \langle N, N \rangle)(\mathbf{K} \varphi_1) = \mathbf{t}$. $\qquad\square$

This lemma can be used to show that every two-valued MKNF model M corresponds to a three-valued MKNF model (M, M), like in [36], and also the converse, i.e. that every three-valued MKNF model (M, M) corresponds to a two-valued MKNF model in the sense of [52].

Proposition 1. *Given a closed MKNF formula φ, M is a two-valued MKNF model of φ if and only if (M, M) is a three-valued MKNF model of φ.*

PROOF. Let (M, M) be a three-valued MKNF model of φ, i.e. (M, M) satisfies the two conditions of Definition 9. We show that M is a two-valued MKNF model of φ. It follows from the first of the two conditions of Definition 9 that $(I, \langle M, M \rangle, \langle M, M \rangle)(\varphi) = \mathbf{t}$ for all $I \in M$ and therefore, by Lemma 1, that $(I, M, M) \models \varphi$ for each $I \in M$. The second condition states, for each MKNF interpretation pair (M', M') with $M \subset M'$, that we have $(I', \langle M', M' \rangle, \langle M, M \rangle)(\varphi) \neq \mathbf{t}$ for some $I' \in M'$. We conclude from Lemma 1 that for any M' with $M' \supset M$ there is an $I' \in M'$ such that $(I', M', M) \not\models \varphi$.

Now, let M be a two-valued MKNF model of φ. We show that (M, M) is a three-valued MKNF model of φ. We know that $(I, M, M) \models \varphi$ for each $I \in M$ since M is a two-valued MKNF model of φ. As such, $(I, \langle M, M \rangle, \langle M, M \rangle)(\varphi) = \mathbf{t}$ holds for all $I \in M$ by Lemma 1, and so the first of the two conditions of Definition 9 is satisfied. Furthermore, since M is a two-valued MKNF model of φ, we know that for all M' with $M' \supset M$ we have $(I', M', M) \not\models \varphi$ for some $I' \in M'$. Again, from Lemma 1, we know that for any MKNF interpretation pair (M', M') with $M' \supset M$ we have $(I', \langle M', M' \rangle, \langle M, M \rangle)(\varphi) \neq \mathbf{t}$ for some $I' \in M'$. This is sufficient since, according to Definition 9, for (M, M) we only need to consider total MKNF interpretation pairs (M', M'). $\qquad\square$

MKNF interpretation pairs can be compared by an order that resembles the knowledge order from logic programming. Intuitively, given such an order and two MKNF interpretation pairs (M_1, N_1) and (M_2, N_2), we have that (M_1, N_1) is greater than (M_2, N_2) w.r.t. such an order if (M_1, N_1) allows us to derive more true and false knowledge than (M_2, N_2). Taking into account that a larger set of interpretations derives less true and more false knowledge, we can define the following order on MKNF interpretation pairs.

Definition 11. Let (M_1, N_1) and (M_2, N_2) be MKNF interpretation pairs. We have that $(M_1, N_1) \succeq_k (M_2, N_2)$ iff $M_1 \subseteq M_2$ and $N_1 \supseteq N_2$.

Such an order is of particular interest for comparing models. In logic programming the least model w.r.t. derviable knowledge among all three-valued models for a given program is the well-founded model. Here, we want to introduce a similar notion referring to the minimal three-valued MKNF models, i.e. the ones among all three-valued MKNF models that leave as much as possible undefined.

Definition 12. Let φ be a closed MKNF formula and (M, N) a partial MKNF model of φ such that $(M_1, N_1) \succeq_k (M, N)$ for all three-valued MKNF models (M_1, N_1) of φ. Then (M, N) is a *well-founded MKNF model* of φ.

Of course, if φ is inconsistent, then there are no three-valued MKNF models and thus no well-founded MKNF models of φ. However, if φ is a consistent hybrid MKNF knowledge base, it is guaranteed that a well-founded MKNF model of φ exists. Moreover, this well-founded model is unique. As we shall see, this model is especially important in that a modal atom $\mathbf{K} H$ is true in the well-founded MKNF model iff $\mathbf{K} H$ is true in all three-valued MKNF models. This way, performing skeptical reasoning in three-valued MKNF models amounts to determining the well-founded MKNF model.

Theorem 1. *If $\mathcal{K}$ is an MKNF-consistent nondisjunctive DL-safe hybrid MKNF KB, then a well-founded MKNF model exists, and it is unique.*

The respective proofs for the uniqueness/existence of the well-founded MKNF model, and how to calculate this unique model, are presented in Section 4 (as a direct consequence of Theorem 5). The following example gives at least an intuitive insight into the correspondence between two-valued and three-valued MKNF models, and the well-founded MKNF model.

Example 6. Consider the knowledge base $\mathcal{K}$ corresponding to the MKNF formula φ from Example 5.

$$\begin{aligned}\mathbf{K}\,q &\leftarrow \mathbf{not}\,p \\ \mathbf{K}\,p &\leftarrow \mathbf{not}\,q\end{aligned}$$

The two-valued MKNF models of $\mathcal{K}$ are $\{\{p\},\{p,q\}\}$ and $\{\{q\},\{p,q\}\}$, i.e. $\mathbf{K}\,p$ and $\mathbf{not}\,q$ are true in the first model, and $\mathbf{K}\,q$ and $\mathbf{not}\,p$ are true in the second one. We obtain two total three-valued MKNF models: $(\{\{p\},\{p,q\}\},\{\{p\},\{p,q\}\})$ and $(\{\{q\},\{p,q\}\},\{\{q\},\{p,q\}\})$. As we have already seen in Example 5, the only other three-valued MKNF model of $\mathcal{K}$ is $M = (\{\emptyset,\{p\},\{q\},\{p,q\}\},\{\{p,q\}\})$. This MKNF model satisfies the condition given in Definition 12, and M is thus a well-founded MKNF model of $\mathcal{K}$. In fact, M is the only well-founded MKNF model. $\square$

In the rest of this section we lift two important properties proven in [50] from the two-valued MKNF semantics to the new three-valued MKNF semantics. The first property states that $\mathbf{K}$ can be introduced in front of an arbitrary closed MKNF formula φ without changing the three-valued MKNF models of φ.

Proposition 2. *Let σ be a closed MKNF formula and (M,N) an MKNF interpretation pair. Then, (M,N) is a three-valued MKNF model of σ if and only if (M,N) is a three-valued MKNF model of $\mathbf{K}\,\sigma$.*

PROOF. Suppose that (M,N) is a three-valued MKNF model of σ. We know for all $I \in M$ that $(I,\langle M,N\rangle,\langle M,N\rangle)(\sigma) = \mathbf{t}$. So $(I,\langle M,N\rangle,\langle M,N\rangle)(\mathbf{K}\,\sigma) = \mathbf{t}$ holds for all $I \in M$ as well. Since for each (M',N') there is an $I' \in M'$ such that $(I',\langle M',N'\rangle,\langle M,N\rangle)(\sigma) \neq \mathbf{t}$, we also obtain the same for $\mathbf{K}\,\sigma$, and (M,N) is a three-valued MKNF model of $\mathbf{K}\,\sigma$. The converse direction follows in an analogous fashion. $\square$

The second property we adapt from the two-valued to the three-valued MKNF semantics says that grounding a hybrid MKNF knowledge base $\mathcal{K}$ does not affect the three-valued MKNF models of $\mathcal{K}$. This shows that $\mathcal{K}$ and $\mathcal{K}_G$ derive exactly the same consequences.

Lemma 2. *Let $\mathcal{K}$ be a DL-safe hybrid MKNF knowledge base and ψ a ground MKNF formula. Then $\mathcal{K} \models^3_{MKNF} \psi$ if and only if $\mathcal{K}_G \models^3_{MKNF} \psi$.*

PROOF. The argument showing the contrapositive statement $\mathcal{K} \not\models^3_{MKNF} \psi$ if and only if $\mathcal{K}_G \not\models^3_{MKNF} \psi$ is absolutely identical to the one in [50]. So we simply refer to the proof given there. $\square$

4. The Well-Founded MKNF Model

In this section we prove that the well-founded MKNF model is unique, and we define a procedure for computing this unique model. For that purpose, the alternating fixpoint construction of [29, 61] for the well-founded semantics of logic programs is adapted to hybrid MKNF knowledge bases, taking into account possible conflicts resulting from the combination of classical negation in ontologies and nonmonotonic negation in rules.

The well-founded semantics for logic programs, originally defined in [62], only applies to nondisjunctive logic programs, and there is no established well-founded semantics that allows for disjunction in the rule heads (see, e.g., [38]). To avoid the problems encountered when extending the well-founded semantics to disjunctive logic programs, and to be compatible with the well-founded semantics of normal programs, we have to restrict to rules that are nondisjunctive. Thus, our approach is more restrictive in the form of rules than the approach of [52]. Note that we can partially compensate this restriction by using first-order disjunction whenever this is available in the considered DL. Therefore, in the rest of the paper we assume that all hybrid MKNF knowledge bases contain only nondisjunctive rules, i.e. no disjunction occurs in the head of any rule.

We start by adapting partitions from [52] as the means of representing MKNF interpretation pairs. Then, based on that representation, we define operators that allow us to compute a unique model for hybrid MKNF knowledge bases. We show that this model is indeed the (unique) well-founded MKNF model, and we present several important properties including the computational complexity, faithfulness w.r.t. the well-founded semantics of logic programs, and discovery of inconsistencies.

4.1. Partitions of Modal Atoms

As argued in [52], since there are infinitely many two-valued MKNF models of an arbitrary hybrid MKNF knowledge base with a countably infinite domain, working directly with two-valued MKNF models is cumbersome. The same holds for MKNF interpretation pairs in the three-valued semantics presented in Section 3. So, some finite representation is required. The solution, applied in [52] and originally from [11], is to represent a two-valued MKNF model by a finite first-order formula whose set of (first-order) models corresponds to the two-valued MKNF model itself. Intuitively, such a first-order formula is obtained in [52] by first dividing the modal atoms occurring in the ground hybrid MKNF knowledge base into true and false modal atoms, and then constructing the first-order formula from the true modal atoms and the ontology. We extend this construction, and the related notions from [52], to three truth values by partitioning atoms into three sets.

Definition 13. Let $\mathcal{K}_G = (\mathcal{O}, \mathcal{P}_G)$ be a ground, nondisjunctive hybrid MKNF knowledge base. The *set of* **K***-atoms of* $\mathcal{K}_G$, written $\mathsf{KA}(\mathcal{K}_G)$, is the smallest set that contains (i) all ground **K**-atoms occurring in $\mathcal{P}_G$, and (ii) a modal atom $\mathbf{K}\,\xi$ for each ground modal atom $\mathbf{not}\,\xi$ occurring in $\mathcal{P}_G$. A *partial partition*

(T, F) of $\mathsf{KA}(\mathcal{K}_G)$ consists of two sets, where $T, F \subseteq \mathsf{KA}(\mathcal{K}_G)$ and $T \cap F = \emptyset$. A third set U is implicitly defined as $\mathsf{KA}(\mathcal{K}_G) \setminus (T \cup F)$.

The set $\mathsf{KA}(\mathcal{K}_G)$ contains all modal atoms occurring in $\mathcal{K}_G$, only with **not**-atoms substituted by corresponding modal **K**-atoms. This set is partitioned into three sets T, F, and U where, intuitively, T contains true modal atoms, F contains false modal atoms, and U contains all the remaining that are considered to be undefined.

In [52], given a knowledge base $\mathcal{K}_G$, a set of first-order formulas is defined with the aim of using the models of this set of formulas to represent the models of both the ontology and a set of true modal atoms. If this set of true modal atoms is properly chosen, then the set of first-order interpretations satisfying that set of formulas corresponds to one two-valued MKNF model of $\mathcal{K}_G$.

Here, this construction will not suffice, and we show below how to adapt the idea to a three-valued setting. The definition of the set of first-order formulas can be recalled from [52].

Definition 14. Let $\mathcal{K}_G = (\mathcal{O}, \mathcal{P}_G)$ be a ground hybrid MKNF knowledge base. For a subset S of $\mathsf{KA}(\mathcal{K}_G)$, the *objective knowledge* of S w.r.t. $\mathcal{K}_G$ is the set of first-order formulas $\mathsf{OB}_{\mathcal{O},S} = \{\pi(\mathcal{O})\} \cup \{\xi \mid \mathbf{K}\,\xi \in S\}$.

This notion is used below to establish a link between three-valued MKNF models and partial partitions. But for this purpose, we need to adapt one more notion from [52].

Definition 15. Let S be a set of ground modal **K**-atoms. The partial partition (T, F) of S is *induced by* an MKNF interpretation pair (M, N) as follows:

(1) $\mathbf{K}\,\xi \in T$ implies $\forall I \in M : (I, \langle M, N\rangle, \langle M, N\rangle)(\mathbf{K}\,\xi) = \mathbf{t}$,

(2) $\mathbf{K}\,\xi \in F$ implies $\forall I \in M : (I, \langle M, N\rangle, \langle M, N\rangle)(\mathbf{K}\,\xi) = \mathbf{f}$, and

(3) $\mathbf{K}\,\xi \notin T$ and $\mathbf{K}\,\xi \notin F$ implies $\forall I \in M : (I, \langle M, N\rangle, \langle M, N\rangle)(\mathbf{K}\,\xi) = \mathbf{u}$.

Based on this relation, we can show that the objective knowledge derived from the partial partition induced by a three-valued MKNF model is identical to that model. This result is used below to show that the specific partition we compute produces a three-valued MKNF model (Theorem 4).

Proposition 3. *Let (M, N) be a three-valued MKNF model of a ground hybrid MKNF knowledge base $\mathcal{K}_G = (\mathcal{O}, \mathcal{P}_G)$, and (T, F) the partition of $\mathsf{KA}(\mathcal{K}_G)$ induced by (M, N). Then $(M, N) = (\{I \mid I \models \mathsf{OB}_{\mathcal{O},T}\}, \{I \mid I \models \mathsf{OB}_{\mathcal{O}, \mathsf{KA}(\mathcal{K}_G)\setminus F}\})$.*

PROOF. For $\mathcal{K}_G = (\mathcal{O}, \mathcal{P}_G)$ a ground hybrid MKNF knowledge base, let (M, N) be a three-valued MKNF model of $\mathcal{K}_G$, (T, F) the partition of $\mathsf{KA}(\mathcal{K}_G)$ induced by (M, N), and $(M', N') = (\{I \mid I \models \mathsf{OB}_{\mathcal{O},T}\}, \{I \mid I \models \mathsf{OB}_{\mathcal{O}, \mathsf{KA}(\mathcal{K}_G)\setminus F}\})$. We show that $(M, N) = (M', N')$.

First, we show that $M \subseteq M'$. Let I be an interpretation in M. We show that $I \in M' = \{I \mid I \models \mathsf{OB}_{\mathcal{O},T}\}$, i.e. that $I \models \{\pi(\mathcal{O})\} \cup \{\xi \mid \mathbf{K}\,\xi \in T\}$. Since

(M, N) is a three-valued MKNF model of $\mathcal{K}_G$, we know that $(M, N) \models \mathbf{K}\,\pi(\mathcal{O})$. Thus, we have $I \models \pi(\mathcal{O})$. Consider each $\mathbf{K}\,\xi \in T$. Since (M, N) induces the partition (T, F) we have $(I, \langle M, N\rangle, \langle M, N\rangle)(\mathbf{K}\,\xi) = \mathbf{t}$ and thus $I \models \xi$. Hence, $I \models \mathsf{OB}_{\mathcal{O},T}$. Consequently, $I \in M'$ holds and therefore $M \subseteq M'$.

Next, we show that $N \subseteq N'$. Let I be an interpretation in N. We show that $I \in N' = \{I \mid I \models \mathsf{OB}_{\mathcal{O},\mathsf{KA}(\mathcal{K}_G)\setminus F}\}$, i.e. that $I \models \{\pi(\mathcal{O})\} \cup \{\xi \mid \mathbf{K}\,\xi \in \mathsf{KA}(\mathcal{K}_G)\setminus F\}$. We already know that, for each $I \in M$, $I \models \pi(\mathcal{O})$. Since $N \subseteq M$, we also have that $I \models \pi(\mathcal{O})$ for each $I \in N$. Consider each $\mathbf{K}\,\xi \notin F$. The premise of condition (2) in Definition 15 is false, but the premises of conditions (1) and (3) in that definition are true. We show for both cases that $I \models \xi$. This suffices to show that $I \in N'$, i.e. that $I \models \mathsf{OB}_{\mathcal{O},\mathsf{KA}(\mathcal{K}_G)\setminus F}$, which shows $N \subseteq N'$. In the case of (1), we already know that $I \models \xi$ for each $I \in M$, and, since $N \subseteq M$ holds, we also have $I \models \xi$ for each $I \in N$. In the case of (3), we know that $(I, \langle M, N\rangle, \langle M, N\rangle)(\mathbf{K}\,\xi) = \mathbf{u}$ for each $I \in M$. Thus, $I \models \xi$ holds for each $I \in N$.

We now show that each of the two sets are in fact identical, i.e. $M = M'$ and $N = N'$. Note first that $T \subseteq \mathsf{KA}(\mathcal{K}_G) \setminus F$. Thus, for any $I \in N'$, we have $I \in \{I \mid I \models \mathsf{OB}_{\mathcal{O},T}\}$ and therefore $N' \subseteq M'$, i.e. (M', N') is an MKNF interpretation pair. So assume that (M', N') is an MKNF interpretation pair with $M \subseteq M'$ and $N \subseteq N'$, where at least one of the inclusions is proper. We show that $(I', \langle M', N'\rangle, \langle M, N\rangle)(\mathcal{K}_G) = \mathbf{t}$ for all $I' \in M'$, and we thus derive a contradiction to (M, N) being a three-valued MKNF model of $\mathcal{K}_G$. For the former, it suffices to prove that $(I', \langle M', N'\rangle, \langle M, N\rangle)(\mathbf{K}\,\pi(\mathcal{O}) \wedge \pi(\mathcal{P}_G)) = \mathbf{t}$ for all $I' \in M'$. By definition of M' we know that $(I', \langle M', N'\rangle, \langle M, N\rangle)(\mathbf{K}\,\pi(\mathcal{O})) = \mathbf{t}$ for all $I' \in M'$. We only have to show the same for $\pi(\mathcal{P}_G)$. We achieve that by showing that, for each case of Definition 15, the modal atoms appearing in $\pi(\mathcal{P}_G)$ are evaluated to identical truth values in (M, N) and (M', N'). This suffices to show that $(I', \langle M', N'\rangle, \langle M, N\rangle)(\pi(\mathcal{P}_G)) = \mathbf{t}$ for all $I' \in M'$ since (M, N), as a three-valued MKNF model of $\mathcal{K}_G$, ensures that $(I', \langle M, N\rangle, \langle M, N\rangle)(\pi(\mathcal{P}_G)) = \mathbf{t}$. We thus obtain a contradiction to (M, N) being a three-valued MKNF model.

- Consider each $\mathbf{K}\,\xi \in T$. We obtain $(I', \langle M', N'\rangle, \langle M, N\rangle)(\mathbf{K}\,\xi) = \mathbf{t}$ for all $I' \in M'$ by definition of M' just as we have $(I, \langle M, N\rangle, \langle M, N\rangle)(\mathbf{K}\,\xi) = \mathbf{t}$ for all $I \in M$ by Definition 15.

- Consider each $\mathbf{K}\,\xi \in F$. We obtain $(I, \langle M, N\rangle, \langle M, N\rangle)(\mathbf{K}\,\xi) = \mathbf{f}$, by Definition 15. We derive that, by Definition 7, $(I, \langle M, N\rangle, \langle M, N\rangle)(\mathbf{K}\,\xi) = \mathbf{f}$ for some $I \in N$. Because of that, and since $N \subseteq N'$, we also have $(I', \langle M', N'\rangle, \langle M, N\rangle)(\mathbf{K}\,\xi) = \mathbf{f}$ for some $I' \in N'$. Thus, by Definition 7, $(I, \langle M', N'\rangle, \langle M, N\rangle)(\mathbf{K}\,\xi) = \mathbf{f}$.

- Consider each $\mathbf{K}\,\xi$ with $\mathbf{K}\,\xi \notin F$ and $\mathbf{K}\,\xi \notin T$. By Definition 15, we obtain $(I, \langle M, N\rangle, \langle M, N\rangle)(\mathbf{K}\,\xi) = \mathbf{u}$. By definition of N' we have $(I', \langle M', N'\rangle, \langle M, N\rangle)(\mathbf{K}\,\xi) \neq \mathbf{f}$. From $(I, \langle M, N\rangle, \langle M, N\rangle)(\mathbf{K}\,\xi) = \mathbf{u}$ and $M \subseteq M'$ we conclude that only $(I', \langle M', N'\rangle, \langle M, N\rangle)(\mathbf{K}\xi) = \mathbf{u}$ is possible.

- Consider any modal **not**-atom appearing in $\pi(\mathcal{P}_G)$. Since the evaluation

of these is done in both cases w.r.t. (M, N), we straightforwardly obtain the identical evaluation.

$\square$

The following example illustrates the previously introduced notions.

Example 7. Consider $\mathcal{K}$ consisting only of rule (8) from Example 4 and an ontology containing just one assertion:

$$CD(BNAW) \tag{15}$$

The ground KB $\mathcal{K}_G$ contains one rule that results from (8) by substituting x with BNAW. We thus obtain

$$
\begin{aligned}
\mathsf{KA}(\mathcal{K}_G) \quad = \quad & \{\mathbf{K}\,\text{Recommend}(BNAW), \mathbf{K}\,CD(BNAW), \mathbf{K}\,\text{owns}(BNAW), \\
& \mathbf{K}\,\text{lowEval}(BNAW), \mathbf{K}\,\text{interesting}(BNAW)\}.
\end{aligned}
$$

One can easily check that there is only one three-valued MKNF model (M, N) of $\mathcal{K}_G$, namely the one in which each $I \in M, N$ satisfies $I \models CD(BNAW)$. This three-valued MKNF model induces the partition in which $CD(BNAW)$ appears in T and all other modal $\mathbf{K}$-atoms in F. The related set of first-order formulas just contains $CD(BNAW)$. This is reasonable since the ground version of (8) does not allow us to derive anything, and so we can ignore (8) when considering three-valued MKNF models of $\mathcal{K}_G$. $\square$

4.2. Computation of the Alternating Fixpoint

As we have seen in Section 3, a knowledge base may in general have several three-valued MKNF models. But we have a special interest in the least one w.r.t. derivable knowledge – the well-founded MKNF model – and the computation of that model. In order to obtain the well-founded MKNF model and the corresponding partial partition, we resort to several existing relations and correspondences with semantics from Logic Programming.

The stable models of a normal logic program Π are the fixpoints of the Gelfond-Lifschitz operator Γ_Π [19]. The same operator can be used to compute the (three-valued) well-founded model of Π by the so-called alternating fixpoint computation (cf. [61]). Intuitively, an operator, which results from applying Γ_Π twice, is used to compute a least and a greatest fixpoint, which correspond, respectively, to the true and non-false knowledge. The term "alternating" stems from the fact that Γ_Π is antitonic, and so successive applications will, in turn, overestimate and underestimate derivable knowledge in the well-founded model, ultimately alternating between the two fixpoints. More precisely, by iteratively applying Γ_Π starting with an empty set of atoms, we first obtain a set of atoms that includes all the true atoms in the well-founded model of Π, i.e. an overestimate of the true atoms in the well-founded model (in other words, a set whose complement is an underestimate of the set of all false atoms). If we apply Γ_Π

again to that result, then we obtain a set of atoms that are true for sure, i.e. an underestimate of the set of all true atoms. If we continue the iteration, we obtain alternating smaller overestimates and larger underestimates until eventually the iteration alternates between two fixpoints – one with all true atoms, and the other one with all atoms that are true or undefined in the well-founded model of Π.

Since stable models of logic programs and two-valued MKNF models are closely related, we adapt this scheme to hybrid MKNF knowledge bases. We define operators that provide a stable condition for nondisjunctive hybrid MKNF knowledge bases, and we use these operators to obtain an alternating fixpoint that corresponds to the well-founded MKNF model.

We start by defining an operator $T_{\mathcal{K}_G}$ that, given a set of **K**-atoms, draws conclusions from a positive ground hybrid MKNF knowledge base $\mathcal{K}_G$, i.e. a ground hybrid MKNF knowledge base where rules are of the form:

$$\mathbf{K}\,H \leftarrow \mathbf{K}\,A_1, \ldots, \mathbf{K}\,A_n \tag{16}$$

Definition 16. Let $\mathcal{K}_G = (\mathcal{O}, \mathcal{P}_G)$ be a positive, ground hybrid MKNF knowledge base. The operators $R_{\mathcal{K}_G}$, $D_{\mathcal{K}_G}$, and $T_{\mathcal{K}_G}$ are defined on subsets of $\mathsf{KA}(\mathcal{K}_G)$ as follows:

$$
\begin{aligned}
R_{\mathcal{K}_G}(S) &= \{\mathbf{K}\,H \mid \mathcal{P}_G \text{ contains a rule of the form (16)} \\
&\qquad \text{such that, for all } i,\, 1 \le i \le n,\, \mathbf{K}\,A_i \in S\} \\
D_{\mathcal{K}_G}(S) &= \{\mathbf{K}\,\xi \mid \mathbf{K}\,\xi \in \mathsf{KA}(\mathcal{K}_G) \text{ and } \mathsf{OB}_{\mathcal{O},S} \models \xi\} \\
\\
T_{\mathcal{K}_G}(S) &= R_{\mathcal{K}_G}(S) \cup D_{\mathcal{K}_G}(S)
\end{aligned}
$$

The operator $R_{\mathcal{K}_G}$ derives immediate consequences from the rules in $\mathcal{K}_G$ while $D_{\mathcal{K}_G}$ yields consequences from the ontology combined with the already known information in S. The operator $T_{\mathcal{K}_G}$, which combines the other two, is monotonic:

Proposition 4. *Let $\mathcal{K}_G = (\mathcal{O}, \mathcal{P}_G)$ be a positive ground hybrid MKNF knowledge base, and $S \subseteq S' \subseteq \mathsf{KA}(\mathcal{K}_G)$. Then $T_{\mathcal{K}_G}(S) \subseteq T_{\mathcal{K}_G}(S')$.*

PROOF. Suppose that $\mathbf{K}\,H \in T_{\mathcal{K}_G}(S)$. By Definition 16, $\mathbf{K}\,H \in R_{\mathcal{K}_G}(S) \cup D_{\mathcal{K}_G}(S)$ holds. If $\mathbf{K}\,H \in R_{\mathcal{K}_G}(S)$, then $\mathcal{P}_G$ contains a rule of the form (16) such that $\mathbf{K}\,A_i \in S$ for each $1 \le i \le n$. Since $S \subseteq S'$, we also have that $\mathbf{K}\,A_i \in S'$ for each $1 \le i \le n$ and $\mathbf{K}\,H \in T_{\mathcal{K}_G}(S')$. If $\mathbf{K}\,H \in D_{\mathcal{K}_G}(S)$, then $\mathbf{K}\,H \in M$ and $\mathsf{OB}_{\mathcal{O},S} \models H$. By monotonicity of first-order logic and since $S \subseteq S'$, we also have $\mathsf{OB}_{\mathcal{O},S'} \models H$. We conclude that $\mathbf{K}\,H \in T_{\mathcal{K}_G}(S')$. $\qquad\square$

Since $T_{\mathcal{K}_G}$ is monotonic, it has a unique least fixpoint (by the Knaster-Tarski Theorem [60]) which we denote using $T_{\mathcal{K}_G} \uparrow \omega$ in reference to the limit ordinal of natural numbers ω. It is important to note that the Knaster-Tarski Theorem in general only says that this fixpoint is reached for some ordinal that might easily be greater than ω. However, in MKNF knowledge bases, since we do not allow function symbols or infinite sets of rules, the iteration is performed over a

finite knowledge base (with finitely many ground rules). As such, the iteration of $T_{\mathcal{K}_G}$ terminates for some finite ordinal below ω. The least fixpoint is obtained as follows:

$$
\begin{aligned}
T_{\mathcal{K}_G} \uparrow 0 &= \emptyset \\
T_{\mathcal{K}_G} \uparrow (n+1) &= T_{\mathcal{K}_G}(T_{\mathcal{K}_G} \uparrow n) \\
T_{\mathcal{K}_G} \uparrow \omega &= \bigcup_{i \geq 0} T_{\mathcal{K}_G} \uparrow i
\end{aligned}
$$

Similarly to stable models of normal logic programs a fixpoint operator can be defined that performs a Gelfond-Lifschitz-like transformation [19] that turns hybrid MKNF knowledge bases into positive ones, and that then applies the operator $T_{\mathcal{K}_G}$ to the resulting knowledge base.

Definition 17. Let $\mathcal{K}_G = (\mathcal{O}, \mathcal{P}_G)$ be a ground hybrid MKNF knowledge base and $S \subseteq \mathsf{KA}(\mathcal{K}_G)$. The *MKNF transform* $\mathcal{K}_G/S$ is defined as $\mathcal{K}_G/S = (\mathcal{O}, \mathcal{P}_G/S)$, where $\mathcal{P}_G/S$ contains all rules

$$\mathbf{K}\, H \leftarrow \mathbf{K}\, A_1, \ldots, \mathbf{K}\, A_n$$

for which there exists a rule

$$\mathbf{K}\, H \leftarrow \mathbf{K}\, A_1, \ldots, \mathbf{K}\, A_n, \mathbf{not}\, B_1, \ldots, \mathbf{not}\, B_m$$

in $\mathcal{P}_G$ with $\mathbf{K}\, B_j \notin S$ for all $1 \leq j \leq m$.

This definition indeed resembles the transformation used to compute stable models [19] of logic programs. I.e., we remove all rules that contain negated atoms contradicting the given set S, and we remove all remaining negated atoms from the other rules. Following [19], we define an operator that computes the least fixpoint of the resulting knowledge base.

Definition 18. Let $\mathcal{K}_G = (\mathcal{O}, \mathcal{P}_G)$ be a ground hybrid MKNF knowledge base and $S \subseteq \mathsf{KA}(\mathcal{K}_G)$. We define $\Gamma_{\mathcal{K}_G}(S) = T_{\mathcal{K}_G/S} \uparrow \omega$.

Inspired by the similarities between the definition of $\Gamma_{\mathcal{K}_G}$ and Γ in [19], the correspondence of stable models for logic programs and two-valued MKNF models for knowledge bases without ontology axioms, and the results in alternating fixpoints of normal logic programs [61], one might wonder whether iteratively applying the operator $\Gamma_{\mathcal{K}_G}$ would yield the least three-valued MKNF model. In fact, as shown below in Lemma 3, the operator $\Gamma_{\mathcal{K}_G}$ is antitonic. Thus, $\Gamma_{\mathcal{K}_G}^2$ is monotonic and guaranteed to have a least fixpoint, which can be obtained by iteratively applying $\Gamma_{\mathcal{K}_G}^2$ starting from the empty set. One may then ask whether this least fixpoint corresponds to the well-founded MKNF model. However, as shown in the following example, this is not the case, and, thus, an adaptation of alternating fixpoints to hybrid MKNF knowledge bases cannot be as straightforward.

Example 8. Consider the hybrid MKNF knowledge base presented in Examples 2 and 4 for recommending CDs, and suppose now that the user wants to stall recommendations until an evaluation is available. This can be achieved, e.g., by adding the rule (17).

$$\mathbf{K}\,\mathrm{LowEval}(x) \quad \leftarrow \quad \mathbf{not}\,\mathrm{Recommend}(x) \tag{17}$$

With this rule, together with (8), a CD is not recommended unless one adds explicit information that the CD has no low evaluation. To ease the reading, we recall here rule (8):

$$\begin{aligned}
\mathbf{K}\,\mathrm{Recommend}(x) \quad \leftarrow \quad & \mathbf{K}\,\mathrm{CD}(x), \mathbf{not}\,\mathrm{owns}(x), \mathbf{not}\,\mathrm{LowEval}(x), \\
& \mathbf{K}\,\mathrm{interesting}(x).
\end{aligned} \tag{8}$$

In fact, if one adds, e.g.,

$$\neg\mathrm{LowEval}(\mathrm{ToTheSea}) \tag{18}$$

then all three-valued MKNF models contain $\mathbf{K}\,\mathrm{Recommend}(\mathrm{ToTheSea})$. However, as shown next, $\mathbf{K}\,\mathrm{Recommend}(\mathrm{ToTheSea})$ is not contained in the least fixpoint of $\Gamma^2_{\mathcal{K}_G}$.

To simplify the computation and presentation of this least fixpoint, we ground all the rules only with ToTheSea (thus ignoring any other CDs), and we add explicitly that ToTheSea is a CD (19).

$$\mathrm{CD}(\mathrm{ToTheSea}) \tag{19}$$

We also limit ourselves to the following set of modal atoms (using appropriate abbreviations):

$$\mathsf{KA}(\mathcal{K}_G) = \{\mathbf{K}\,\mathrm{Rec}(\mathrm{Tts}), \mathbf{K}\,\mathrm{LowEv}(\mathrm{Tts}), \mathbf{K}\,\mathrm{CD}(\mathrm{Tts}), \mathbf{K}\,\mathrm{owns}(\mathrm{Tts}), \mathbf{K}\,\mathrm{int}(\mathrm{Tts})\}$$

We start with $\mathbf{S}_0 = \emptyset$, so we compute $\Gamma_{\mathcal{K}_G}(\mathbf{S}_0)$ and $\mathbf{S}_1 = \Gamma_{\mathcal{K}_G}(\Gamma_{\mathcal{K}_G}(\mathbf{S}_0))$:

$$\begin{aligned}
\Gamma_{\mathcal{K}_G}(\mathbf{S}_0) \quad &= \quad \mathsf{KA}(\mathcal{K}_G) \\
\mathbf{S}_1 \quad &= \quad \{\mathbf{K}\,\mathrm{CD}(\mathrm{Tts}), \mathbf{K}\,\mathrm{int}(\mathrm{Tts})\}
\end{aligned}$$

Note that, since $\mathbf{K}\,\mathrm{LowEv}(\mathrm{Tts}) \in T_{\mathcal{K}_G/\mathbf{S}_0}(\emptyset)$, then, by (18), $\mathsf{OB}_{\mathcal{O},T_{\mathcal{K}_G/\mathbf{S}_0}(\emptyset)}$ is inconsistent. So, the subsequent application of $D_{\mathcal{K}_G}$ allows us to derive everything, and thus $\Gamma_{\mathcal{K}_G}(\mathbf{S}_0) = \mathsf{KA}(\mathcal{K}_G)$.

We continue with $\Gamma_{\mathcal{K}_G}(\mathbf{S}_1)$ and $\mathbf{S}_2 = \Gamma_{\mathcal{K}_G}(\Gamma_{\mathcal{K}_G}(\mathbf{S}_1))$ and obtain:

$$\begin{aligned}
\Gamma_{\mathcal{K}_G}(\mathbf{S}_1) \quad &= \quad \mathsf{KA}(\mathcal{K}_G) \\
\mathbf{S}_2 \quad &= \quad \{\mathbf{K}\,\mathrm{CD}(\mathrm{Tts}), \mathbf{K}\,\mathrm{int}(\mathrm{Tts})\}
\end{aligned}$$

Now, since $\mathbf{S}_1 = \mathbf{S}_2$, the fixpoint is reached, and, indeed, does not contain $\mathbf{K}\,\mathrm{Rec}(\mathrm{Tts})$. This is so because, since $\mathbf{K}\,\mathrm{LowEv}(\mathrm{Tts}) \in \Gamma_{\mathcal{K}_G}(\mathbf{S}_1)$, rule (8) grounded with Tts is removed in $\mathcal{P}_G/\Gamma_{\mathcal{K}_G}(\mathbf{S}_1)$.

Note that $\mathbf{K}\,\mathrm{LowEv(Tts)} \in \Gamma_{\mathcal{K}_G}(\mathbf{S}_1)$ because rule (17) is not removed in $\mathcal{P}_G/\mathbf{S}_1$, given that $\mathbf{K}\,\mathrm{Rec(Tts)} \notin \mathbf{S}_1$. In an analogy with [61], $\mathbf{K}\,\mathrm{LowEv(Tts)}$ is thus either true or undefined , since it belongs to the overestimate in the alternating fixpoint. This shows that, in opposite to the three-valued MKNF semantics, $\neg\mathrm{LowEv(Tts)}$ does not imply $\mathbf{not}\,\mathrm{LowEv(Tts)}$.[15] In fact, for any three-valued MKNF model (M, N) of the restricted knowledge base, if $\neg\mathrm{LowEv(Tts)}$ holds, then, for all $I \in M$, $\mathrm{LowEv(Tts)} \notin I$. Thus, since $N \subseteq M$, we also have, for all $I \in N$, $\mathrm{LowEv(Tts)} \notin I$, i.e. $\mathbf{not}\,\mathrm{LowEv(Tts)}$ should be true in any three-valued MKNF model of the knowledge base. $\qquad\square$

One way of guaranteeing that the classical negation of some DL-atom H in the ontology imposes the truth of $\mathbf{not}\,H$ (despite the existence of rules with head $\mathbf{K}\,H$) is to change the MKNF transform defined above, so that rules with head $\mathbf{K}\,H$ are removed whenever $\neg H$ holds:

Definition 19. Let $\mathcal{K}_G = (\mathcal{O}, \mathcal{P}_G)$ be a ground hybrid MKNF knowledge base and $S \subseteq \mathsf{KA}(\mathcal{K}_G)$. The *MKNF-coherent transform* $\mathcal{K}_G//S$ is defined as $\mathcal{K}_G//S = (\mathcal{O}, \mathcal{P}_G//_S)$, where $\mathcal{P}_G//_S$ contains all rules

$$\mathbf{K}\,H \leftarrow \mathbf{K}\,A_1, \ldots, \mathbf{K}\,A_n$$

for which there exists a rule

$$\mathbf{K}\,H \leftarrow \mathbf{K}\,A_1, \ldots, \mathbf{K}\,A_n, \mathbf{not}\,B_1, \ldots, \mathbf{not}\,B_m$$

in $\mathcal{P}_G$ with $\mathbf{K}\,B_j \notin S$ for all $1 \leq j \leq m$ and $\mathsf{OB}_{\mathcal{O},S} \not\models \neg H$.

Note the difference between this definition and Definition 17: we also remove a rule from the MKNF-coherent transform, in case the classical negation of the head is derivable from the ontology augmented by S.

Definition 20. Let $\mathcal{K}_G = (\mathcal{O}, \mathcal{P}_G)$ be a ground hybrid MKNF knowledge base and $S \subseteq \mathsf{KA}(\mathcal{K}_G)$. We define $\Gamma'_{\mathcal{K}_G}(S) = T_{\mathcal{K}_G//_S} \uparrow \omega$.

The operator $\Gamma'_{\mathcal{K}_G}$ is also antitonic (cf. Lemma 3), and so applying $\Gamma'_{\mathcal{K}_G}$ twice is guaranteed to have a least fixpoint. Clearly, in the case of the knowledge base of Example 8, this least fixpoint includes $\mathbf{K}\,\mathrm{Rec(Tts)}$. The reason is that the new MKNF-coherent transform does not contain any rule with head $\mathbf{K}\,\mathrm{LowEval(Tts)}$ once $\neg\mathrm{LowEval(Tts)}$ is derived, and so rule (8) instantiated with Tts is not removed at some step of the iteration. However, the next example shows that the operator $\Gamma'_{\mathcal{K}_G}$ literally hides inconsistencies from the iteration. A modal atom $\mathbf{K}\,H$ may be simply considered false, even though there is a rule with head $\mathbf{K}\,H$ such that the body is true in all three-valued MKNF models of the respective knowledge base.

---

[15]This problem is akin to the *coherence problem* [55] in extended logic programs, where a (classical false) formula $\neg\varphi$ has to impose $\mathbf{not}\,\varphi$ explicitly.

Example 9. Consider again only the hybrid MKNF knowledge base presented in Example 2 and 4 for recommending CDs. Now suppose that the user wants to ensure that only inexpensive CDs are to be recommended. Note that this is different from recommending CDs that have a discount. The ontology axiom (20) states that any expensive CD must never be recommended. In general, comparing prices requires some predicates from the numerical domain, such as concrete domains for the DL $\mathcal{EL}^{++}$ [2]. For simplicity, we assume here that this is handled internally, so we simply add a fact (21) saying that ToTheSea is expensive.

$$\text{Expensive} \quad \sqsubseteq \quad \neg\text{Recommend} \tag{20}$$
$$\mathbf{K}\,\text{Expensive(ToTheSea)} \quad \leftarrow \tag{21}$$

This knowledge base is clearly MKNF-inconsistent: simply note that we can conclude that ToTheSea is recommended (from (8) instantiated by ToTheSea) and not recommended at the same time (by (20)–(21)). However, it is easy to check that the least fixpoint of applying $\Gamma'_{\mathcal{K}_G}$ twice does not include $\mathbf{K}\,\text{Rec(Tts)}$, simply because the MKNF-coherent transform removes the rule (8) instantiated with Tts once $\neg\text{Recommend(Tts)}$ is true, even though the rule body is true. $\square$

This example shows that $\Gamma'_{\mathcal{K}_G}$ cannot be applied always in the alternating fixpoint. Examining again the computation of Example 8, we may see that the application of $\Gamma'_{\mathcal{K}_G}$ would only be required when we compute overestimates of the true knowledge. In this case, it would suffice to apply $\Gamma'_{\mathcal{K}_G}$ for computing $\Gamma'_{\mathcal{K}_G}(\mathbf{S}_1)$. Then, $\mathbf{S}_2$ could be obtained by simply applying $\Gamma_{\mathcal{K}_G}$ to the previous result, thus computing $\mathbf{S}_2$ now by $\Gamma_{\mathcal{K}_G}(\Gamma'_{\mathcal{K}_G}(\mathbf{S}_1))$. At the same time, in case of Example 9, the partial usage of $\Gamma_{\mathcal{K}_G}$ would ensure that Recommend(Tts) is kept in one part of the iteration so that inconsistencies may still be detectable.

This suggests that the computation of the well-founded MKNF model could be obtained by alternating the application of the operators $\Gamma'_{\mathcal{K}_G}$ and $\Gamma_{\mathcal{K}_G}$. In fact, as we will show, this interaction of the two operators yields the well-founded MKNF model. But before we formally define this interaction and prove its correspondence to the well-founded MKNF model, we show that both operators are indeed antitonic.

Lemma 3. *If $\mathcal{K}_G$ is a ground hybrid MKNF knowledge base and $S \subseteq S' \subseteq \mathsf{KA}(\mathcal{K}_G)$, then $\Gamma_{\mathcal{K}_G}(S') \subseteq \Gamma_{\mathcal{K}_G}(S)$ and $\Gamma'_{\mathcal{K}_G}(S') \subseteq \Gamma'_{\mathcal{K}_G}(S)$.*

PROOF. We show the argument for $\Gamma_{\mathcal{K}_G}$. The proof for $\Gamma'_{\mathcal{K}_G}$ is identical.

By Definition 18, we have to show that $T_{\mathcal{K}_G/S'} \uparrow \omega \subseteq T_{\mathcal{K}_G/S} \uparrow \omega$. We prove by induction on n that $T_{\mathcal{K}_G/S'} \uparrow n \subseteq T_{\mathcal{K}_G/S} \uparrow n$ holds. The base case for $n = 0$ is trivial since $\emptyset \subseteq \emptyset$. Assume that $T_{\mathcal{K}_G/S'} \uparrow n \subseteq T_{\mathcal{K}_G/S} \uparrow n$ holds and consider $\mathbf{K}\,H \in T_{\mathcal{K}_G/S'} \uparrow (n+1)$. Then $\mathbf{K}\,H \in T_{\mathcal{K}_G/S'}(T_{\mathcal{K}_G/S'} \uparrow n)$ and there are two cases to consider. First, $\mathcal{K}_G/S'$ contains a rule of the form $\mathbf{K}\,H \leftarrow \mathbf{K}\,A_1, \ldots, \mathbf{K}\,A_n$ such that $\mathbf{K}\,A_i \in T_{\mathcal{K}_G/S'} \uparrow n$ for each $1 \le i \le n$. Since $S \subseteq S'$, we also have $\mathbf{K}\,H \leftarrow \mathbf{K}\,A_1, \ldots, \mathbf{K}\,A_n$ in $\mathcal{K}_G/S$ and, by the induction hypothesis, $\mathbf{K}\,A_i \in T_{\mathcal{K}_G/S} \uparrow n$ for each $1 \le i \le n$. Hence, $\mathbf{K}\,H \in T_{\mathcal{K}_G/S} \uparrow (n+1)$.

Alternatively, $\mathbf{K}\,H$ is a consequence obtained from $D_{\mathcal{K}_G/S'}(T_{\mathcal{K}_G/S'} \uparrow n)$. By the induction hypothesis, $T_{\mathcal{K}_G/S'} \uparrow n \subseteq T_{\mathcal{K}_G/S} \uparrow n$ holds, and we conclude from the monotonicity of first-order logic that $\mathbf{K}\,H \in D_{\mathcal{K}_G/S}(T_{\mathcal{K}_G/S} \uparrow n)$. $\qquad\square$

Since both operators are antitonic, we can define an alternating iteration for the two operators as motivated above:

Definition 21. Let $\mathcal{K}_G$ be a ground hybrid MKNF knowledge base. We define two sequences $\mathbf{P}_i$ and $\mathbf{N}_i$ as follows.

$$\begin{aligned}
\mathbf{P}_0 &= \emptyset & \mathbf{N}_0 &= \mathsf{KA}(\mathcal{K}_G) \\
\mathbf{P}_{n+1} &= \Gamma_{\mathcal{K}_G}(\mathbf{N}_n) & \mathbf{N}_{n+1} &= \Gamma'_{\mathcal{K}_G}(\mathbf{P}_n) \\
\mathbf{P}_\omega &= \bigcup \mathbf{P}_i & \mathbf{N}_\omega &= \bigcap \mathbf{N}_i
\end{aligned}$$

The sequence of $\mathbf{P}_i$ is intended to compute modal atoms that are true, while the sequence $\mathbf{N}_i$ computes modal atoms that are not false. The former is an increasing sequence, while the latter is decreasing:

Lemma 4. *Let $\mathcal{K}_G$ be a ground hybrid MKNF knowledge base. Then $\mathbf{P}_\alpha \subseteq \mathbf{P}_\beta$ and $\mathbf{N}_\beta \subseteq \mathbf{N}_\alpha$ for all ordinals α, β with $\alpha \leq \beta \leq \omega$.*

PROOF. Whenever $\alpha = \beta$, the statement holds automatically. It thus suffices to consider $\alpha < \omega$ and to show via induction over α that the statement holds. If β is a successor ordinal, then it is sufficient to show the property for $\beta = \alpha + 1$, all the other successor cases follow by transitivity of $\subseteq$.

If $\alpha = 0$, then $\mathbf{P}_0 = \emptyset$ and $\mathbf{P}_0 \subseteq \mathbf{P}_\beta$ holds for any β. Equivalently, $\mathbf{N}_0 = \mathsf{KA}(\mathcal{K}_G)$, thus $\mathbf{N}_\beta \subseteq \mathbf{N}_0$ also holds for any β.

Suppose the property holds for all $\alpha \leq n$. We must show that $\mathbf{P}_{n+1} \subseteq \mathbf{P}_{n+2}$ and $\mathbf{N}_{n+2} \subseteq \mathbf{N}_{n+1}$. We have $\mathbf{P}_{n+1} = \Gamma(\mathbf{N}_n)$ and $\mathbf{P}_{n+2} = \Gamma(\mathbf{N}_{n+1})$. Since $\mathbf{N}_{n+1} \subseteq \mathbf{N}_n$ by the induction hypothesis, $\mathbf{P}_{n+1} \subseteq \mathbf{P}_{n+2}$ holds in virtue of the antimonotonicity of Γ'. Likewise, we know that $\mathbf{N}_{n+1} = \Gamma'(\mathbf{P}_n)$ and $\mathbf{N}_{n+2} = \Gamma'(\mathbf{P}_{n+1})$. Since $\mathbf{P}_n \subseteq \mathbf{P}_{n+1}$ by the induction hypothesis, we obtain by antitonicity of Γ' that $\mathbf{N}_{n+2} \subseteq \mathbf{N}_{n+1}$.

The only case left is the one where $\beta = \omega$. But this case holds by definition. $\qquad\square$

Like the iteration of $T_\mathcal{K}$, and for the very same reasons, these iterations are finite and reach a fixpoint before ω – in the case of $\mathbf{P}_i$ a least fixpoint, and in the case of $\mathbf{N}_i$ a greatest fixpoint:

Proposition 5. *Let $\mathcal{K}_G$ be a ground hybrid MKNF knowledge base. Then $\mathbf{P}_\omega$ is the least fixpoint of the sequence of $\mathbf{P}_i$ and $\mathbf{N}_\omega$ is the greatest fixpoint of the sequence of $\mathbf{P}_i$.*

PROOF. We show the argument for $\mathbf{P}_\omega$. The argument for $\mathbf{N}_\omega$ is analogous.

We define an operator $\Phi(S) = \Gamma_{\mathcal{K}_G}(\Gamma'_{\mathcal{K}_G}(S))$ on subsets S of $\mathsf{KA}(\mathcal{K}_G)$, iterated as usual. It is easy to see that $\Phi \uparrow i = \mathbf{P}_{2i}$ and, thus, that Φ is monotonic. By the Knaster-Tarski Theorem we conclude that $\mathbf{P}_\omega$ is equal to the least fixpoint of the sequence of $\mathbf{P}_i$. $\qquad\square$

This proposition also allows us to show that the least fixpoint can be directly computed from the greatest one and vice versa.

Proposition 6. *Let $\mathcal{K}_G$ be a ground hybrid MKNF knowledge base. Then $\mathbf{P}_\omega = \Gamma_{\mathcal{K}_G}(\mathbf{N}_\omega)$ and $\mathbf{N}_\omega = \Gamma'_{\mathcal{K}_G}(\mathbf{P}_\omega)$.*

PROOF. We show the case of $\mathbf{N}_\omega = \Gamma'(\mathbf{P}_\omega)$; the other case proceeds identically. By Proposition 5, we know that $\mathbf{P}_\omega$ is the least fixpoint of the sequence of $\mathbf{P}_i$. Since the ground knowledge base is finite, there is an n such that $\mathbf{P}_n = \mathbf{P}_\omega$, and so $\mathbf{P}_n = \mathbf{P}_m$ for any $m \geq n$. Subsequently, we have $\mathbf{N}_{n+1} = \mathbf{N}_m$ for any m with $m \geq n+1$, i.e. $\mathbf{N}_{n+1} = \Gamma'_{\mathcal{K}_G}(\mathbf{P}_\omega)$ is a fixpoint of the sequence $\mathbf{N}_i$. Assume that $\mathbf{N}_{n+1}$ is not the greatest fixpoint. Then there is an $\mathbf{N}_l$, $l < n+1$, with $\mathbf{N}_l = \mathbf{N}_{l+2}$ and $\mathbf{N}_l \supset \mathbf{N}_{n+1}$. Then $\mathbf{P}_{l+1}$ also equals a fixpoint in the sequence $\mathbf{P}_i$ with $\mathbf{P}_{l+1}$ being necessarily smaller than $\mathbf{P}_n$. This contradicts the initial assumption that $\mathbf{P}_n$ is the least fixpoint and finishes the proof. $\qquad\square$

Thus, we can either compute the two sequences $\mathbf{P}_i$ and $\mathbf{N}_i$ in parallel until we reach an n such that $\mathbf{P}_n = \mathbf{P}_{n+1}$ and $\mathbf{N}_n = \mathbf{N}_{n+1}$ or we compute just one of the two fixpoints in the manner sketched in the proof of Proposition 5 (alternating between $\Gamma_{\mathcal{K}_G}$ and $\Gamma'_{\mathcal{K}_G}$) and let the other one follow by one application of either $\Gamma_{\mathcal{K}_G}$ or $\Gamma'_{\mathcal{K}_G}$.

The two fixpoints can be used to define the well-founded partition which is, as we show in Section 4.3, the partition inducing the well-founded MKNF model.

Definition 22. The *well-founded partition* of an MKNF-consistent ground hybrid MKNF knowledge base $\mathcal{K}_G = (\mathcal{O}, \mathcal{P}_G)$ is defined by:

$$(T_W, F_W) = (\mathbf{P}_\omega, \mathsf{KA}(\mathcal{K}_G) \setminus \mathbf{N}_\omega)$$

Note that we restrict the definition to MKNF-consistent hybrid MKNF knowledge bases. This is reasonable since in many cases the pair (T_W, F_W) obtained for an MKNF-inconsistent knowledge base would not satisfy the conditions imposed in the definition of a partition (cf. Definition 13). Therefore, in Section 4.3, we show that all modal **K**-atoms derived in $\mathbf{P}_\omega$, $\mathsf{KA}(\mathcal{K}_G) \setminus \mathbf{N}_\omega$ respectively, are true, false respectively, in all three-valued MKNF models of $\mathcal{K}_G$ (see Proposition 7), including the special case, in which $\mathcal{K}_G$ is MKNF-inconsistent. This can be used to present necessary and sufficient conditions to check for MKNF-consistency (see Theorem 2), which are based on two comparisons, each of which compares a further iteration of the operators $\Gamma_{\mathcal{K}_G}$ and $\Gamma'_{\mathcal{K}_G}$ w.r.t. one of the fixpoints. Given an established check for MKNF-consistency, we can show that the well-founded partition is in fact a partial partition (see Proposition 8), if the considered knowledge base $\mathcal{K}_G$ is consistent. In this case, we can also show that a corresponding MKNF interpretation pair exists that satisfies $\mathcal{K}_G$ (see Theorem 3), and that this interpretation pair is a three-valued MKNF model of $\mathcal{K}_G$ (see Theorem 4). This allows us to conclude that this specific MKNF interpretation pair corresponding to the well-founded partition

is the well-founded MKNF model (see Theorem 5). We can then show that, given $\mathcal{K}_G$ with empty $\mathcal{O}$, the well-founded partition and the well-founded model for logic programs coincide (see Theorem 6) and finish with the results for data complexity (see Theorem 7). But before we come to that (in Section 4.3), we illustrate the alternating fixpoint construction in the two motivating examples (Examples 8, 9) presented before.

Example 10. Consider the hybrid MKNF knowledge base presented in Example 8, with the same limitation on the set of modal atoms.

For similarity with the computation presented in Example 8, we compute the fixpoints as sketched in the proof of Proposition 5. We start with $\mathbf{P}_0 = \emptyset$ and compute $\mathbf{N}_1 = \Gamma'_{\mathcal{K}_G}(\mathbf{P}_0)$ and $\mathbf{P}_2 = \Gamma_{\mathcal{K}_G}(\mathbf{N}_1)$:

$$
\begin{aligned}
\mathbf{N}_1 &= \{\mathbf{K}\,\mathrm{CD(Tts)}, \mathbf{K}\,\mathrm{int(Tts)}, \mathbf{K}\,\mathrm{Rec(Tts)}\} \\
\mathbf{P}_2 &= \{\mathbf{K}\,\mathrm{CD(Tts)}, \mathbf{K}\,\mathrm{int(Tts)}, \mathbf{K}\,\mathrm{Rec(Tts)}\}
\end{aligned}
$$

It is easy to check that $\mathbf{P}_2$ is already the least fixpoint. Note the difference to the iteration in Example 8. Now, $\mathbf{K}\,\mathrm{LowEv(Tts)}$ does not occur in $\mathbf{N}_1$, since $\mathsf{OB}_{\mathcal{O},P_0} \models \neg\mathrm{LowEv(Tts)}$. So rule (17) instantiated with Tts is removed in the MKNF coherent-transform, and thus $\mathbf{K}\,\mathrm{LowEv(Tts)} \notin \Gamma'_{\mathcal{K}_G}(\mathbf{P}_0)$. As a consequence, we obtain $\mathbf{K}\,\mathrm{Rec(Tts)}$ in $\mathbf{P}_2$. We can compute the greatest fixpoint $\mathbf{N}_\omega = \Gamma'_{\mathcal{K}_G}(\mathbf{P}_\omega)$, and we obtain that $\mathbf{N}_\omega$ equals $\mathbf{N}_1$. Note that if axiom (18) is omitted, then both $\mathbf{K}\,\mathrm{LowEv(Tts)}$ and $\mathbf{K}\,\mathrm{Rec(Tts)}$ remain undefined. Thus, operator $\Gamma'_{\mathcal{K}_G}$, in combination with (18), shows how the formula $\neg\mathrm{LowEv(Tts)}$ imposes that **not** $\mathrm{LowEv(Tts)}$ holds, ensuring in this example the derivability of $\mathbf{K}\,\mathrm{Rec(Tts)}$. $\qquad\square$

The knowledge base in Example 9 is MKNF-inconsistent. Thus, there cannot be a well-founded partition as in Definition 22. Nevertheless, we present the computation of the alternating fixpoint, in order to show the difference to the computation in Example 9, and to hint on how to detect inconsistencies, a topic that is detailed in the next section.

Example 11. Consider the ground hybrid MKNF knowledge base $\mathcal{K}_G$ presented in Example 9, where all rules are only grounded with Tts, and the following restricted set of modal atoms is used:

$$
\begin{aligned}
\mathsf{KA}(\mathcal{K}_G) = \ &\{\mathbf{K}\,\mathrm{Rec(Tts)}, \mathbf{K}\,\mathrm{CD(Tts)}, \mathbf{K}\,\mathrm{LowEv(Tts)}, \mathbf{K}\,\mathrm{owns(Tts)}, \\
&\ \mathbf{K}\,\mathrm{int(Tts)}, \mathbf{K}\,\mathrm{Exp(Tts)}\}
\end{aligned}
$$

To further simplify the presentation, we only consider rule (8), axioms (19)–(21), and we simplify (9) to the fact (26). To ease the reading we repeat here

the complete knowledge base obtained after all simplifications:

$$\text{Exp} \quad \sqsubseteq \quad \neg\text{Rec} \tag{22}$$

$$\text{CD}(\text{Tts}) \tag{23}$$

$$\mathbf{K}\,\text{Exp}(\text{Tts}) \quad \leftarrow \tag{24}$$

$$\mathbf{K}\,\text{Rec}(\text{Tts}) \quad \leftarrow \quad \mathbf{K}\,\text{CD}(\text{Tts}), \mathbf{not}\,\text{owns}(Tts), \mathbf{not}\,\text{LowEval}(Tts),$$
$$\mathbf{K}\,\text{int}(Tts). \tag{25}$$

$$\mathbf{K}\,\text{int}(\text{Tts}) \quad \leftarrow \tag{26}$$

For computing the two fixpoints, we start with $\mathbf{P}_0 = \emptyset$ and $\mathbf{N}_0 = \mathsf{KA}(\mathcal{K}_G)$. We continue with $\mathbf{P}_1 = \Gamma_{\mathcal{K}_G}(\mathbf{N}_0)$ and $\mathbf{N}_1 = \Gamma'_{\mathcal{K}_G}(\mathbf{P}_0)$:

$$\mathbf{P}_1 \quad = \quad \{\mathbf{K}\,\text{CD}(\text{Tts}), \mathbf{K}\,\text{int}(\text{Tts}), \mathbf{K}\,\text{Exp}(\text{Tts})\}$$
$$\mathbf{N}_1 \quad = \quad \mathsf{KA}(\mathcal{K}_G)$$

Note that once $\mathbf{K}\,\text{Rec}(\text{Tts})$ is derived in the computation of $\mathbf{N}_1$ and added to the set S of derived knowledge of $T_{\mathcal{K}_G/\emptyset}$, then $D_{\mathcal{K}_G/\emptyset}$ allows us to derive everything, simply because $\mathsf{OB}_{\mathcal{O},S}$ with $\{\mathbf{K}\,\text{Exp}(\text{Tts}), \mathbf{K}\,\text{Rec}(\text{Tts})\} \subseteq S$ is inconsistent.

We continue with $\mathbf{P}_2 = \Gamma_{\mathcal{K}_G}(\mathbf{N}_1)$ and $\mathbf{N}_2 = \Gamma'_{\mathcal{K}_G}(\mathbf{P}_1)$, and obtain:

$$\mathbf{P}_2 \quad = \quad \{\mathbf{K}\,\text{CD}(\text{Tts}), \mathbf{K}\,\text{int}(\text{Tts}), \mathbf{K}\,\text{Exp}(\text{Tts})\}$$
$$\mathbf{N}_2 \quad = \quad \{\mathbf{K}\,\text{CD}(\text{Tts}), \mathbf{K}\,\text{int}(\text{Tts}), \mathbf{K}\,\text{Exp}(\text{Tts})\}$$

Since $\mathsf{OB}_{\mathcal{O},P_1} \models \neg\text{Rec}(\text{Tts})$, the rule (25) no longer appears in the transform used for computing $\mathbf{N}_2$, and the explosive behaviour of $D_{\mathcal{K}_G/\emptyset}$ disappears as well. As a consequence, in the next iteration we obtain $\mathbf{K}\,\text{Rec}(\text{Tts}) \in \mathbf{P}_3$, which again yields the explosive inconsistency and the derivation of $\mathsf{KA}(\mathcal{K}_G)$.

$$\mathbf{P}_3 \quad = \quad \mathsf{KA}(\mathcal{K}_G)$$
$$\mathbf{N}_3 \quad = \quad \{\mathbf{K}\,\text{CD}(\text{Tts}), \mathbf{K}\,\text{int}(\text{Tts}), \mathbf{K}\,\text{Exp}(\text{Tts})\}$$

It is easy to check that these are the fixpoints. We have $\mathbf{K}\,\text{Rec}(\text{Tts}) \in \mathbf{P}_3$ but $\mathbf{K}\,\text{Rec}(\text{Tts}) \notin \mathbf{N}_3$. Intuitively, this means that $\mathbf{K}\,\text{Rec}(\text{Tts})$ is true and false at the same time, something that is already a clear indication for the inconsistency of the considered knowledge base. □

4.3. The Well-Founded MKNF Model and Related Properties

The well-founded partition (T_W, F_W) consists of modal atoms that are intended to be true (T_W), false (F_W) or undefined (those modal atoms neither occurring in T_W nor in F_W). But this is not merely an intention. The two sequences of $\mathbf{P}_i$ and $\mathbf{N}_i$ allow us to show that any modal atom that is added to an element of the sequence of $\mathbf{P}_i$ (resp. removed from an element of the sequence of $\mathbf{N}_i$) must be true in all three-valued MKNF models of $\mathcal{K}_G$ (resp. false). For that purpose we need to define a notion of dependency that captures more precisely the derivations from $\mathsf{OB}_{\mathcal{O},S}$, for some S, by the operator $D_{\mathcal{K}_G}$.

Definition 23. Let $\mathcal{K}_G$ be a ground hybrid MKNF knowledge base, $\mathbf{K}\,H$ a modal $\mathbf{K}$-atom with $\mathbf{K}\,H \in \mathsf{KA}(\mathcal{K}_G)$, and S a (possibly empty) set of modal $\mathbf{K}$-atoms with $S \subseteq \mathsf{KA}(\mathcal{K}_G)$. We say that $\mathbf{K}\,H$ *depends* on S if and only if

(i) $\mathsf{OB}_{\mathcal{O},S} \models H$ and

(ii) there is no S' with $S' \subset S$ such that $\mathsf{OB}_{\mathcal{O},S'} \models H$.

Intuitively, S is a minimal set that, in combination with $\mathcal{O}$, allows us to derive $\mathbf{K}\,H$. Note that there may exist several such minimal sets. Furthermore, S may also be minimal such that $\mathsf{OB}_{\mathcal{O},S}$ is inconsistent. An appropriate restriction in the proof of the following proposition then ensures that this notion can be used properly.

Proposition 7. *Let $\mathcal{K}_G$ be a ground hybrid MKNF knowledge base and (T, F) the pair $(\mathbf{P}_\omega, \mathsf{KA}(\mathcal{K}_G) \setminus \mathbf{N}_\omega)$. Then $\mathbf{K}\,H \in T$ implies that $\mathbf{K}\,H$ is true (and $\mathbf{not}\,H$ is false) in all three-valued MKNF models (M, N) of $\mathcal{K}_G$, and $\mathbf{K}\,H \in F$ implies that $\mathbf{K}\,H$ is false (and $\mathbf{not}\,H$ is true) in all three-valued MKNF models (M, N) of $\mathcal{K}_G$.*

PROOF. According to Proposition 5, we have to show that, for all i, $\mathbf{K}\,H \in \mathbf{P}_i$ implies that $\mathbf{K}\,H$ is true (and $\mathbf{not}\,H$ is false) in all three-valued MKNF models (M, N) of $\mathcal{K}_G$, and $\mathbf{K}\,H \notin \mathbf{N}_i$ implies that $\mathbf{K}\,H$ is false (and $\mathbf{not}\,H$ is true) in all three-valued MKNF models (M, N) of $\mathcal{K}_G$. We show the argument for $\mathbf{K}\,H$ by an induction on i. This also shows the argument for $\mathbf{not}\,H$ since, for all partial MKNF models (M, N) of any given $\mathcal{K}$, we have that $(I, \langle M, N \rangle, \langle M, N \rangle)(\mathbf{K}\,H) = \neg(I, \langle M, N \rangle, \langle M, N \rangle)(\mathbf{not}\,H)$.

The base case $i = 0$ trivially holds, since $\mathbf{P}_0$ is empty and $\mathbf{N}_0$ is equal to $\mathsf{KA}(\mathcal{K}_G)$.

(i) Suppose that the property holds for all $i \leq n$. We consider $i = n + 1$ for two cases, namely $\mathbf{K}\,H \in \mathbf{P}_{n+1}$ and $\mathbf{K}\,H \notin \mathbf{N}_{n+1}$.

Let $\mathbf{K}\,H \in \mathbf{P}_{n+1}$. If $\mathbf{K}\,H$ already occurs in $\mathbf{P}_n$, then $\mathbf{K}\,H$ is true in all three-valued MKNF models (M, N) of $\mathcal{K}_G$, by the induction hypothesis (i). Otherwise, $\mathbf{K}\,H \in \Gamma_{\mathcal{K}_G}(\mathbf{N}_n)$, i.e. $\mathbf{K}\,H \in T_{\mathcal{K}_G/\mathbf{N}_n} \uparrow \omega$ but $\mathbf{K}\,H \notin \mathbf{P}_n$. Since $\mathbf{K}\,H$ is introduced by $T_{\mathcal{K}_G/\mathbf{N}_n} \uparrow \omega$, we know that $\mathbf{K}\,H \in T_{\mathcal{K}_G/\mathbf{N}_n} \uparrow j$ for some j, and we show by induction on j that $\mathbf{K}\,H$ is true in all three-valued MKNF models (M, N) of $\mathcal{K}_G$.

The base case holds trivially, since $T_{\mathcal{K}_G/\mathbf{N}_n} \uparrow 0$ is empty.

(ii) Suppose that the claim holds for all $j \leq m$, and consider $\mathbf{K}\,H \in T_{\mathcal{K}_G/\mathbf{N}_n} \uparrow m + 1$.

If $\mathbf{K}\,H$ already occurs in $T_{\mathcal{K}_G/\mathbf{N}_n} \uparrow m$, then the claim holds automatically by the induction hypothesis (ii). Otherwise, there are two cases to consider. Either there is a positive rule $\mathbf{K}\,H \leftarrow \mathbf{K}\,A_1, \ldots \mathbf{K}\,A_n$ in $\mathcal{K}_G/\mathbf{N}_n$ with $\mathbf{K}\,A_i \in T_{\mathcal{K}_G/\mathbf{N}_n} \uparrow m$, or $\mathbf{K}\,H$ is the consequence of $D_{\mathcal{K}_G/\mathbf{N}_n}(T_{\mathcal{K}_G/\mathbf{N}_n} \uparrow m)$. In the first case, by the induction hypothesis (ii), all $\mathbf{K}\,A_i$ are true in all three-valued MKNF models (M, N) of $\mathcal{K}_G$. Additionally, there is a rule $\mathbf{K}\,H \leftarrow \mathbf{K}\,A_1, \ldots \mathbf{K}\,A_n, \mathbf{not}\,B_1, \ldots, \mathbf{not}\,B_m$ in $\mathcal{K}_G$, and since the positive version of this

rule occurs in $\mathcal{K}_G/\mathbf{N}_n$, no $\mathbf{K}\,B_j$ occurs in $\mathbf{N}_n$, and thus (by the induction hypothesis (i)), all $\mathbf{K}\,B_j$ are false in all three-valued MKNF models (M, N) of $\mathcal{K}_G$. Consequently, $\mathbf{K}\,H$ has to be true in all three-valued MKNF models (M, N) of $\mathcal{K}_G$. In the second case, $\mathsf{OB}_{\mathcal{O},S} \models H$ with $S = T_{\mathcal{K}_G/\mathbf{N}_n} \uparrow m$ holds. Since $\mathcal{O}$ and all modal atoms occurring in $T_{\mathcal{K}_G/\mathbf{N}_n} \uparrow m$ are true in all three-valued MKNF models of $\mathcal{K}_G$ (by the induction hypothesis (ii)), we can immediately conclude that $\mathbf{K}\,H$ also has to be true in all three-valued MKNF models (M, N) of $\mathcal{K}_G$.

Alternatively, consider all $\mathbf{K}\,H \notin \mathbf{N}_{n+1}$, i.e. all $\mathbf{K}\,H \notin \Gamma'_{\mathcal{K}_G}(\mathbf{P}_n)$. Let U be the set of all such $\mathbf{K}\,H \notin \Gamma'_{\mathcal{K}_G}(\mathbf{P}_n)$. Note that $\mathsf{OB}_{\mathcal{O},\mathbf{P}_n}$ has to be consistent, otherwise U would be empty. Now, from the definition of $\Gamma'_{\mathcal{K}_G}(\mathbf{P}_n)$ we can conclude that, for each modal $\mathbf{K}$-atom $\mathbf{K}\,H$ in U, the following conditions are satisfied:[16]

(Ui) for each rule $\mathbf{K}\,H \leftarrow \mathsf{body}$ in $\mathcal{P}_G$ at least one of the following holds.

  (Uia) Some modal $\mathbf{K}$-atom $\mathbf{K}\,A$ appears in body and in $U \cup \mathsf{KA}(\mathcal{K}_G) \setminus \mathbf{N}_n$.

  (Uib) Some modal $\mathbf{not}$-atom $\mathbf{not}\,B$ appears in body and in $\mathbf{P}_n$.

  (Uic) $\mathsf{OB}_{\mathcal{O},\mathbf{P}_n} \models \neg H$

(Uii) for each (possibly empty) S with $S \subseteq \mathbf{P}_n$, on which $\mathbf{K}\,H$ depends, there is at least one modal $\mathbf{K}$-atom $\mathbf{K}\,A$ such that $\mathsf{OB}_{\mathcal{O},S \setminus \{\mathbf{K}\,A\}} \not\models H$ and $\mathbf{K}\,A$ in $U \cup \mathsf{KA}(\mathcal{K}_G) \setminus \mathbf{N}_n$.

The cases of (Uib) and (Uic) correspond to the removal of rules performed for the MKNF-coherent transform (cf. Definition 19). By the induction hypothesis (i), we know that, for all $\mathbf{K}\,B \in \mathbf{P}_n$, $\mathbf{K}\,B$ is true for all three-valued MKNF models of $\mathcal{K}_G$. Thus, in case of (Uib), body is false in all three-valued MKNF models of $\mathcal{K}_G$, and, in case of (Uic), $\neg H$ is true in all three-valued MKNF models of $\mathcal{K}_G$. In case of (Uia), the corresponding rule with head $\mathbf{K}\,H$ still appears in the MKNF-coherent transform but some modal $\mathbf{K}$-atom $\mathbf{K}\,A$ is not derivable and, thus, $\mathbf{K}\,H$ is not derivable either. Likewise, in case of (Uii), some modal $\mathbf{K}$-atom $\mathbf{K}\,A$ is not (any longer) derivable, so that $\mathbf{K}\,H$ cannot be derived from the ontology. If $\mathbf{K}\,A$ appears in $\mathsf{KA}(\mathcal{K}_G) \setminus \mathbf{N}_n$, then, by the induction hypothesis (i), we know that $\mathbf{K}\,A$ is false in all three-valued MKNF models of $\mathcal{K}_G$. The only remaining case are those $\mathbf{K}\,A$ in U. Assume first that all conditions are satisfied without reference to U. Then, clearly, all modal $\mathbf{K}$-atoms have to be false in all three-valued MKNF models of $\mathcal{K}_G$. Now, consider the complete conditions (Uia)–(Uii): since all $\mathbf{K}\,H$ satisfy the conditions (Uia)–(Uii), and since three-valued MKNF models minimize derivable knowledge in the order $\mathbf{t} > \mathbf{u} > \mathbf{f}$, we derive that all $\mathbf{K}\,H$ are false in all three-valued MKNF models of $\mathcal{K}_G$. $\qquad\square$

For an MKNF-consistent knowledge base $\mathcal{K}_G$, the pair (T, F) in Proposition 7 is defined exactly in the same way as the well-founded partition, and we show below that this correspondence indeed holds. Of course, there is still the

---

[16]These conditions resemble the notion of unfounded sets in [62].

issue of determining, based on the iterations and the consistency of $\mathcal{O}$ alone, whether or not the knowledge base is MKNF-consistent. The next theorem presents the necessary and sufficient conditions for MKNF-inconsistency:

Theorem 2. *Let $\mathcal{K}_G = (\mathcal{O}, \mathcal{P}_G)$ be a ground hybrid MKNF knowledge base, $\mathbf{P}_\omega$ the fixpoint of the sequence $\mathbf{P}_i$, and $\mathbf{N}_\omega$ the fixpoint of the sequence $\mathbf{N}_i$. $\mathcal{K}_G$ is MKNF-inconsistent iff $\Gamma'_{\mathcal{K}_G}(\mathbf{P}_\omega) \subset \Gamma_{\mathcal{K}_G}(\mathbf{P}_\omega)$ or $\Gamma'_{\mathcal{K}_G}(\mathbf{N}_\omega) \subset \Gamma_{\mathcal{K}_G}(\mathbf{N}_\omega)$ or $\mathcal{O}$ is inconsistent.*

PROOF. First, we show that if any of the three conditions holds, then $\mathcal{K}_G$ is MKNF-inconsistent. For the two cases w.r.t. $\mathbf{N}_\omega$ and $\mathbf{P}_\omega$, we present the proof for $\mathbf{N}_\omega$. The other case can be proven analogously.

From Proposition 6 we know that $\Gamma_{\mathcal{K}_G}(\mathbf{N}_\omega) = \mathbf{P}_\omega$. Furthermore, by Proposition 7, we have that all modal atoms $\mathbf{K}\,H \in \mathbf{P}_\omega$ are true in all three-valued MKNF models (M, N) of $\mathcal{K}_G$. If $\Gamma'_{\mathcal{K}_G}(\mathbf{N}_\omega) \subset \Gamma_{\mathcal{K}_G}(\mathbf{N}_\omega)$, then there is at least one $\mathbf{K}\,H$ such that $\mathbf{K}\,H \in \Gamma_{\mathcal{K}_G}(\mathbf{N}_\omega) \setminus \Gamma'_{\mathcal{K}_G}(\mathbf{N}_\omega)$. The only reason for $\mathbf{K}\,H$ not to occur in $\Gamma'_{\mathcal{K}_G}(\mathbf{N}_\omega)$ is that there is a modal $\mathbf{K}$-atom $\mathbf{K}\,A$ such that $\mathbf{K}\,A \in \Gamma_{\mathcal{K}_G}(\mathbf{N}_\omega) \setminus \Gamma'_{\mathcal{K}_G}(\mathbf{N}_\omega)$ and $\mathsf{OB}_{\mathcal{O},\mathbf{N}_\omega} \models \neg A$. Either $\mathbf{K}\,H = \mathbf{K}\,A$ or $\mathbf{K}\,H$ and $\mathbf{K}\,A$ appear in a set U that is constructed as in the proof of Proposition 7, e.g., for each rule $\mathbf{K}\,H \leftarrow \mathsf{body}$ in $\mathcal{K}_G//\mathbf{N}_\omega$, we have $\mathbf{K}\,A$ in body. In both cases, $\mathbf{K}\,A$ is true in all three-valued MKNF models of $\mathcal{K}_G$ but the addition of all modal $\mathbf{K}$-atoms that are not false in all three-valued MKNF models of $\mathcal{K}_G$ (including $\mathbf{K}\,A$) to $\mathcal{O}$ derives $\neg A$. We conclude that $\mathcal{K}_G$ is MKNF-inconsistent.

The third case is a direct consequence of the way evaluation of MKNF formulas is defined: if $\mathcal{O}$ is inconsistent, then there is no first-order model of $\mathcal{O}$. Assume that (M, N) is a three-valued MKNF model of $\mathcal{K}_G$. Then, (M, N) satisfies $\mathcal{K}_G$ and thus also $\mathcal{O}$, i.e., for each $I \in M$, we have $(I, \langle M, N \rangle, \langle M, N \rangle)(\pi(\mathcal{O})) = \mathbf{t}$. Since M must not be empty, we derive a contradiction.

For the other direction, we have to show that any possibly occurring MKNF-inconsistency is detected. So, suppose that $\mathcal{K}_G$ is MKNF-inconsistent. If $\mathcal{O}$ is inconsistent, then we are done immediately. Otherwise, the rules in $\mathcal{P}_G$ alone cannot be MKNF-inconsistent, since they only consist of modal atoms without any appearance of classical negation. Likewise, rules without DL-atoms or rules without DL-atoms in at least some head cannot be inconsistent since the derivation from the ontology $\mathcal{O}$ never conflicts with any rule. Consider thus such an arbitrary DL-atom $\mathbf{K}\,H$ with a rule $\mathbf{K}\,H \leftarrow \mathbf{K}\,A_1, \ldots, \mathbf{K}\,A_n, \mathbf{not}\,B_1, \ldots, \mathbf{not}\,B_m$ in $\mathcal{P}_G$. If H is true as a consequence of $\mathcal{O}$, then the operator $D_{\mathcal{K}_G}$ ensures that $\mathbf{K}\,H$ is true as well, and no inconsistency occurs.

So, let H be first-order false and $\mathbf{K}\,H \in \mathbf{P}_\omega$, i.e. $\mathbf{K}\,H$ is true in all three-valued MKNF models of $\mathcal{K}_G$. But then $\Gamma'_{\mathcal{K}_G}(\mathbf{N}_\omega) \subset \Gamma_{\mathcal{K}_G}(\mathbf{N}_\omega)$ and the inconsistency is detected. Alternatively, $\mathbf{K}\,H$ could be undefined but then $\mathbf{K}\,H \in \mathbf{N}_\omega$, and this is not possible since H is first-order false and $\Gamma'_{\mathcal{K}_G}$ suppresses $\mathbf{K}\,H$. So the only case missing is the one where $\mathbf{K}\,H$ is false in all three-valued MKNF models (as enforced by the operator $\Gamma'_{\mathcal{K}_G}$) but the body of at least one rule with head $\mathbf{K}\,H$ is undefined. Thus $\Gamma'_{\mathcal{K}_G}(\mathbf{P}_\omega) \subset \Gamma_{\mathcal{K}_G}(\mathbf{P}_\omega)$. $\qquad\square$

We apply this check for consistency to our previous examples:

Example 12. Consider again $\mathcal{K}_G$ from Example 11. We have $\mathbf{P}_\omega = \mathbf{P}_3$ and $\mathbf{N}_\omega = \mathbf{N}_3$. We check for inconsistency (assuming $\mathcal{O}$ is consistent) and obtain $\Gamma'_{\mathcal{K}_G}(\mathbf{P}_\omega) = \Gamma_{\mathcal{K}_G}(\mathbf{P}_\omega)$ and $\Gamma'_{\mathcal{K}_G}(\mathbf{N}_\omega) \subset \Gamma_{\mathcal{K}_G}(\mathbf{N}_\omega)$. So we (rightly) conclude that $\mathcal{K}_G$ is inconsistent.

Now reconsider $\mathcal{K}_G$ from Example 10. We have $\mathbf{P}_\omega = \mathbf{P}_2$ and $\mathbf{N}_\omega = \mathbf{N}_1$. We check for consistency and obtain $\Gamma'_{\mathcal{K}_G}(\mathbf{P}_\omega) = \Gamma_{\mathcal{K}_G}(\mathbf{P}_\omega)$ and $\Gamma'_{\mathcal{K}_G}(\mathbf{N}_\omega) = \Gamma_{\mathcal{K}_G}(\mathbf{N}_\omega)$. Hence, the knowledge base is consistent, and we obtain the well-founded partition

$$(T_W, F_W) = (\{\mathbf{K}\,\mathrm{CD}(\mathrm{Tts}), \mathbf{K}\,\mathrm{int}(\mathrm{Tts}), \mathbf{K}\,\mathrm{Rec}(\mathrm{Tts})\},$$
$$\{\mathbf{K}\,\mathrm{owns}(\mathrm{Tts}), \mathbf{K}\,\mathrm{LowEv}(\mathrm{Tts})\}).$$

$\square$

The following example shows that we in fact need both the calculations w.r.t. the two fixpoints.

Example 13. Consider the following MKNF-inconsistent knowledge base that is.

$$\begin{aligned} R &\sqsubseteq \neg P \\ R(a) & \\ \mathbf{K}\,P(a) &\leftarrow \mathbf{not}\,P(a) \end{aligned}$$

$P(a)$ must be false from the ontology alone, and so $\mathbf{not}\,P(a)$ must hold, which immediately causes an inconsistency.

Only the test using $\mathbf{N}_\omega$ is able to detect this inconsistency. For $\mathsf{KA}(\mathcal{K}_G) = \{\mathbf{K}\,P(a)\}$, we obtain $\mathbf{P}_\omega = \{\mathbf{K}\,P(a)\}$ and $\mathbf{N}_\omega = \emptyset$ and thus $\Gamma'_{\mathcal{K}_G}(\mathbf{P}_\omega) = \Gamma_{\mathcal{K}_G}(\mathbf{P}_\omega) = \emptyset$ and $\Gamma'_{\mathcal{K}_G}(\mathbf{N}_\omega) = \emptyset \subset \Gamma_{\mathcal{K}_G}(\mathbf{N}_\omega) = \{\mathbf{K}\,P(a)\}$.

On the other hand, the following knowledge base is also MKNF-inconsistent, but only the test with $\mathbf{P}_\omega$ allows us to discover this.

$$\begin{aligned} R &\sqsubseteq \neg P \\ R(a) & \\ \mathbf{K}\,P(a) &\leftarrow \mathbf{not}\,u \\ \mathbf{K}\,u &\leftarrow \mathbf{not}\,u \end{aligned}$$

For $\mathsf{KA}(\mathcal{K}_G) = \{\mathbf{K}\,P(a), \mathbf{K}\,u\}$ we obtain $\mathbf{P}_\omega = \emptyset$ and $\mathbf{N}_\omega = \{\mathbf{K}\,u\}$ and thus $\Gamma'_{\mathcal{K}_G}(\mathbf{P}_\omega) = \mathbf{N}_\omega \subset \Gamma_{\mathcal{K}_G}(\mathbf{P}_\omega) = \mathsf{KA}(\mathcal{K}_G)$ and $\Gamma'_{\mathcal{K}_G}(\mathbf{N}_\omega) = \Gamma_{\mathcal{K}_G}(\mathbf{N}_\omega) = \mathbf{P}_\omega$.

The difference between the two examples is that in the first example there is a rule with true body and false head, while in the second example there is a rule with undefined body and false head. Each of the two conditions captures one of the cases, which explains why two conditions need to be checked. $\square$

As already said, normal rules alone cannot be inconsistent, unless integrity constraints (i.e. rules whose head is $\mathbf{K}\,\mathbf{f}$, cf. [52]) are allowed. In this simpler case, inconsistencies are easily detected since $\mathbf{K}\,\mathbf{f}$ must occur in $\mathsf{KA}(\mathcal{K}_G) \setminus \mathbf{N}_\omega$.

If $\mathcal{K}_G$ is MKNF-consistent, then the well-founded partition is in fact a partial partition.

Proposition 8. *Let $\mathcal{K}_G$ be an MKNF-consistent ground hybrid MKNF knowledge base and $(T_W, F_W) = (\mathbf{P}_{\mathcal{K}_G}, \mathsf{KA}(\mathcal{K}_G) \setminus \mathbf{N}_{\mathcal{K}_G})$ the well-founded partition. Then (T_W, F_W) is a partial partition.*

PROOF. From Theorem 2 and since $\mathcal{K}_G$ is MKNF-consistent, we obtain that $\Gamma'_{\mathcal{K}_G}(\mathbf{P}_\omega) = \Gamma_{\mathcal{K}_G}(\mathbf{P}_\omega)$ and $\Gamma'_{\mathcal{K}_G}(\mathbf{N}_\omega) = \Gamma_{\mathcal{K}_G}(\mathbf{N}_\omega)$. Those two equalities also yield that $T_W \cap F_W = \emptyset$, which shows that (T_W, F_W) is a partition (since T_W and F_W are subsets of $\mathsf{KA}(\mathcal{K}_G)$). $\qquad\square$

It can also be shown that the well-founded partition yields an MKNF interpretation pair that satisfies $\mathcal{K}_G$.

Theorem 3. *Let $\mathcal{K}_G$ be an MKNF-consistent ground hybrid MKNF KB and $(T_W, F_W) = (\mathbf{P}_{\mathcal{K}_G}, \mathsf{KA}(\mathcal{K}_G) \setminus \mathbf{N}_{\mathcal{K}_G})$ the well-founded partition of $\mathcal{K}_G$. Then $(I_P, I_N) \models \mathcal{K}_G$ where $I_P = \{I \mid I \models \mathsf{OB}_{\mathcal{O}, \mathbf{P}_{\mathcal{K}_G}}\}$ and $I_N = \{I \mid I \models \mathsf{OB}_{\mathcal{O}, \mathbf{N}_{\mathcal{K}_G}}\}$.*

PROOF. First of all, (I_P, I_N) is a proper MKNF interpretation pair, i.e. since any $I \in I_N$ also satisfies $\mathsf{OB}_{\mathcal{O}, \mathbf{P}_{\mathcal{K}_G}}$ we obtain $I_N \subseteq I_P$. By Definition 3, we know that $\mathcal{K}_G = \mathbf{K}\,\pi(\mathcal{O}) \wedge \pi(\mathcal{P}_G)$. Since $\pi(\mathcal{O})$ occurs in $\mathsf{OB}_{\mathcal{O}, \mathbf{P}_{\mathcal{K}_G}}$ and all $I \in I_P$ satisfy $\mathsf{OB}_{\mathcal{O}, \mathbf{P}_{\mathcal{K}_G}}$, we have $(I, \langle I_P, I_N \rangle, \langle I_P, I_N \rangle)(\mathbf{K}\,\pi(\mathcal{O})) = \mathbf{t}$ for all $I \in I_P$. Thus, we only have to consider the evaluation of $(I, \langle I_P, I_N \rangle, \langle I_P, I_N \rangle)(\pi(\mathcal{P}_G))$.

We start by evaluating the modal atoms occurring in $\pi(\mathcal{P}_G)$. Let $\mathbf{K}\,H \in \pi(\mathcal{P}_G)$. Suppose at first that $\mathbf{K}\,H \in T_W$. As such $(I, \langle I_P, I_N \rangle, \langle I_P, I_N \rangle)(\mathbf{K}\,H) = \mathbf{t}$. Alternatively, suppose $\mathbf{K}\,H \in F_W$ and assume that $\mathsf{OB}_{\mathcal{O}, \mathbf{N}_{\mathcal{K}_G}} \models H$. In this case, $\mathbf{K}\,H \in \mathbf{N}_{\mathcal{K}_G}$ by means of $\mathcal{D}_{\mathcal{K}_G}$, and we conclude that $\mathsf{OB}_{\mathcal{O}, \mathbf{N}_{\mathcal{K}_G}} \not\models H$. Therefore, we have $(I, \langle I_P, I_N \rangle, \langle I_P, I_N \rangle)(\mathbf{K}\,H) = \mathbf{f}$. Finally, let $\mathbf{K}\,H$ occur in $\mathbf{N}_{\mathcal{K}_G}$ but not in T_W or in F_W. We know that $\mathsf{OB}_{\mathcal{O}, \mathbf{N}_{\mathcal{K}_G}} \models H$. Assume $\mathsf{OB}_{\mathcal{O}, \mathbf{P}_{\mathcal{K}_G}} \models H$. In this case, $\mathbf{K}\,H \in \mathbf{P}_{\mathcal{K}_G}$ by means of $\mathcal{D}_{\mathcal{K}_G}$, and we conclude that $\mathsf{OB}_{\mathcal{O}, \mathbf{P}_{\mathcal{K}_G}} \not\models H$. Therefore, we have $(I, \langle I_P, I_N \rangle, \langle I_P, I_N \rangle)(\mathbf{K}\,H) = \mathbf{u}$.

The cases for $\mathbf{not}\,H \in \pi(\mathcal{P})$ proceed analogously. Indeed, if $\mathbf{K}\,H \in T_W$, then we have $(I, \langle I_P, I_N \rangle, \langle I_P, I_N \rangle)(\mathbf{not}\,H) = \mathbf{f}$; if $\mathbf{K}\,H \in T_W$, then we have $(I, \langle I_P, I_N \rangle, \langle I_P, I_N \rangle)(\mathbf{not}\,H) = \mathbf{t}$; and otherwise $(I, \langle I_P, I_N \rangle, \langle I_P, I_N \rangle)(\mathbf{not}\,H) = \mathbf{u}$.

Now consider $\pi(\mathcal{P}_G)$ which consists of a set of implications, each corresponding to one rule in $\mathcal{P}_G$. To show $(I, \langle I_P, I_N \rangle, \langle I_P, I_N \rangle)(\pi(\mathcal{P}_G)) = \mathbf{t}$, we only have to guarantee that the three cases that map an implication $\supset$ to false do not occur, i.e. the cases where the body of the original rule is true but the head is not (respectively, the body is undefined and the head is false). Assume that any of the three cases holds. If the body of such a rule is true, then by the alternating fixpoint construction we have that the head is true as well, contradicting these two cases. If the rule body is undefined, then (by $\mathbf{N}_{\mathcal{K}_G}$ and the alternating fixpoint) we obtain that the head has to be undefined or true, again contradicting our assumption. Thus, $(I, \langle I_P, I_N \rangle, \langle I_P, I_N \rangle)(\pi(\mathcal{P}_G)) = \mathbf{t}$ holds. $\square$

This result can be combined with Proposition 3 to show that the well-founded partition results in a three-valued MKNF model.

Theorem 4. *Let $\mathcal{K}_G$ be an MKNF-consistent ground hybrid MKNF KB and $(T_W, F_W) = (\mathbf{P}_{\mathcal{K}_G}, \mathsf{KA}(\mathcal{K}_G) \setminus \mathbf{N}_{\mathcal{K}_G})$ the well-founded partition of $\mathcal{K}_G$. Then (I_P, I_N) is a three-valued MKNF model of $\mathcal{K}_G$, where $I_P = \{I \mid I \models \mathsf{OB}_{\mathcal{O}, \mathbf{P}_{\mathcal{K}_G}}\}$ and $I_N = \{I \mid I \models \mathsf{OB}_{\mathcal{O}, \mathbf{N}_{\mathcal{K}_G}}\}$.*

PROOF. We know from Theorem 3 that (I_P, I_N) satisfies $\mathcal{K}_G$. By Proposition 3, this MKNF interpretation pair exactly corresponds to the one that equals to a three-valued MKNF model inducing that partition. Thus (I_P, I_N) is a three-valued MKNF model of $\mathcal{K}_G$. $\square$

In fact, (I_P, I_N) is the unique well-founded MKNF model, i.e. the least partial MKNF model w.r.t. derivable knowledge.

Theorem 5. *Let $\mathcal{K}_G$ be an MKNF-consistent ground hybrid MKNF KB and (I_P, I_N) the three-valued MKNF model of $\mathcal{K}_G$ induced by the well-founded partition (T_W, F_W). For any three-valued MKNF model (M, N) of $\mathcal{K}_G$ we have $(M, N) \succeq_k (I_P, I_N)$. Indeed, (I_P, I_N) is the well-founded MKNF model of $\mathcal{K}_G$.*

PROOF. We have shown in Proposition 3 that any three-valued MKNF model (M, N) of $\mathcal{K}_G$ induces a partition (T, F) which in turn gives rise to the same three-valued MKNF model (via the objective knowledge). By Proposition 7, $\mathbf{K}\, H \in T_W$ implies that $\mathbf{K}\, H$ is true (and $\mathbf{not}\, H$ is false) in all three-valued MKNF models (M, N) of $\mathcal{K}_G$, and $\mathbf{K}\, H \in F_W$ implies that $\mathbf{K}\, H$ is false (and $\mathbf{not}\, H$ is true) in all three-valued MKNF models (M, N) of $\mathcal{K}_G$. We conclude that $T_W \subseteq T$ and $F_W \subseteq F$. Furthermore, we know that $I_P = \{I \mid I \models \mathsf{OB}_{\mathcal{O}, T_W}\}$ and $I_N = \{I \mid I \models \mathsf{OB}_{\mathcal{O}, \mathsf{KA}(\mathcal{K}_G) \setminus F_W}\}$, and also that $M = \{I \mid I \models \mathsf{OB}_{\mathcal{O}, T}\}$ and $N = \{I \mid I \models \mathsf{OB}_{\mathcal{O}, \mathsf{KA}(\mathcal{K}_G) \setminus F}\}$. It is straightforward to see that $M \subseteq I_P$ and $I_N \subseteq N$, which by Definition 11 finishes the proof. $\square$

This central theorem not only shows that the well-founded MKNF model is unique and well-defined, since the well-founded MKNF model is exactly the three-valued MKNF model that is least w.r.t. $\succeq_k$, but also that the well-founded MKNF model is a sound approximation of any total three-valued MKNF model and therefore of any two-valued MKNF model. Thus, the well-founded partition can also be used in the algorithms presented in [52] for computing a subset of the knowledge that holds in all partitions corresponding to a two-valued MKNF model.

The well-founded partition of knowledge bases consisting of only rules, coincides with the well-founded model of the corresponding (normal) logic program.

Theorem 6. *Let $\mathcal{K}_G$ be a ground program of MKNF rules, Π a normal logic program obtained from $\mathcal{P}_G$ by transforming each MKNF rule*

$$\mathbf{K}\, H \leftarrow \mathbf{K}\, A_1, \ldots, \mathbf{K}\, A_n, \mathbf{not}\, B_1, \ldots, \mathbf{not}\, B_m$$

into a rule

$$H \leftarrow A_1, \ldots, A_n, \mathbf{not}\, B_1, \ldots, \mathbf{not}\, B_m$$

of Π, (T_W, F_W) the well-founded partition of $\mathcal{K}_G$, and W_Π the well-founded model of Π. Then $\mathbf{K} H \in T_W$ if and only if $H \in W_\Pi$, and $\mathbf{K} H \in F_W$ if and only if $\mathbf{not}\ H \in W_\Pi$.

Finally, the following theorem is obtained from the data complexity results for positive nondisjunctive MKNF knowledge bases in [52], where data complexity is measured in terms of A-Box assertions and rule facts.

Theorem 7. *Let $\mathcal{K}$ be a hybrid MKNF KB. Assuming that entailment of ground DL-atoms in $\mathcal{DL}$ is decidable with data complexity $\mathcal{C}$, the data complexity of computing the well-founded partition is in* $\mathrm{P}^\mathcal{C}$.

For comparison, the data complexity for reasoning with two-valued MKNF models in nondisjunctive programs is shown to be $\mathcal{E}^{\mathrm{P}^\mathcal{C}}$ where $\mathcal{E} = \mathrm{NP}$ if $\mathcal{C} \subseteq \mathrm{NP}$, and $\mathcal{E} = \mathcal{C}$ otherwise. Thus, computing the well-founded partition ends up in a strictly smaller complexity class than deriving the two-valued MKNF models. In fact, if the description logic fragment is tractable,[17] then we obtain a formalism whose model is computed with a data complexity in P. This is remarkable, since to the best of our knowledge this is the first time that a general tractable local closed world extension for DLs has been identified.

5. Related Work

Several proposals exist for combining rules and ontologies (see, e.g., [13, 28] for a brief survey). They can be split into two groups, namely those semantically based on first-order logics solely (such as description logics alone), and the hybrid approaches (such as hybrid MKNF) providing a semantics combining elements of first-order logics with nonmonotonicity.

The most general approach in the first group is SWRL [32], an unrestricted combination of OWL-DL with function-free Horn rules (i.e., rules without negation). The approach is very expressive but undecidable, yet nevertheless generalises many approaches in this group. Applying, e.g., DL-safety to SWRL rules yields DL-safe rules [53], a decidable subset of SWRL. $\mathcal{AL}$-log [10], a combination of DL-safe positive rules and $\mathcal{ALC}$, and CARIN [45] are also notable formalisms generalized by SWRL. In both cases, the ontology only serves as input to the rules and not vice versa, i.e., information flow is one way. Description Logic Programs (DLP) [24, 44] are a fragment of OWL that can be transformed into logic programs of positive rules. In the same spirit, Horn-$\mathcal{SHIQ}$ [33] is a fragment of OWL that can be translated into Datalog, and (like DLP) is of tractable data complexity.[18] Recently, DLP has been generalised to Description Logic Rules [39, 41], i.e. rules that may contain description logic expressions. This enriches the DL, on which the description logic rules are based, with sophisticated constructs normally only available to more expressive description

---

[17]See, e.g., the OWL2 profiles at http://www.w3.org/TR/owl2-profiles/.
[18]Further analyses of Horn description logics are provided in [43].

logics, without increasing the complexity. Similarly, ELP [39, 42, 48] is a polynomial language covering important parts of OWL 2. ELP also allows some axioms that cannot be expressed in OWL 2. All of these approaches have the advantage of fitting semantically into the original (first-order) OWL semantics, which also means that existing reasoners for ontologies alone can be used for reasoning in the combined knowledge bases. On the other hand, none of these approaches can express nonmonotonic negation, and as such none cover the motivating cases discussed in the introduction.

In the second group, which includes hybrid MKNF, the approach in [14] combines ontologies and rules in a modular way, i.e. both parts and their semantics are kept separate. The two reasoning engines nevertheless interact bidirectionally (with some limitations in the direction of the ontology to rules) via interfaces, and the dlvhex system [15] provides an implementation that generalises the approach by allowing multiple sources for external knowledge (with differing semantics). This work has been extended in various ways (e.g., probabilities, uncertainty, and priorities; for references see the related work section of [14]). A related well-founded semantics [16] has been proposed with a computational complexity quite similar to the one of our well-founded MKNF semantics. However, the integration is less tight, and the formalism restricts the transfer of information from ontologies to rules. The only other well-founded semantics approach is called hybrid programs [12] but this approach only allows the transfer of information from the ontology to the rules. Thus, [12] is strictly less expressive than the well-founded MKNF semantics. The advantage of such a restriction is that, contrary to [16] and the well-founded MKNF semantics, the semantics remains compatible with the standard semantics: consider two DL-atoms B_1 and B_2 and an ontology that expresses that at least one of them is true but none is a logical consequence of the ontology. Then, given rules $p \leftarrow B_1$ and $p \leftarrow B_2$, p is obtained by [12] but not in our work or in the formalism described by Eiter et. al. [14].

There are several further approaches related to stable models of logic programs. [8] uses an embedding into autoepistemic logic to tightly combine ontologies and rules. The approach is quite similar in spirit to hybrid MKNF [52]. In fact, the embedding with epistemic rule bodies and epistemic rule heads seems to be the one most closely related, not only syntactically but also with respect to the semantic consequences. However, a precise relation to hybrid MKNF is far from obvious since an autoepistemic interpretation in [8] is a pair of a first-order interpretations and a set of beliefs and both are not necessarily related. DL+log [59] provides a combination of rules and ontologies that separates predicates into rule and ontology predicates and evaluates the former w.r.t. the answer set semantics and the latter w.r.t. a first-order semantics with weak DL-safety, i.e. each variable in the head of a rule appears in an arbitrary positive atom in the body of the rule. Like [52], [9] generalises [59] and several earlier related works (e.g. [58]) within the framework of equilibrium logics. Quite similar to [59] is [47], although this approach does not distinguish between ontology and rules predicates. In fact, the work originates from [14] and it is thus from the perspective of rules but permits a much tighter integration. Open

answer set programming [25] extends rules with open domains and adds some syntactic limitations for ensuring decidability. Based on that, an algorithm has been provided for f-hybrid knowledge bases [17], i.e. a combination of ontologies and rules without DL-safety but which limits predicates to tree-shapedness. A loose layering of Prolog on top of DLs, employing four-valued logic, is presented in [49].

An alternative way of introducing nonmonotonicity into ontologies is to enrich DLs with further syntactic constructs representing nonmonotonic features. Among these approaches the most closely related to our work is Description Logics of MKNF [11], which allows two modal operators in ontology axioms. An algorithm was provided in [11] for $\mathcal{ALC}$ with MKNF, and it has been improved in [34]. In [5, 21], circumscription was used for adding nonmonotonic reasoning to DLs, and several other formalisms introducing defaults to ontologies exist (e.g., [4]).

6. Conclusions and Future Work

Summarising, we have defined a well-founded semantics of (tightly integrated) hybrid KBs that is sound w.r.t. the semantics defined in [52] for MKNF KBs but has a strictly lower complexity. In particular, we obtain tractability whenever the underlying description logic is tractable. To the best of our knowledge, the well-founded MKNF semantics is the first approach for the combination of arbitrary nonmonotonic rules and ontologies without any limitations on the transfer of information between the two. Our approach coincides with the first-order semantics of the DL fragment if there are no rules, and with the well-founded semantics of normal programs if the DL-part is empty. Moreover, we defined a construction for computing the well-founded MKNF model that is also capable of detecting inconsistencies in a straightforward way.

Several lines of future research can be considered. First of all, we are working towards a general query-driven procedure capable of answering conjunctive queries under the well-founded semantics of hybrid MKNF knowledge bases. In fact, we already have some results on this issue: in [1], a procedure is defined using tabled resolution that is sound and complete w.r.t. the well-founded semantics defined here, and that is terminating for several classes of knowledge bases. This procedure, which takes an oracle capable of answering queries in the underlying description logic as a parameter, is able to answer DL-safe conjunctive queries (i.e., conjunctive predicates with variables, where queries have to be ground when processed in the ontology) returning all correct answer substitutions for variables in the query. An implementation of this procedure, that is based on XSB Prolog[19] for the tabling resolution, is already part of the CVS version of XSB Prolog, and the description of this implementation can be found at [20].

---

[19]http://xsb.sourceforge.net/

Another line of current research is the specialisation of the semantics defined here (with corresponding procedures and implementations) to particular tractable description logics, as opposed to considering DLs in general. This specific study aims at $\mathcal{EL}^{++}$ [2] and ELP [39, 42, 48], one extension of $\mathcal{EL}^{++}$. We intend to provide a transformation of such hybrid knowledge bases into rules that can be used as input to a logic programming system capable of computing the well-founded MKNF model of a set of rules. We have advanced already in this work, by providing just that for $\mathcal{EL}^+$ [35].

Another topic that we are pursuing is the definition of a paraconsistent version of the semantics defined here. It is worth noting that when inconsistencies come from the combination of rules and the DL-part (i.e. for inconsistent KBs with a consistent DL-part), the construction still yields some results, e.g., in Example 11 we still derive that Bts is interesting. This suggests that the method could be further exploited in the direction of defining a paraconsistent semantics for hybrid KBs.

References

[1] J. J. Alferes, M. Knorr, and T. Swift. Queries to hybrid MKNF knowledge bases through oracular tabling. In Abraham Bernstein, David R. Karger, Tom Heath, Lee Feigenbaum, Diana Maynard, Enrico Motta, and Krishnaprasad Thirunarayan, editors, *International Semantic Web Conference*, pages 1–16. Springer, 2009.

[2] F. Baader, S. Brandt, and C. Lutz. Pushing the $\mathcal{EL}$ envelope further. In K. Clark and P. F. Patel-Schneider, editors, *In Proceedings of the OWLED 2008 DC Workshop on OWL: Experiences and Directions*, 2008.

[3] F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, and P. F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge University Press, 2 edition, 2007.

[4] F. Baader and B. Hollunder. Embedding defaults into terminological representation systems. *Journal of Automated Reasoning*, 14:149–180, 1995.

[5] P. Bonatti, C. Lutz, and F. Wolter. Expressive non-monotonic description logics based on circumscription. In P. Doherty, J. Mylopoulos, and C. Welty, editors, *Proceedings of the Tenth International Conference on Principles of Knowledge Representation and Reasoning (KR'06)*, pages 400–410. AAAI Press, 2006.

[6] W. Chen and D. S. Warren. Tabled Evaluation with Delaying for General Logic Programs. *Journal of the ACM*, 43(1):20–74, 1996.

[7] E. Dantsin, T. Eiter, G. Gottlob, and A. Voronkov. Complexity and expressive power of logic programming. *ACM Computing Surveys*, 33(3):374–425, 2001.

[8] J. de Bruijn, T. Eiter, A. Polleres, and H. Tompits. Embedding non-ground logic programs into autoepistemic logic for knowledge-base combination. In *Proceedings of the Twentieth International Joint Conference on Artificial Intelligence (IJCAI-07)*, pages 304–309, Hyderabad, India, January 6–12 2007. AAAI Press.

[9] J. de Bruijn, D. Pearce, A. Polleres, and A. Valverde. Quantified equilibrium logic and hybrid rules. In *Proceedings of the First International Conference on Web Reasoning and Rule Systems*. Springer, 2007.

[10] F.M. Donini, M. Lenzerini, D. Nardi, and A. Schaerf. AL-log: Integrating datalog and description logics. *Journal of Intelligent Information Systems*, 10(3):227–252, 1998.

[11] F.M. Donini, D. Nardi, and R. Rosati. Description logics of minimal knowledge and negation as failure. *ACM Transactions on Computational Logic*, 3(2):177–225, 2002.

[12] W. Drabent and J. Małuszynski. Well-Founded semantics for hybrid rules. In *Proceedings of the First International Conference on Web Reasoning and Rule Systems (RR2007)*, pages 1–15. Springer, 2007.

[13] T. Eiter, G. Ianni, T. Krennwallner, and A. Polleres. Rules and ontologies for the semantic web. pages 1–53, 2008.

[14] T. Eiter, G. Ianni, T. Lukasiewicz, R. Schindlauer, and H. Tompits. Combining answer set programming with description logics for the semantic web. *Artificial Intelligence*, 172(12–13):1495–1539, August 2008.

[15] T. Eiter, G. Ianni, R. Schindlauer, and H. Tompits. Effective integration of declarative rules with external evaluations for semantic web reasoning. In Y. Sure and J. Domingue, editors, *Proceedings of the 3rd European Conference on Semantic Web (ESWC 2006)*, pages 273–287. Springer, 2006.

[16] T. Eiter, T. Lukasiewicz, R. Schindlauer, and H. Tompits. Well-Founded semantics for description logic programs in the semantic web. In G. Antoniou and H. Boley, editors, *Rules and Rule Markup Languages for the Semantic Web, RuleML'04*, pages 81–97. Springer, LNCS, 2004.

[17] C. Feier and S. Heymans. Hybrid reasoning with forest logic programs. In *6th Annual European Semantic Web Conference (ESWC2009)*, pages 338–352, June 2009.

[18] M. Fitting. *First-order logic and automated theorem proving (2nd ed.).* Springer, 1996.

[19] M. Gelfond and V. Lifschitz. The stable model semantics for logic programming. In R. A. Kowalski and K. A. Bowen, editors, *Int. Conference on Logic Programming, ICLP*, pages 1070–1080. MIT Press, 1988.

[20] A. S. Gomes, J. J. Alferes, and T. Swift. Implementing query answering for hybrid mknf knowledge bases. In *Practical Aspects of Declarative Languages, 12th International Symposium, PADL 2010, Madrid, Spain, January 18-19, 2010. Proceedings*, pages 25–39. Springer, 2010.

[21] S. Grimm and P. Hitzler. A preferential tableaux calculus for circumscriptive ALCO. In A. Polleres and T. Swift, editors, *Web Reasoning and Rule Systems, Third International Conference, RR 2009, Chantilly, VA, USA, October 20009, Proceedings*, pages 197–211. Springer, 2009.

[22] S. Grimm and P. Hitzler. Semantic matchmaking of web resources with local closed-world reasoning. *International Journal of e-Commerce*, 12(2):89–126, Winter 2007–08.

[23] S. Grimm, B. Motik, and C. Preist. Matching semantic service descriptions with local closed-world reasoning. In Y. Sure and J. Domingue, editors, *The Semantic Web: Research and Applications, 3rd European Semantic Web Conference, ESWC 2006, Budva, Montenegro, June 11–14, 2006, Proceedings*, pages 575–589. Springer, 2006.

[24] B. Grosof, I. Horrocks, R. Volz, and S. Decker. Description logic programs: Combining logic programs with description logics. In *Proceedings of the World Wide Web Conference (WWW2003), Budapest, Hungary*, pages 48–57. ACM, 2003.

[25] S. Heymans, D. Van Nieuwenborgh, and D. Vermeir. Open answer set programming for the semantic web. *Journal of Applied Logic*, 5(1):144–169, 2007.

[26] P. Hitzler, M. Krötzsch, B. Parsia, P. F. Patel-Schneider, and S. Rudolph, editors. *OWL 2 Web Ontology Language: Primer.* W3C Recommendation 27 October 2009, 2009. Available from `http://www.w3.org/TR/owl2-primer/`.

[27] P. Hitzler, M. Krötzsch, and S. Rudolph. *Foundations of Semantic Web Technologies.* Chapman & Hall/CRC, 2009.

[28] P. Hitzler and B. Parsia. Ontologies and rules. In S. Staab and R. Studer, editors, *Handbook on Ontologies*, pages 111–132. Springer, 2 edition, 2009.

[29] P. Hitzler and M. Wendt. A uniform approach to logic programming semantics. *Theory and Practice of Logic Programming*, 5(1–2):123–159, 2005.

[30] I. Horrocks, O. Kutz, and U. Sattler. The even more irresistible SROIQ. In P. Doherty, J. Mylopoulos, and C. A. Welty, editors, *In Proc. of the 10th Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR 2006)*, pages 57–67. AAAI Press, 2006.

[31] I. Horrocks, B. Motik, R. Rosati, and U. Sattler. Can OWL and logic programming live together happily ever after? In I. F. Cruz, S. Decker, D. Allemang, C. Preist, D. Schwabe, P. Mika, Michael Uschold, and Lora Aroyo, editors, *Proc. of the 5th Int. Semantic Web Conference (ISWC 2006), volume 4273 of LNCS*, pages 501–514. Springer, 2006.

[32] I. Horrocks and P. F. Patel-Schneider. A proposal for an OWL rules language. In *Proc. of the 13th Int. World Wide Web Conference (WWW 2004)*, pages 723–731. ACM, 2004.

[33] U. Hustadt, B. Motik, and U. Sattler. Data complexity of reasoning in very expressive description logics. In *Proc. of the 19th Int. Joint Conference on Artifical Intelligence (IJCAI)*, pages 466–471. ACM, 2005.

[34] P. Ke and U. Sattler. Next steps for description logics of minimal knowledge and negation as failure. In F. Baader, C. Lutz, and B. Motik, editors, *Proc. of the 2008 Description Logic Workshop (DL 2008)*, 2008.

[35] M. Knorr and J. J. Alferes. Querying in $\mathcal{EL}^+$ with nonmonotonic rules. In *Proceedings of the 19th European Conference on Artificial Intelligence, ECAI2010*, pages 1079–1080. IOS Press, 2010.

[36] M. Knorr, J. J. Alferes, and P. Hitzler. Towards tractable local closed world reasoning for the semantic web. In J. Neves, M. F. Santos, and J. Machado, editors, *Progress in Artificial Intelligence*, pages 3–14. Springer, 2007.

[37] M. Knorr, J. J. Alferes, and P. Hitzler. A coherent well-founded model for hybrid MKNF knowledge bases. In M. Ghallab, C. D. Spyropoulos, N. Fakotakis, and N. Avouris, editors, *Proceedings of the 18th European Conference on Artificial Intelligence, ECAI2008*, pages 99–103. IOS Press, 2008.

[38] M. Knorr and P. Hitzler. A comparison of disjunctive well-founded semantics. In *FAInt-07, Foundations of Artificial Intelligence, Workshop at KI 2007*. CEUR Workshop Proceedings, Vol. 277, 2007.

[39] M. Krötzsch. *Description Logic Rules*, volume 008 of *Studies on the Semantic Web*. IOS Press/AKA, 2010.

[40] M. Krötzsch. Efficient inferencing for OWL EL. In T. Janhunen and I. Niemelä, editors, *Proc. of the 12th European Conf. on Logics in Artificial Intelligence (JELIA'10)*, pages 234–246. Springer, 2010.

[41] M. Krötzsch, S. Rudolph, and P. Hitzler. Description logic rules. In Malik Ghallab, Constantine D. Spyropoulos, Nikos Fakotakis, and Nikos Avouris, editors, *Proceedings of the 18th European Conference on Artificial Intelligence, ECAI2008*, pages 80–84. IOS Press, 2008.

[42] M. Krötzsch, S. Rudolph, and P. Hitzler. ELP: Tractable rules for OWL 2. In A. Sheth, S. Staab, M. Dean, M. Paolucci, D. Maynard, T. Finin, and K. Thirunarayan, editors, *Proceedings of the 7th International Semantic Web Conference (ISWC-08)*, pages 649–664. Springer, 2008.

[43] M. Krötzsch, S. Rudolph, and Pascal Hitzler. Complexity boundaries for Horn description logics. In *Proceedings of the 22nd AAAI Conference on Artificial Intelligence (AAAI'07)*, pages 452–457. AAAI Press, 2007.

[44] M. Krötzsch, S. Rudolph, and Peter H. Schmitt. On the semantic relationship between datalog and description logics. In *Proceedings of the 4th Interational Conference on Web Reasoning and Rule Systems (RR'10)*, LNCS, pages 88–102. Springer, 2010.

[45] A. Y. Levy and M. Rousset. Combining Horn rules and description logics in CARIN. *Artificial Intelligence*, 104(1–2):165–209, 1998.

[46] V. Lifschitz. Nonmonotonic databases and epistemic queries. In *International Joint Conferences on Artifical Intelligence, IJCAI'91*, pages 381–386, 1991.

[47] T. Lukasiewicz. A novel combination of answer set programming with description logics for the semantic web. In E. Franconi, M. Kifer, and W. May, editors, *Proceedings of the 4th European Semantic Web Conference (ESWC 2007)*, pages 384–398. Springer, 2007.

[48] F. Maier, A.A. Krisnadhi, M. Krötzsch, and P. Hitzler. A better uncle for OWL: Nominal schemas for integrating rules and ontologies. In *Proceedings of the 20th International World Wide Web Conference, WWW2011, March/April 2011*. to appear.

[49] T. Matzner and P. Hitzler. Any-world access to OWL from Prolog. In J. Hertzberg, M. Beetz, and R. Englert, editors, *KI 2007: Advances in Artificial Intelligence, 30th Annual German Conference on AI, KI 2007, Osnabrck, Germany, September 2007, Proceedings*, pages 84–98. Springer, 2007.

[50] B. Motik and R. Rosati. Closing semantic web ontologies. Technical report, University of Manchester, UK, 2006.

[51] B. Motik and R. Rosati. A faithful integration of description logics with logic programming. In *Proceedings of the Twentieth International Joint Conference on Artificial Intelligence (IJCAI-07)*, pages 477–482, Hyderabad, India, January 6–12 2007. AAAI Press.

[52] B. Motik and R. Rosati. Reconciling Description Logics and Rules. *Journal of the ACM*, 57(5), 2010.

[53] B. Motik, U. Sattler, and R. Studer. Query-answering for OWL-DL with rules. *Journal of Web Semantics*, 3(1):41–60, 2005.

[54] C. Patel, J. J. Cimino, J. Dolby, A. Fokoue, A. Kalyanpur, A. Kershenbaum, L. Ma, E. Schonberg, and K. Srinivas. Matching patient records to clinical trials using ontologies. In *ISWC/ASWC*, pages 816–829. Springer, 2007.

[55] L. M. Pereira and J. J. Alferes. Well founded semantics for logic programs with explicit negation. In *European Conference on Artifical Intelligence, ECAI*, pages 102–106, 1992.

[56] T. Przymusinski. Stable semantics for disjunctive programs. *New Generation Computing*, 9:401–424, 1991.

[57] R. Reiter. What should a database know? *Journal of Logic Programming*, 14(1–2):127–153, 1992.

[58] R. Rosati. On the decidability and complexity of integrating ontologies and rules. *Journal of Web Semantics*, 3(1):41–60, 2005.

[59] R. Rosati. DL+Log: A tight integration of description logics and disjunctive datalog. In P. Doherty, J. Mylopoulos, and C. Welty, editors, *Tenth International Conference on the Principles of Knowledge Representation and Reasoning, KR'06*, pages 68–78. AAAI Press, 2006.

[60] A. Tarski. Lattice-theoretic fixpoint theorem and its applications. *Pacific Journal of Mathematics*, pages 285–309, 1955.

[61] A. van Gelder. The alternating fixpoint of logic programs with negation. In *Principles of Database Systems*, pages 1–10. ACM Press, 1989.

[62] A. van Gelder, K. A. Ross, and J. S. Schlipf. The well-founded semantics for general logic programs. *Journal of the ACM*, 38(3):620–650, 1991.

[63] M. Y. Vardi. Why is modal logic so robustly decidable? In Neil Immerman and Phokion G. Kolaitis, editors, *Descriptive Complexity and Finite Models, Proceedings of a DIMACS Workshop, January 14-17, 1996, Princeton University*, pages 149–184. American Mathematical Society, 1996.