On the decidability and complexity of integrating ontologies and rules

Riccardo Rosati
Dipartimento di Informatica e Sistemistica
Università di Roma "La Sapienza"
Via Salaria 113, 00198 Roma, Italy
rosati@dis.uniroma1.it

Abstract

We define the formal framework of r-hybrid knowledge bases (KBs) integrating ontologies and rules. A r-hybrid KB has a structural component (ontology) and a rule component. Such a framework is very general, in the sense that: (i) the construction is parametric with respect to the logic used to specify the structural component; (ii) the rule component is very expressive, since it consists of a Datalog$^{\neg\vee}$ program, i.e., a Datalog program with negation as failure and disjunction. (iii) the rule component is constrained in its interaction with the structural component according to a safeness condition: such a safe interaction between rules and structural KB captures (and is a generalization of) several previous proposals. As a consequence, we are able to show that such a framework of r-hybrid KBs comprises many systems proposed for combining rules and Description Logics. Then, we study reasoning in r-hybrid KBs. We provide a general algorithm for reasoning in r-hybrid KBs, and prove that, under very general conditions, decidability of reasoning is preserved when we add safe Datalog$^{\neg\vee}$ rules to a KB: in other words, if reasoning in the logic $\mathcal{L}$ used to specify the structural component $\mathcal{T}$ is decidable, then reasoning in the extension of $\mathcal{T}$ with safe Datalog$^{\neg\vee}$ rules is still decidable. We also show that an analogous property holds for the complexity of reasoning in r-hybrid KBs. Our decidability and complexity results generalize in a broad sense previous results obtained in recent research on this topic. In particular, we prove that reasoning in r-hybrid KBs whose structural component is specified in the Web Ontology Language OWL-DL is decidable.

1 Introduction

Research in knowledge-based systems has in the last years dealt with the problem of overcoming the limitations imposed by a single knowledge representation language. *Hybrid systems* [15] have thus been proposed, which are constituted of two or more subsystems, each of which deals with a distinct portion of the knowledge base and uses specific representation formalisms and reasoning procedures. The improvement in the deductive power of hybrid systems is in terms of both the inferences the system is able to make, and the efficiency of the reasoning process, since any subsystem can take advantage of the inferential power of the other subsystems, whereas the use of specialized reasoning procedures allows for improving the efficiency of the deduction process.

The idea of adding rules to structured knowledge representation systems follows this line of research, and dates back to early Description Logic systems like CLASSIC [33], LOOM [29], and CLASP [36]. Then, the idea of building hybrid systems combining rules and structured representation of information has been pursued in a formally more rigorous and coherent way. The first approaches in this direction [27, 6, 9] studied the problem of integrating Datalog rules with Description Logics (DLs). Informally, the basic idea underlying hybrid formalisms integrating rules and descriptions is to deal with knowledge bases (KBs) constituted by a rule component (a Datalog program) and a structural component (a DL knowledge base). The interaction between the two subsystems is obtained by allowing some variables in Datalog rules to range over the set of instances of a specified concept of the DL knowledge base.

Such hybrid formalisms proved well-suited for the construction of tools for accessing heterogeneous information systems. In particular, one of such formalisms (CARIN) has been used in the Information Manifold, a system developed at AT&T for integrated access to different structured information sources on the World Wide Web [24].

More recently, a renewed interest towards the integration of structured KBs and rules has emerged in the research on ontologies and the Semantic Web [21, 1]. DLs are also playing a central role in this field, since they are currently the most used formalisms for building ontologies, and have been proposed as standard languages for the specification of ontologies in the Semantic Web [32].

However, as shown by the first studies in this field [27], decidability (and complexity) of reasoning is a crucial issue in systems combining DL KBs and rules. In fact, the interaction does not preserve decidability, i.e., starting from a KB in which reasoning is decidable and a rule KB in which reasoning is decidable, reasoning in the KB obtained by integrating the two components may not be a decidable problem.

In this paper, we study reasoning in description logic knowledge bases augmented with rules expressed in Datalog (and its nonmonotonic extensions).

We start by defining r-hybrid knowledge bases (KBs). A r-hybrid KB has a structural component $\mathcal{T}$, that is a theory in a subset of first-order logic (for instance, a DL knowledge base), and a rule component $\mathcal{P}$. Such a framework is very general, in the sense that:

1. the construction is parametric with respect to the structural language, i.e., the logic used to specify the structural component. The only condition imposed by the framework is that the logic is a subset of function-free first-order logic. In particular, any description logic can be chosen as the structural language;

2. the rule component is very expressive, since it consists of a Datalog$^{\neg\vee}$ program, i.e., a Datalog program in which negation as failure in the body of rules and disjunction in the head of rules are allowed.

3. the rule component is constrained in its interaction with the structural component according to a *safeness condition*: such a safe interaction between rules and structural KB captures (and is a generalization of) several previous proposals [9, 34, 31].

As a consequence, we are able to show that the framework of r-hybrid KBs comprises many systems proposed for integrating rules and Description Logics.

Then, we study reasoning in r-hybrid KBs. First, we provide a general algorithm for reasoning in r-hybrid KBs, and prove that, very often, decidability of reasoning is preserved when we add safe Datalog$^{\neg\vee}$ rules to a KB: in other words, under very general conditions, if reasoning in the logic $\mathcal{L}$ used to specify the structural component $\mathcal{T}$ is decidable, then reasoning in the extension of $\mathcal{T}$ with safe Datalog$^{\neg\vee}$ rules is still decidable. We also show that an analogous property holds for the complexity of reasoning in r-hybrid KBs.

Our decidability and complexity results generalize in a broad sense previous results shown in [9, 34, 31]. In particular, [31] established decidability of reasoning in the description logic $\mathcal{SHOIN}$

enhanced with safe, positive Datalog rules. Our results imply that we can extend such a framework to nonmonotonic Datalog$^{\neg\vee}$ rules and to more expressive, decidable DLs, and preserve decidability of reasoning. Notably, one such DL, $\mathcal{SHOIN}(\mathbf{D})$, is equivalent to the Web Ontology language OWL-DL, which is currently playing a crucial role in the Semantic Web initiative [32], since it is a W3C recommendation language for ontology representation in the Semantic Web: therefore, our results immediately imply that extending OWL-DL ontology specifications with safe Datalog$^{\neg\vee}$ rules preserves decidability of reasoning.

Finally, our algorithm highlights that reasoning in r-hybrid KBs can be done by strongly separating reasoning about the structural component and reasoning about the rule component. This is a very important property, which allows for reusing deductive techniques (and implemented systems) developed for the structural language and for Datalog$^{\neg\vee}$ [12].

The paper is structured as follows. In Section 2 we define r-hybrid KBs. In Section 3 we study reasoning in r-hybrid KBs: we first define an algorithm for satisfiability of r-hybrid KBs, then address decidability and complexity of reasoning with r-hybrid KBs. We discuss related work in Section 4. Finally, we draw some conclusions in Section 5.

2 Framework

In this section we define syntax and semantics of r-hybrid KBs. We introduce both a monotonic, first-order semantics and a nonmonotonic semantics based on stable models.

2.1 R-Hybrid KBs: Syntax

We denote by $\mathcal{L}$ any subset of the function-free first-order logic language (for example, a description logic language) over an alphabet of predicates $\mathcal{A} = \mathcal{A}_P \cup \mathcal{A}_R$, with $\mathcal{A}_P \cap \mathcal{A}_R = \emptyset$, and an alphabet of constants $\mathcal{C}$. Every $p \in \mathcal{A}_P$ is called a *structural predicate.* An *atom* is an expression of the form $r(X)$, where r is a predicate in $\mathcal{A}$ of arity n and X is a n-tuple of variables and constants. If no variable symbol occurs in X, then $r(X)$ is called a *ground atom.*

Definition 1 *A* r-hybrid knowledge base *(KB) $\mathcal{H}$ is a pair $(\mathcal{T}, \mathcal{P})$, where:*

- *$\mathcal{T} \subseteq \mathcal{L}$ and no predicate in $\mathcal{A}_R$ occurs in $\mathcal{T}$. $\mathcal{L}$ is called the* structural language of *$\mathcal{H}$;*
- *$\mathcal{P}$ is a* Datalog$^{\neg\vee}$ program *over the predicate alphabet $\mathcal{A}$ and the alphabet of constants $\mathcal{C}$, i.e., a set of Datalog$^{\neg\vee}$ rules where each rule R has the form*

  $$p_1(X_1) \vee \ldots \vee p_n(X_n) \leftarrow r_1(Y_1), \ldots, r_m(Y_m), s_1(Z_1), \ldots, s_k(Z_k), not\, u_1(W_1), \ldots, not\, u_h(W_h)$$

  such that $n \geq 0$, $m \geq 0$, $k \geq 0$, $h \geq 0$, each $p_i(X_i)$, $r_i(Y_i)$, $s_i(Z_i)$, $u_i(W_i)$ is an atom and:

  - *each p_i is a predicate from $\mathcal{A}$;*
  - *each r_i, u_i is a predicate from $\mathcal{A}_R$;*
  - *each s_i is a predicate from $\mathcal{A}_P$;*
  - (safeness condition) *each variable occurring in R must occur in one of the r_i's.*

  If $n = 0$, we call R a constraint. *If, for all $R \in \mathcal{P}$, $n \leq 1$, $\mathcal{P}$ is called a* Datalog$^{\neg}$ program. *If, for all $R \in \mathcal{P}$, $n \leq 1$ and $h = 0$, $\mathcal{P}$ is called a* positive Datalog program. *If there are no occurrences of variable symbols in $\mathcal{P}$, $\mathcal{P}$ is called a* ground program.

Informally, $\mathcal{P}$ is a Datalog$^{\neg\vee}$ program with a special safeness condition: in each rule R, each variable occurring in R must occur in a positive atom in the body of R whose predicate is from $\mathcal{A}_R$, i.e., does not occur in $\mathcal{T}$. Notice that such a condition strengthens the standard Datalog range restriction condition on the use of variables in rules.

Thus, the structural component and the rule component share the predicates in $\mathcal{A}_P$ and the constants in $\mathcal{C}$, while the alphabet of predicates $\mathcal{A}_R$ is only used by the rule component $\mathcal{P}$.

2.2 First-order semantics for r-hybrid KBs

We now define two semantics for r-hybrid KBs: the first one relies on a first-order logic interpretation of both the structural and the rule component of the r-hybrid KB, while the second semantics provides a nonmonotonic meaning to rules.

From now on, unless specified otherwise, with the term *interpretation* we indicate a first-order interpretation of the predicates in $\mathcal{A}$ and the constants in $\mathcal{C}$.

In order to be able to provide a nonmonotonic semantics for r-hybrid KBs, we impose the following semantic condition, known as the *standard names assumption*, which is commonly adopted when defining nonmonotonic extensions of first-order formalisms (see e.g. [23, 28, 8, 10]): every interpretation is over the same fixed, countably infinite, domain Δ, and in addition, the alphabet of constants $\mathcal{C}$ is such that it is in the same one-to-one correspondence with Δ in every interpretation: that is, there is a constant symbol for each element of Δ, each constant denotes the same element of Δ in every interpretation, and two distinct constants denote two distinct elements (this last property is known as the *unique name assumption*).

Under this assumption, we can, with a little abuse of notation, use the same symbol to denote both a constant and its semantic interpretation. Moreover, since the meaning of constants is fixed and is the same in evey interpretation, from now on we assume that first-order interpretations do not explicitly deal with constants, i.e., they only define the interpretation of predicates.

Under the standard names assumption, the notion of satisfaction of a first-order sentence in a first-order interpretation is defined as follows:

- $\mathcal{I}$ satisfies a ground atom $p(t)$ iff $t \in p^{\mathcal{I}}$;
- $\mathcal{I}$ satisfies the equality $c_1 = c_2$ iff c_1 and c_2 are the same constant symbol;
- $\mathcal{I}$ satisfies $\psi_1 \wedge \psi_2$ iff $\mathcal{I}$ satisfies ψ_1 and $\mathcal{I}$ satisfies ψ_2;
- $\mathcal{I}$ satisfies $\neg\psi$ iff $\mathcal{I}$ does not satisfy ψ;
- $\mathcal{I}$ satisfies $\exists x.\psi$ iff there exists $c \in \mathcal{C}$ such that $\mathcal{I}$ satisfy $\psi(c/x)$, where $\psi(c/x)$ is the formula obtained from ψ by replacing each occurrence of the variable x with the constant c.

The first-order semantics of a r-hybrid KB consists of a first-order interpretation not only of the structural component, but also of the rule component of the r-hybrid KB. Formally, let R be the following Datalog$^{\neg\vee}$ rule:

$$R = p_1(X_1, c_1) \vee \ldots \vee p_n(X_n, c_n) \leftarrow \begin{array}{l} r_1(Y_1, d_1), \ldots, r_m(Y_m, d_m), s_1(Z_1, e_1), \ldots, s_k(Z_k, e_k), \\ \quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad\quad \\ \textit{not}\ u_1(W_1, f_1), \ldots, \textit{not}\ u_h(W_h, f_h) \end{array} \quad (1)$$

where each X_i, Y_i, Z_i, W_i is a set of variables and each c_i, d_i, e_i, f_i is a set of constants. Then, $FO(R)$ is the first-order sentence

$\forall \overline{x}_1, \ldots, \overline{x}_n, \overline{y}_1, \ldots, \overline{y}_m, \overline{z}_1, \ldots, \overline{z}_k, \overline{w}_1, \ldots, \overline{w}_h.$
$r_1(\overline{y}_1, d_1) \wedge \ldots \wedge r_m(\overline{y}_m, d_m) \wedge s_1(\overline{z}_1, e_1) \wedge \ldots \wedge s_k(\overline{z}_k, e_k) \wedge \neg u_1(\overline{w}_1, f_1) \wedge \ldots \wedge \neg u_h(\overline{w}_h, f_h) \rightarrow$
$p_1(\overline{x}_1, c_1) \vee \ldots \vee p_n(\overline{x}_n, c_n)$

Given a Datalog$^{\neg\vee}$ program $\mathcal{P}$, $FO(\mathcal{P})$ is the set of first-order sentences $\{FO(R) \mid R \in \mathcal{P}\}$.

A *FOL-model* of a r-hybrid KB $\mathcal{H}$ is an interpretation $\mathcal{I}$ such that $\mathcal{I}$ satisfies $\mathcal{T} \cup FO(\mathcal{P})$. $\mathcal{H}$ is called *FOL-satisfiable* if it has at least a FOL-model.

Finally, we define skeptical entailment under the FOL semantics. A sentence $\varphi \in \mathcal{L}$ is *FOL-entailed* by $\mathcal{H}$, denoted by $\mathcal{H} \models_{FOL} \varphi$ iff, for each FOL-model $\mathcal{I}$ of $\mathcal{H}$, $\mathcal{I}$ satisfies φ.

Notice that the above first-order semantics of rules does not distinguish between negated atoms in the body and disjunction in the head of rules: e.g., according to such semantics, the rules $A \leftarrow B, \mathit{not}\, C$ and $A \vee C \leftarrow B$ have the same meaning.

2.3 Nonmonotonic semantics for r-hybrid KBs

An alternative semantics to r-hybrid KBs is based on a nonmonotonic interpretation of the rule component, according to the notion of *stable model* [16]. This is the semantics most commonly adopted in Disjunctive Logic Programming (DLP). We now formalize such a semantics in the framework of r-hybrid KBs.

We start by introducing some auxiliary definitions.

Given an interpretation $\mathcal{I}$, we denote by $\mathcal{I}_R$ the projection of $\mathcal{I}$ to $\mathcal{A}_R$, i.e., $\mathcal{I}_R$ is obtained from $\mathcal{I}$ by restricting it to the interpretation of the predicates in $\mathcal{A}_R$. Analogously, we denote by $\mathcal{I}_P$ the projection of $\mathcal{I}$ to $\mathcal{A}_P$, and denote $\mathcal{I}$ as $\mathcal{I}_P \cup \mathcal{I}_R$.

Given a set of constants $\mathcal{C}$, the *ground instantiation of* $\mathcal{P}$ *with respect to* $\mathcal{C}$, denoted by $gr(\mathcal{P}, \mathcal{C})$, is the program obtained from $\mathcal{P}$ by replacing every rule R in $\mathcal{P}$ with the set of rules obtained by applying all possible substitutions of variables in R with constants in $\mathcal{C}$.

Given an interpretation $\mathcal{I}$ of an alphabet of predicates $\mathcal{A}' \subset \mathcal{A}$, and a ground program $\mathcal{P}_g$ over the predicates in $\mathcal{A}$, the *projection of* $\mathcal{P}_g$ *with respect to* $\mathcal{I}$, denoted by $\Pi(\mathcal{P}_g, \mathcal{I})$, is the ground program obtained from $\mathcal{P}_g$ as follows. For each rule $R \in \mathcal{P}_g$:

- delete R if there exists an atom $r(t)$ in the head of R such that $r \in \mathcal{A}'$ and $t \in r^{\mathcal{I}}$;
- delete each atom $r(t)$ in the head of R such that $r \in \mathcal{A}'$ and $t \not\in r^{\mathcal{I}}$;
- delete R if there exists an atom $r(t)$ in the body of R such that $r \in \mathcal{A}'$ and $t \not\in r^{\mathcal{I}}$;
- delete each atom $r(t)$ in the body of R such that $r \in \mathcal{A}'$ and $t \in r^{\mathcal{I}}$;

Informally, the projection of $\mathcal{P}_g$ with respect to $\mathcal{I}$ corresponds to evaluating $\mathcal{P}_g$ with respect to $\mathcal{I}$, thus eliminating from $\mathcal{P}_g$ every atom whose predicate is interpreted in $\mathcal{I}$. Thus, when $\mathcal{A}' = \mathcal{A}_P$, all occurrences of structural predicates are eliminated in the projection of $\mathcal{P}_g$ with respect to $\mathcal{I}$, according to the evaluation in $\mathcal{I}$ of the atoms with structural predicates occurring in $\mathcal{P}_g$.

Given two interpretations $\mathcal{I}, \mathcal{I}'$ of the set of predicates $\mathcal{A}$, we write $\mathcal{I}' \subset_{\mathcal{A}} \mathcal{I}$ if (i) for each $p \in \mathcal{A}$, $p^{\mathcal{I}'} \subseteq p^{\mathcal{I}}$, and (ii) there exists $p \in \mathcal{A}$ such that $p^{\mathcal{I}'} \subset p^{\mathcal{I}}$. In words, $\mathcal{I}' \subset_{\mathcal{A}} \mathcal{I}$ if the extension of the predicates of $\mathcal{A}$ in $\mathcal{I}$ is strictly larger than in $\mathcal{I}'$.

Given a positive ground Datalog$^{\neg\vee}$ program $\mathcal{P}$ over an alphabet of predicates $\mathcal{A}_R$ and an interpretation $\mathcal{I}$, we say that $\mathcal{I}$ is a *minimal model* of $\mathcal{P}$ if $\mathcal{I}$ satisfies $FO(\mathcal{P})$ and there is no interpretation $\mathcal{I}'$ such that $\mathcal{I}'$ satisfies $FO(\mathcal{P})$ and $\mathcal{I}' \subset_{\mathcal{A}_R} \mathcal{I}$.

Given a ground Datalog$^{\neg\vee}$ program $\mathcal{P}$ and an interpretation $\mathcal{I}$ for $\mathcal{P}$, the *GL-reduct* [16] of $\mathcal{P}$ with respect to $\mathcal{I}$, denoted by $GL(\mathcal{P}, \mathcal{I})$, is the positive ground program obtained from $\mathcal{P}$ as follows.

For each rule $R \in \mathcal{P}$: (i) delete R if there exists a negated atom $not\, r(t)$ in the body of R such that $t \in r^{\mathcal{I}}$; (ii) delete each negated atom $not\, r(t)$ in the body of R such that $t \notin r^{\mathcal{I}}$.

Given a ground Datalog$^{\neg\vee}$ program $\mathcal{P}$ and an interpretation $\mathcal{I}$, $\mathcal{I}$ is a *stable model* for $\mathcal{P}$ iff $\mathcal{I}$ is a minimal model of $GL(\mathcal{P},\mathcal{I})$.

Definition 2 *An interpretation $\mathcal{I}$ is a NM-model for $\mathcal{H} = (\mathcal{T},\mathcal{P})$ if the following conditions hold:*

1. *$\mathcal{I}_P$ satisfies $\mathcal{T}$;*

2. *$\mathcal{I}_R$ is a stable model for $\Pi(gr(\mathcal{P},\mathcal{C}),\mathcal{I}_P)$.*

*$\mathcal{H}$ is called NM-*satisfiable *(or simply* satisfiable*) if $\mathcal{H}$ has at least a NM-model.*

Finally, we define skeptical entailment in r-hybrid KBs under the nonmonotonic semantics, which is analogous to the previous notion of entailment under the first-order semantics. We say that a first-order sentence $\varphi \in \mathcal{L}$ is *NM*-entailed by $\mathcal{H}$, denoted by $\mathcal{H} \models_{NM} \varphi$ iff, for each *NM*-model $\mathcal{I}$ of $\mathcal{H}$, $\mathcal{I}$ satisfies φ.

In other words, the nonmonotonic semantics for a r-hybrid KB $\mathcal{H} = (\mathcal{T},\mathcal{P})$ is obtained in the following way. Take a first-order interpretation $\mathcal{I} = \mathcal{I}_P \cup \mathcal{I}_R$ such that $\mathcal{I}_P$ satisfies $\mathcal{T}$; then, evaluate $\mathcal{P}$ in $\mathcal{I}_P$, obtaining the program $\Pi(gr(\mathcal{P},\mathcal{C}),\mathcal{I}_P)$; if $\mathcal{I}_R$ is a stable model for such a program, then $\mathcal{I}$ is a *NM*-model for $\mathcal{H}$.

If the rule component is a disjunctive positive Datalog$^{\neg\vee}$ program, then, from well-known results for disjunctive Datalog [11], it follows that, for each sentence φ, $\mathcal{H} \models_{NM} \varphi$ iff $\mathcal{T} \cup FO(\mathcal{P}) \models_{FOL} \varphi$. That is, entailment under the first-order interpretation and the DLP interpretation of the rule component coincide. In contrast, the two interpretations differ in the presence of negated atoms in the body of program rules. However, it can be shown that satisfiability of r-hybrid KBs under the first-order semantics can be reduced to satisfiability under the nonmonotonic semantics. Indeed, let $\mathcal{H} = (\mathcal{T},\mathcal{P})$ and let $\mathcal{P}'$ be the program obtained as follows: for every rule R of the form (1), add to $\mathcal{P}'$ the rule

$$p_1(X_1) \vee \ldots \vee p_n(X_n) \vee u_1(W_1) \ldots \vee u_h(W_h) \leftarrow r_1(Y_1), \ldots, r_m(Y_m), s_1(Z_1), \ldots, s_k(Z_k)$$

From the first-order semantics of r-hybrid KBs defined in Section 2.2, it follows that $\mathcal{H}$ is FOL-satisfiable iff $\mathcal{H}' = (\mathcal{T},\mathcal{P}')$ is FOL-satisfiable; moreover, from the above explained equivalence between the two semantics for disjunctive positive Datalog$^{\neg\vee}$ programs, it follows that $\mathcal{H}'$ is FOL-satisfiable iff $\mathcal{H}'$ is *NM*-satisfiable.

Therefore, in the rest of the paper, we study r-hybrid KBs under the nonmonotonic semantics. In particular, when we speak about satisfiability of r-hybrid KBs we always mean satisfiability under the nonmonotonic semantics.

We conclude the section with a simple example of r-hybrid KB.

Example 3 Let $\mathcal{H}$ be the r-hybrid KB where the following structural component $\mathcal{T}$ defines an ontology about persons:

$$\begin{array}{l}
\forall x.PERSON(x) \rightarrow \exists y.FATHER(y,x) \wedge MALE(y)\\
\forall x.MALE(x) \rightarrow PERSON(x)\\
\forall x.FEMALE(x) \rightarrow PERSON(x)\\
\forall x.FEMALE(x) \rightarrow \neg MALE(x)\\
MALE(Bob)\\
PERSON(Mary)\\
PERSON(Paul)
\end{array}$$

and the rule component $\mathcal{P}$ defines nonmonotonic rules about students, as follows:

$\mathit{boy}(X) \leftarrow \mathit{enrolled}(X, c1), \mathit{PERSON}(X), \mathit{not}\ \mathit{girl}(X)\ \ [R1]$
$\mathit{girl}(X) \leftarrow \mathit{enrolled}(X, c2), \mathit{PERSON}(X)\ \ [R2]$
$\mathit{boy}(X) \vee \mathit{girl}(X) \leftarrow \mathit{enrolled}(X, c3), \mathit{PERSON}(X)\ \ [R3]$
$\mathit{FEMALE}(X) \leftarrow \mathit{girl}(X)\ \ [R4]$
$\mathit{MALE}(X) \leftarrow \mathit{boy}(X)\ \ [R5]$
$\mathit{enrolled}(\mathit{Paul}, c1)$
$\mathit{enrolled}(\mathit{Mary}, c1)$
$\mathit{enrolled}(\mathit{Mary}, c2)$
$\mathit{enrolled}(\mathit{Bob}, c3)$

It can be easily verified that all *NM*-models for $\mathcal{H}$ satisfy the following ground atoms:

- $\mathit{boy}(\mathit{Paul})$ (since rule R1 is always applicable for $X = \mathit{Paul}$ and R1 acts like a *default rule*, which can be read as follows: if X is a person enrolled in course $c1$, then X is a boy, unless we know for sure that X is a girl)
- $\mathit{girl}(\mathit{Mary})$ (since rule R2 is always applicable for $X = \mathit{Mary}$)
- $\mathit{boy}(\mathit{Bob})$ (since rule R3 is always applicable for $X = \mathit{Bob}$, and, by rule R4, the conclusion $\mathit{girl}(\mathit{Bob})$ is inconsistent with $\mathcal{T}$)
- $\mathit{MALE}(\mathit{Paul})$ (due to rule R5)
- $\mathit{FEMALE}(\mathit{Mary})$ (due to rule R4)

Notice that $\mathcal{H} \models_{NM} \mathit{FEMALE}(\mathit{Mary})$, while $\mathcal{T} \not\models_{FOL} \mathit{FEMALE}(\mathit{Mary})$. In other words, adding a rule component has indeed an effect on the conclusions one can draw about structural predicates. Moreover, such an effect also holds under the first-order semantics of r-hybrid KBS, since it can be immediately verified that in this case $\mathcal{H} \models_{FOL} \mathit{FEMALE}(\mathit{Mary})$. □

Among other things, the above example shows that, in r-hybrid KBs, the information flow is bidirectional: not only the structural component constrains the forms of the stable models of the rule component (through the structural predicates in the body of the rules), but also vice versa, since the rule component imposes, through the rules with structural predicates in the head, constraints that the models of the structural components must satisfy. Hence, the rule component has an effect on the conclusions that can be drawn from the structural component, since it filters out those models $\mathcal{I}_P$ of the structural component for which the program $\Pi(gr(\mathcal{P}, \mathcal{C}), \mathcal{I}_P)$ has no stable models. However, as will be clear in the next section, such an effect on the structural component is limited to a form of "extensional" knowledge (e.g., instance assertions in Description Logics KBs), which does not affect purely structural (intensional) knowledge (e.g., subsumption relationships).

3 Reasoning in r-hybrid KBs

In this section we study reasoning in r-hybrid KBs. In particular, we study satisfiability of r-hybrid KBs, which is the basic reasoning task: as in many other logics, in r-hybrid KBs many important reasoning tasks (e.g., skeptical entailment) can be easily reduced to (un)satisfiability.

In the following, we first define an algorithm for deciding satisfiability of r-hybrid KBs; then, based on such an algorithm, we analyze decidability and complexity of reasoning in r-hybrid KBs, and examine in particular the case when Description Logics are used to specify the structural component of r-hybrid KBs.

3.1 The algorithm

In order to provide a reasoning method for r-hybrid KBs, we give the following preliminary definitions.

First, given a Datalog$^{\neg\vee}$ program $\mathcal{P}$, we denote by $\mathcal{C}_{\mathcal{P}}$ the set of constant symbols occurring in $\mathcal{P}$, and denote by $\mathcal{A}_P/\mathcal{P}$ the set of predicates from $\mathcal{A}_P$ occurring in $\mathcal{P}$.

Definition 4 *Let $\mathcal{H} = (\mathcal{T}, \mathcal{P})$ be a r-hybrid KB. The* grounding of the structural predicates in $\mathcal{P}$, *denoted by $gr_p(\mathcal{P})$ is the following set of ground atoms:*

$$gr_p(\mathcal{P}) = \{m(t) \mid m \in \mathcal{A}_P/\mathcal{P} \textit{ and } m \textit{ has arity } k \textit{ and } t \textit{ is a } k\textit{-tuple of constants of } \mathcal{C}_{\mathcal{P}}\}$$

The idea behind the above definition is that, in the case of r-hybrid KBs, $gr_p(\mathcal{P})$ identifies the set of *all* the relevant instantiations of the predicates in $\mathcal{A}_P$ needed to decide satisfiability of the rule component of the r-hybrid KB $\mathcal{H}$. In fact, due to the safeness condition in the program rules, we can restrict the grounding of the rules only to the instantiations which substitute each variable with a symbol in $\mathcal{C}_{\mathcal{P}}$: indeed, it can be immediately verified that the stable models of $gr(\mathcal{P}, \mathcal{A}_C)$ and of $gr(\mathcal{P}, \mathcal{C}_{\mathcal{P}})$ coincide (notice that $gr(\mathcal{P}, \mathcal{C}_{\mathcal{P}})$ is the standard grounding in Datalog, i.e., the grounding of the program on the set of constants occurring in it).

Thus, we can divide the set of all interpretations for $\mathcal{T}$ into equivalence classes, based on the way in which such interpretations evaluate the ground atoms in $gr_p(\mathcal{P})$. Each such equivalence class can be represented by a partition (G_P, G_N) of $gr_p(\mathcal{P})$. More precisely, G_P is the set of ground atoms in $gr_p(\mathcal{P})$ satisfied by the interpretations in the equivalence class, while G_N is the set of atoms in $gr_p(\mathcal{P})$ which are not satisfied by such interpretations.

However, not all the partitions of $gr_p(\mathcal{P})$ represent a guess of the ground atoms that is compatible with the KB $\mathcal{T}$. The following definition formalizes the notion of consistency of a partitions of ground atoms with respect to $\mathcal{T}$.

Definition 5 *A partition (G_P, G_N) of $gr_p(\mathcal{P})$* is consistent with $\mathcal{T}$ *iff the first-order theory*

$$\mathcal{T} \cup \{m(t) \mid m(t) \in G_P\} \cup \{\neg m(t) \mid m(t) \in G_N\}$$

is satisfiable.

Informally, the above definition indicates that, if a partition is consistent with $\mathcal{T}$, then there exists at least one interpretation that both satisfies $\mathcal{T}$ and evaluates the atoms in $gr_p(\mathcal{P})$ according to the partition (G_P, G_N).

We now define the notion of partition of ground atoms induced by an interpretation.

Definition 6 *Let $\mathcal{I}$ be an interpretation and Let (G_P, G_N) be the partition of $gr_p(\mathcal{P})$ such that, for each $r(t) \in gr_p(\mathcal{P})$, $t \in r^{\mathcal{I}}$ if and only if $r(t) \in G_P$. We call (G_P, G_N) the partition of $gr_p(\mathcal{P})$* induced by $\mathcal{I}$.

Finally, we denote by $\mathcal{P}(G_P, G_N)$ the Datalog$^{\neg\vee}$ program

$$\mathcal{P}(G_P, G_N) = \mathcal{P} \cup G_P \cup \{\leftarrow r(t) \mid r(t) \in G_N\}$$

We are now ready to define an algorithm for reasoning in r-hybrid KBs. In Figure 1 we report the algorithm R-Hybrid-Sat for deciding satisfiability of a r-hybrid KB $\mathcal{H} = (\mathcal{T}, \mathcal{P})$. The algorithm formalizes the idea that a way to decide satisfiability of $\mathcal{H}$ is to look for a partition of $gr_p(\mathcal{P})$ that is consistent with $\mathcal{T}$ and such that the program $\mathcal{P}(G_P, G_N)$ has a stable model.

```
Algorithm R-Hybrid-Sat($\mathcal{H}$)
Input: r-hybrid KB $\mathcal{H} = (\mathcal{T}, \mathcal{P})$
Output: true if $\mathcal{H}$ is satisfiable, false otherwise
begin
  if there exists partition $(G_P, G_N)$ of $gr_p(\mathcal{P})$
  such that
    (a) $(G_P, G_N)$ is consistent with $\mathcal{T}$ and
    (b) $\mathcal{P}(G_P, G_N)$ has a stable model
  then return true
  else return false
end
```

Figure 1: The algorithm R-Hybrid-Sat

3.2 Correctness

We now prove soundness and completeness of the algorithm. In the following, we denote by $\mathcal{I}_{G_P}$ the interpretation of $\mathcal{A}_P/\mathcal{P}$ defined as follows: $t \in r^{\mathcal{I}_{G_P}}$ iff $r(t) \in G_P$. We first prove the following lemma.

Lemma 7 *Let $\mathcal{H} = (\mathcal{T}, \mathcal{P})$ be a r-hybrid KB, and let $\mathcal{I}$ be an interpretation. $\mathcal{I}_R$ is a stable model for $\Pi(gr(\mathcal{P}, \mathcal{C}), \mathcal{I}_P)$ if and only if $\mathcal{I}_{G_P} \cup \mathcal{I}_R$ is a stable model for $\mathcal{P}(G_P, G_N)$.*

Proof. First, due to the safeness condition in Definition 1, it can be easily verified that every interpretation is a stable model for $\Pi(gr(\mathcal{P}, \mathcal{C}), \mathcal{I}_P)$ if and only if it is a stable model for $\Pi(gr(\mathcal{P}, \mathcal{C}_\mathcal{P}), \mathcal{I}_P)$.

Now let $\mathcal{P}' = \Pi(gr(\mathcal{P}, \mathcal{C}), \mathcal{I}_P) \cup G_P \cup \{\leftarrow r(t) \mid r(t) \in G_N\}$. It is immediate to verify that $\mathcal{P}'$ is equivalent to $\mathcal{P}(G_P, G_N)$, in the sense that the set of stable models of the two programs is the same. Moreover, since $\Pi(gr(\mathcal{P}, \mathcal{C}), \mathcal{I}_P)$ does not contain occurrences of predicates of $\mathcal{A}_P/\mathcal{P}$, it follows that the set of stable models of $\mathcal{P}'$ is $\{\mathcal{I}_{G_P} \cup \mathcal{I}' \mid \mathcal{I}'$ is a stable model for $\Pi(gr(\mathcal{P}, \mathcal{C}), \mathcal{I}_P)\}$, which proves the thesis. □

Theorem 8 *Let $\mathcal{H} = (\mathcal{T}, \mathcal{P})$ be a r-hybrid KB. Then, $\mathcal{H}$ is satisfiable iff R-Hybrid-Sat($\mathcal{H}$) returns* true.

Proof. (⇒) Suppose $\mathcal{H}$ is satisfiable. Then, there exists an interpretation $\mathcal{I}$ such that $\mathcal{I}$ satisfies $\mathcal{T}$ and $\mathcal{I}_R$ is a stable model for $\Pi(gr(\mathcal{P}, \mathcal{C}), \mathcal{I}_P)$. Let (G_P, G_N) be the partition of $gr_p(\mathcal{P})$ induced by $\mathcal{I}$: by Definition 6, $\mathcal{I}$ satisfies $\{m(t) \mid m(t) \in G_P\} \cup \{\neg m(t) \mid m(t) \in G_N\}$. Since $\mathcal{I}$ also satisfies $\mathcal{T}$, by Definition 5 it follows that (G_P, G_N) is consistent with $\mathcal{T}$. Moreover, by Lemma 7, $\mathcal{I}_{G_P} \cup \mathcal{I}_R$ is a stable model for $\mathcal{P}(G_P, G_N)$. Therefore, (G_P, G_N) satisfies both condition (a) and condition (b) of the algorithm, hence R-Hybrid-Sat($\mathcal{H}$) returns true.

(⇐) Suppose R-Hybrid-Sat($\mathcal{H}$) returns true. Then, by condition (a) of the algorithm, there exists a partition (G_P, G_N) of $gr_p(\mathcal{P})$ that is consistent with $\mathcal{T}$, which implies there exists an interpretation $\mathcal{I}_P$ of $\mathcal{A}_P$ such that $\mathcal{I}_P$ satisfies $\mathcal{T}$ and (G_P, G_N) is the partition of $gr_p(\mathcal{P})$ induced by $\mathcal{I}_P$. Moreover, by condition (b) of the algorithm, the program $\mathcal{P}(G_P, G_N)$ has a stable model. Let $\mathcal{I}_{G_P} \cup \mathcal{I}_R$ be such a model. By Lemma 7, $\mathcal{I}_R$ is a stable model for $\Pi(gr(\mathcal{P}, \mathcal{C}), \mathcal{I}_P)$. Now let $\mathcal{I}$ be the interpretation obtained by composing $\mathcal{I}_P$ and $\mathcal{I}_R$, i.e., $\mathcal{I}$ is such that $r^\mathcal{I} = r^{\mathcal{I}_P}$ for every $r \in \mathcal{A}_P$, and $r^\mathcal{I} = r^{\mathcal{I}_R}$ for every $r \in \mathcal{A}_R$. $\mathcal{I}$ satisfies both conditions of Definition 2: consequently, $\mathcal{H}$ is satisfiable. □

3.3 Decidability and complexity

We now study decidability and complexity issues in the framework of r-hybrid KBs.

We start by recalling a decidability and complexity result for Datalog$^{\neg\vee}$ programs.

Proposition 9 ([11]) *Satisfiability of Datalog$^{\neg\vee}$ programs is NEXPTIME^{NP}-complete. Moreover, satisfiability of Datalog$^{\neg}$ programs is NEXPTIME-complete.*

Then, since satisfiability of a Datalog$^{\neg\vee}$ program $\mathcal{P}$ can be trivially reduced to satisfiability of the r-hybrid KB $(\emptyset, \mathcal{P})$, the following hardness result follows.

Lemma 10 *Satisfiability of r-hybrid KBs is NEXPTIME^{NP}-hard. Moreover, it is NEXPTIME-hard if the rule component is a Datalog$^{\neg}$ program.*

We now prove a very general result on the decidability of reasoning in r-hybrid KBs.

Theorem 11 *Let $\mathcal{H} = (\mathcal{T}, \mathcal{P})$ be a r-hybrid KB. If establishing consistency of a partition of $gr_p(\mathcal{P})$ with $\mathcal{T}$ is decidable, then satisfiability of $\mathcal{H}$ is a decidable problem.*

Proof. First, observe that the set $gr_p(\mathcal{P})$ is finite, therefore the number of partitions of $gr_p(\mathcal{P})$ is finite. Then, since by hypothesis establishing consistency of a partition (G_P, G_N) with $\mathcal{T}$ is decidable, for each such partition (G_P, G_N) condition (a) of the algorithm can be verified in a finite amount of time; moreover, since $\mathcal{P}(G_P, G_N)$ is a finite Datalog$^{\neg\vee}$ program, from Proposition 9 it follows that condition (b) of the algorithm can also be verified in a finite amount of time. □

We remark that, starting from a logic $\mathcal{L}$ in which reasoning is decidable, it is very often the case that deciding satisfiability of a theory of an $\mathcal{L}$-KB augmented with a finite set of ground literals is still decidable, and therefore that reasoning in r-hybrid KBs made of $\mathcal{L}$ theories as structural components is decidable. In this sense, the previous theorem can be read as a very strong result, stating that the framework of r-hybrid KBs generally preserves decidability of reasoning.

In this respect, we now analyze the case when the structural language $\mathcal{L}$ is a *description logic* (DL) [5], which is of particular interest, since Description Logics are the subsets of first-order logic commonly used in the Semantic Web to specify ontologies [22].

The language of DLs is a syntactic variant of a subset of function-free first-order logic in which only unary and binary predicates are allowed. Unary predicates are called *concepts*, binary predicates are called *roles*. Moreover, there are precise syntactic restrictions concerning the use of quantifiers and boolean connectives in the logic.

Generally speaking, a DL defines a syntax for forming *concept expressions* and *role expressions*. Such a syntax imposes a restriction on the use of boolean connectives and first-order quantifiers in the logic. Different DLs are obtained by imposing different restrictions in the concept and role expressions allowed. A DL-KB is typically constituted of a TBox and an ABox. A TBox is a set of *inclusion assertions*, i.e., expressions of the form $C_1 \sqsubseteq C_2$ or $R_1 \sqsubseteq R_2$, where C_1, C_2 are concept expressions, and R_1, R_2 are role expressions. An assertion $C_1 \sqsubseteq C_2$ is semantically equivalent to a first-order sentence of the form $\forall x.FOL(C_1(x)) \rightarrow FOL(C_2(x))$, where $FOL(C(x))$ is an open formula with free variable x obtained by translating the concept expression C to first-order logic [5]. Analogously, an assertion $R_1 \sqsubseteq R_2$ corresponds to a first-order sentence $\forall x.FOL(R_1(x, y)) \rightarrow FOL(R_2(x, y))$, where $FOL(R(x, y))$ is an open formula with free variables x, y obtained by translating the concept expression R to first-order logic. An ABox is a set *instance assertions* ground atoms of the form $C(a)$, $R(a, b)$, where C is a concept name, R is a role name, and a, b are constants (called *individuals* in DLs).

Now let us go back to the problem of reasoning in DL-based r-hybrid KBs, in particular deciding the consistency of a partition of ground atoms (G_P, G_N) with a DL-KB $\mathcal{T}$: actually, the syntax of DL-KBs does not immediately imply that the theory $\mathcal{T} \cup \{m(t) \mid m(t) \in G_P\} \cup \{\neg m(t) \mid m(t) \in G_N\}$ can be encoded in terms of a DL-KB, since most DLs do not allow negation of arbitrary ground atoms in the KB. However, recent results on *boolean ABoxes* [35, 4] immediately imply that, under general conditions, it is easy to (polynomially) encode a set of ground literals in terms of an equivalent set of assertions allowed in the DL-KB. Therefore, in such cases, extending a decidable DL by allowing negated ground atoms in the ABox (in particular, the negation of role instance assertions) preserves decidability of reasoning.

Moreover, since in almost all DLs only unary and binary predicates are used, the size of $gr_p(\mathcal{P})$ is at most polynomial (quadratic) in the size of $\mathcal{H}$. This allows us to easily establish, through the algorithm R-Hybrid-Sat, the computational characterization of reasoning in DL-based r-hybrid KBs.

In particular, the DL that currently plays a central role in the Semantic Web is $\mathcal{SHOIN}(\mathbf{D})$: as mentioned in Section 1, it is equivalent to OWL-DL [32], which is a W3C recommendation language for ontology representation in the Semantic Web. Reasoning in $\mathcal{SHOIN}(\mathbf{D})$, and hence in OWL-DL, is decidable, as stated by the following property.

Proposition 12 ([20, 35]) *Satisfiability of $\mathcal{SHOIN}(\mathbf{D})$ KBs is NEXPTIME-complete.*

We now prove that reasoning in $\mathcal{SHOIN}(\mathbf{D})$ r-hybrid KBs is decidable, and provide a computational characterization of the problem.

Theorem 13 *Let $\mathcal{H} = (\mathcal{T}, \mathcal{P})$ be a r-hybrid KB where $\mathcal{T}$ is a $\mathcal{SHOIN}(\mathbf{D})$ KB and $\mathcal{P}$ is a Datalog$^{\neg\vee}$ program. Deciding satisfiability of $\mathcal{H}$ is NEXPTIME^{NP}-complete. Moreover, if $\mathcal{P}$ is a Datalog$^{\neg}$ program, deciding satisfiability of $\mathcal{H}$ is NEXPTIME-complete.*

Proof. Given a $\mathcal{SHOIN}(\mathbf{D})$ KB $\mathcal{T}$, it can be immediately verified that a partition (G_P, G_N) is consistent with $\mathcal{T}$ if and only if the $\mathcal{SHOIN}(\mathbf{D})$ KB

$$\begin{aligned} \mathcal{T}' \;=\; & \mathcal{T} \cup \{C(a) \mid C(a) \in G_P\} \cup \{\neg C(a) \mid C(a) \in G_N\} \cup \\ & \{\exists R.\{b\}(a) \mid R(a,b) \in G_P\} \cup \{\neg\exists R.\{b\}(a) \mid R(a,b) \in G_N\} \end{aligned}$$

is satisfiable (see e.g., [31]). Observe that, since $gr_p(\mathcal{P})$ has size polynomial in the size of $\mathcal{H}$, $\mathcal{T}'$ has also size polynomial in the size of $\mathcal{H}$. Consequently, from Proposition 12, deciding satisfiability of $\mathcal{T}'$ (and hence verifying condition (a) of the algorithm) is in NEXPTIME. Moreover, since $gr_p(\mathcal{P})$ has size polynomial in the size of $\mathcal{H}$, the Datalog$^{\neg\vee}$ program $\mathcal{P}(G_P, G_N)$ has also size polynomial in the size of $\mathcal{H}$. Consequently, from Proposition 9 it follows that verifying condition (b) of the algorithm is in NEXPTIME^{NP}, and is in NEXPTIME if $\mathcal{P}$ is a Datalog$^{\neg}$ program. Now, since the algorithm R-Hybrid-Sat has to nondeterministically find a partition verifying condition (a) and condition (b), we immediately obtain an upper bound of NEXPTIME^{NP}, and of NEXPTIME if $\mathcal{P}$ is a Datalog$^{\neg}$ program. Then, since, in the two respective cases, by Lemma 10 the problem is also hard with respect to such complexity classes, the thesis follows. □

As a byproduct, the previous theorem closes an open problem in [31], i.e., decidability of satisfiability of $\mathcal{SHOIN}(\mathbf{D})$ with DL-safe rules, although under a slightly different semantics (see the remark below). This problem exactly corresponds in our framework to deciding satisfiability of a r-hybrid KB composed of a $\mathcal{SHOIN}(\mathbf{D})$ KB and a positive Datalog program: as a corollary of the above results, it immediately follows that satisfiability in such r-hybrid KBs is decidable and is

NEXPTIME-complete. Moreover, the previous theorem not only shows that adding positive Datalog programs (satisfying the safeness condition) to OWL-DL preserves decidability, but also that we can further extend the rule component language to Datalog$^{\neg\vee}$, and that the overall complexity of reasoning is no worse than reasoning in the rule component only.

Remark. We recall here that the semantics of r-hybrid KBs is based on the standard names assumption, which implies the unique name assumption over constants (two distinct constants denote two different individuals), as explained in Section 2.2. However, differently from most DLs, in its original definition $\mathcal{SHOIN}(\mathbf{D})$ does not adopt the unique name assumption. Therefore, we point out that the above results concerning $\mathcal{SHOIN}(\mathbf{D})$ hold for the version of this logic in which the unique name assumption is adopted.

4 Related work

In this section we relate our approach to recent work in integrating ontologies and rules. We divide such studies in two main streams: (i) studies that deal with forms of "safe" (or loose) interaction between the structural and the rule components, and hence close, at least conceptually, to our proposal; (ii) studies concerning forms of "non-safe" (or strict) interaction.

4.1 Safe interaction

The first formal proposal for the integration of Description Logics and rules is $\mathcal{AL}$-log [9]. $\mathcal{AL}$-log is a framework which integrates KBs expressed in the description logic $\mathcal{ALC}$ and positive Datalog programs. Then, *disjunctive* $\mathcal{AL}$*-log* was proposed in [34] as an extension of $\mathcal{AL}$-log, based on the use of Datalog$^{\neg\vee}$ instead of positive Datalog, and on the possibility of using binary predicates (roles) besides unary predicates (concepts) in rules. When choosing $\mathcal{ALC}$ as the structural language, the framework of r-hybrid KBs captures disjunctive $\mathcal{AL}$-log and can be seen as a generalization of it: indeed, differently from r-hybrid KBs, in disjunctive $\mathcal{AL}$-log structural predicates can occur only in the bodies of rules, which restricts the information flow only from the structural KB to the rule KB, but not vice versa.

The framework of $\mathcal{AL}$-log has been extended in a different way in [31]. There, the problem of extending OWL-DL with positive Datalog programs is analyzed. The interaction between OWL-DL and rules is restricted through a safeness condition which is exactly the one adopted in r-hybrid KBs. With respect to disjunctive $\mathcal{AL}$-log, in [31] a more expressive structural language and a less expressive rule language are adopted: moreover, the information flow is bidirectional, i.e., structural predicates may appear in the head of rules. As we have shown in Section 3, such a framework is perfectly captured by r-hybrid KBs.

The work presented in [17] can also be seen as an approach based on a form of safe interaction between the structural DL-KB and the rules: in particular, a rule language is defined such that it is possible to encode a set of rules into a semantically equivalent DL-KB. As a consequence, such a rule language is very restricted.

A different approach is presented in [19, 18], which proposes Conceptual Logic Programming (CLP), an extension of answer set programming (i.e., Datalog$^{\neg\vee}$) towards infinite domains. In order to keep reasoning decidable, a syntactic restriction on CLP program rules is imposed. This approach is related to integrating DLs and rules, since the authors also show that CLPs can embed expressive DL-KBs, which in turn implies decidability of adding CLP rules to such DLs. However, the syntactic restriction on CLP rules, whose purpose is to impose a "forest-like" structure to the models of the program, is different from the safeness conditions analyzed so far, which makes it impossible to compare this approach with r-hybrid KBs (and with the approaches previously mentioned).

Another approach for extending DLs with Datalog$^{\neg}$ rules is presented in [13, 14]. Differently from r-hybrid KBs and from the other approaches above described, this proposal allows for specifying in rule bodies *queries* to the structural component, where every query also allows for specifying an input from the rule component, and thus for an information flow from the rule component to the structural component. The meaning of such queries in rule bodies is given at the meta-level, through the notion of skeptical entailment in the DL-KB. Thus, from the semantic viewpoint, this form of interaction-via-entailment between the two components is more restricted than in r-hybrid KBs (and in the similar approaches previously mentioned); on the other hand, such an increased separation in principle allows for more modular reasoning methods, which are able to completely separate reasoning about the structural component and reasoning about the rule component. However, in this paper we have shown that an analogous form of modularization of reasoning is possible also in the presence of a semantically richer form of interaction between the two components of a r-hybrid KB.

Finally, [2, 1, 3] present approaches for the combination of defeasible reasoning with Description Logics, under a safe interaction-via-entailment scheme which is semantically analogous to the one proposed in [13]. Besides the differences with our approach (and with the studies on nonmonotonic extensions of DL-KBs previously mentioned) concerning the semantics of nonmonotonic rules, a main characteristic of these proposals consists in the fact the information flow is unidirectional, i.e., it goes from the structural component to the rule component.

4.2 Non-safe interaction

Research in non-safe interaction of DLs and rules actually started with the work on CARIN [25, 26, 27], which established very important undecidability results concerning non-safe interaction between DL-KBs and rules. Roughly speaking, such results clearly indicate that, in case of unrestricted interaction between the structural component and the rule component in hybrid KBs, decidability of reasoning holds only if at least one of the two component KBs has very limited expressive power: e.g., in order to retain decidability of reasoning, allowing recursion in the rule KB imposes very severe restrictions on the expressiveness of the structural KB.

Then, we remark that query answering over a knowledge base can be seen as a problem of reasoning in a hybrid KB in which the rule KB corresponds to the query. In this respect, an important undecidability result concerning query answering over databases with integrity constraints is reported in [7]. More precisely, it is shown that answering recursive Datalog queries over a database with simple integrity constraints (keys, foreign keys), interpreted as a knowledge base, i.e., under an open-world assumption, is undecidable. This setting also can be viewed as a hybrid KB with non-safe interaction between a knowledge base (database with integrity constraints) and a rule component (the query).

Generally speaking, it is difficult to provide a good semantic account for non-safe interaction between DL-KBs and nonmonotonic rules, due to the classical, open-world semantics of DL-KBs, and the closed-world assumption underlying nonmonotonic systems. For instance, [30] illustrates the problems in providing a semantic account for non-safe interaction of ontologies and Datalog$^{\neg\vee}$ programs.

4.3 Comparison

Summarizing, what emerges from the studies in hybrid KBs is that, while, on the one hand, a safe form of interaction between structural and rule KBs generally allows for decidable reasoning and nice computational properties, on the other hand, the results concerning non-safe interaction indicate that a tight connection between the two components can only be obtained at the price of severely restricting the expressive power of either the structural KB or the rule KB.

In this respect, the major contribution of the present work is the formal proof that the form of safe interaction introduced in [9] and extended in various forms by [34, 31] can be generally applied, and constitutes a good choice for the design of integrated KBs when we want to keep expressive power both in the structural component and in the rule component, and when decidability and complexity of (sound and complete) reasoning is a crucial aspect: indeed, in general, such safe interaction preserves decidability of reasoning and, in many cases, does not increase the complexity of reasoning, in the sense that reasoning in the integrated KB is computationally no harder than reasoning separately in the two components.

5 Conclusions

The results presented in this paper can be summarized as follows:

- we have defined a very general framework for integrating ontologies and rules;
- we have defined a general, modular reasoning method for r-hybrid KBs;
- based on such a method, we have shown that the safe combination of decidable first-order KBs and Datalog$^{\neg\vee}$ rules preserves decidability of reasoning, under very general conditions. An analogous general property holds for the complexity of reasoning in r-hybrid KBs;
- as a byproduct of our results, we have been able to refine a decidability result in [31], and have shown that extending OWL-DL with safe, positive Datalog rules preserves decidability of reasoning. Moreover, we have been able to further extend this result, showing that reasoning in OWL-DL with safe Datalog$^{\neg\vee}$ rules is decidable.

As for future extensions of the present work, we believe that the framework of r-hybrid KBs will prove very useful both to provide a semantics to systems integrating ontologies (not necessarily expressed through Description Logics) and nonmonotonic rules, and to easily establish decidability and complexity results for reasoning in such systems.

Also, it should be worth studying possible relaxations of the safeness condition in r-hybrid KBs. In this respect, as mentioned in Section 4, a central issue is defining a clear, satisfactory semantics for tightly-coupled hybrid KBs with nonmonotonic rules.

The semantic issue concerning the unique name assumption is also interesting (see Section 3.3). In particular, we believe that the framework of r-hybrid KBs can be extended to deal with a different semantics in which the unique name assumption does not hold.

Finally, it would be very interesting to study *data* complexity in the framework of r-hybrid KBs, continuing the research presented in [6], which analyzes data complexity for $\mathcal{AL}$-log.

Acknowledgments

The author wishes to thank Diego Calvanese, Maurizio Lenzerini and Daniele Nardi for helpful discussions on this topic. The author is also thankful to the anonymous reviewers for their precious comments. This research has been partially supported by the projects INFOMIX (IST-2001-33570), SEWASIE (IST-2001-34825) and INTEROP Network of Excellence (IST-508011) funded by the EU, by the project "Società dell'Informazione" subproject SP1 "Reti Internet: Efficienza, Integrazione e Sicurezza" funded by MIUR – Fondo Speciale per lo Sviluppo della Ricerca di Interesse Strategico, and by project HYPER, funded by IBM through a Shared University Research (SUR) Award grant.

References

[1] Grigoris Antoniou. A nonmonotonic rule system using ontologies. In *Proceedings of the First International Workshop on Rules and Rule Markup Languages for the Semantic Web (RuleML 2002)*, volume 60 of *CEUR Workshop Proceedings*, 2002.

[2] Grigoris Antoniou. Nonmonotonic rule systems on top of ontology layers. In *Proceedings of the 2002 International Semantic Web Conference (ISWC 2002)*, pages 394–398, 2002.

[3] Grigoris Antoniou, A. Bikakis, and Gerd Wagner. A system for nonmonotonic rules on the web. In *Proceedings of the Third International Workshop on Rules and Rule Markup Languages for the Semantic Web (RuleML 2004)*, pages 23–36, 2004.

[4] Carlos Areces, Patrick Blackburn, Bernadette Martinez Hernandez, and Maarten Marx. Handling boolean ABoxes. In *Proceedings of the 2003 Description Logic Workshop (DL 2003)*, 2003.

[5] Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation and Applications*. Cambridge University Press, 2003.

[6] Marco Cadoli, Luigi Palopoli, and Maurizio Lenzerini. Datalog and description logics: Expressive power. In *Proceedings of the Sixth International Workshop on Database Programming Languages (DBPL'97)*, 1997.

[7] Diego Calvanese and Riccardo Rosati. Answering recursive queries under keys and foreign keys is undecidable. In *Proceedings of the Tenth International Workshop on Knowledge Representation meets Databases (KRDB 2003)*. CEUR Electronic Workshop Proceedings, `http://ceur-ws.org/Vol-79/`, 2003.

[8] Francesco M. Donini, Maurizio Lenzerini, Daniele Nardi, Werner Nutt, and Andrea Schaerf. An epistemic operator for description logics. *Artificial Intelligence*, 100(1–2):225–274, 1998.

[9] Francesco M. Donini, Maurizio Lenzerini, Daniele Nardi, and Andrea Schaerf. $\mathcal{AL}$-log: Integrating Datalog and description logics. *Journal of Intelligent Information Systems*, 10(3):227–252, 1998.

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

[11] Thomas Eiter, Georg Gottlob, and Heikki Mannilla. Disjunctive Datalog. *ACM Transactions on Database Systems*, 22(3):364–418, 1997.

[12] Thomas Eiter, Nicola Leone, Cristinel Mateis, Gerald Pfeifer, and Francesco Scarcello. The KR system dlv: Progress report, comparison and benchmarks. In *Proceedings of the Sixth International Conference on Principles of Knowledge Representation and Reasoning (KR'98)*, pages 636–647, 1998.

[13] Thomas Eiter, Thomas Lukasiewicz, Roman Schindlauer, and Hans Tompits. Combining answer set programming with description logics for the semantic web. In *Proceedings of the Ninth International Conference on Principles of Knowledge Representation and Reasoning (KR 2004)*, pages 141–151, 2004.

[14] Thomas Eiter, Thomas Lukasiewicz, Roman Schindlauer, and Hans Tompits. Well-founded semantics for description logic programs in the semantic web. In *Proceedings of the Third International Workshop on Rules and Rule Markup Languages for the Semantic Web (RuleML 2004)*, pages 81–97, 2004.

[15] Alan M. Frisch and A. G. Cohn. Thoughts and afterthoughts on the 1988 workshop on principles of hybrid reasonig. *AI Magazine*, 12:77–83, 1991.

[16] Michael Gelfond and Vladimir Lifschitz. Classical negation in logic programs and disjunctive databases. *New Generation Computing*, 9:365–385, 1991.

[17] Benjamin N. Grosof, Ian Horrocks, Raphael Volz, and Stefan Decker. Description logic programs: combining logic programs with description logic. In *Proceedings of the 12th international conference on World Wide Web (WWW 2003)*, pages 48–57, 2003.

[18] Stijn Heymans, Davy Van Nieuwenborgh, and Dirk Vermeir. Semantic web reasoning with conceptual logic programs. In *Proceedings of the Third International Workshop on Rules and Rule Markup Languages for the Semantic Web (RuleML 2004)*, pages 113–127, 2004.

[19] Stijn Heymans and Dirk Vermeir. Integrating description logics and answer set programming. In *Proceedings of the 2003 International Workshop on Principles and Practice of Semantic Web Reasoning (PPSWR 2003)*, pages 146–159, 2003.

[20] Ian Horrocks and Peter F. Patel-Schneider. Reducing OWL entailment to Description Logic satisfiability. In *Proceedings of the 2003 International Semantic Web Conference (ISWC 2003)*, pages 17–29, 2003.

[21] Ian Horrocks and Peter F. Patel-Schneider. A proposal for an OWL rules language. In *Proceedings of the 13th international conference on World Wide Web (WWW 2004)*, pages 723–731, 2004.

[22] Ian Horrocks, Peter F. Patel-Schneider, and Frank van Harmelen. From SHIQ and RDF to OWL: The making of a web ontology language. *Journal of Web Semantics*, 1(1):7–26, 2003.

[23] Hector J. Levesque. All I know: a study in autoepistemic logic. *Artificial Intelligence*, 42:263–310, 1990.

[24] Alon Y. Levy, Anand Rajaraman, and Joann J. Ordille. Query answering algorithms for information agents. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence (AAAI'96)*, pages 40–47, 1996.

[25] Alon Y. Levy and Marie-Christine Rousset. CARIN: A representation language combining Horn rules and description logics. In *Proceedings of the Twelfth European Conference on Artificial Intelligence (ECAI'96)*, pages 323–327, 1996.

[26] Alon Y. Levy and Marie-Christine Rousset. The limits on combining recursive Horn rules with description logics. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence (AAAI'96)*, pages 577–584, 1996.

[27] Alon Y. Levy and Marie-Christine Rousset. Combining Horn rules and description logics in CARIN. *Artificial Intelligence*, 104(1–2):165–209, 1998.

[28] Vladimir Lifschitz. Minimal belief and negation as failure. *Artificial Intelligence*, 70:53–72, 1994.

[29] Robert MacGregor. Inside the LOOM description classifier. *SIGART Bulletin*, 2(3):88–92, 1991.

[30] Jing Mei, Shengping Liu, Anbu Yue, and Zuoquan Lin. An extension to OWL with general rules. In *Proceedings of the Third International Workshop on Rules and Rule Markup Languages for the Semantic Web (RuleML 2004)*, pages 155–169, 2004.

[31] Boris Motik, Ulrike Sattler, and Rudi Studer. Query answering for OWL-DL with rules. In *Proceedings of the 2004 International Semantic Web Conference (ISWC 2004)*, pages 549–563, 2004.

[32] Peter F. Patel-Schneider, Patrick J. Hayes, Ian Horrocks, and Frank van Harmelen. OWL web ontology language; semantics and abstract syntax. W3C candidate recommendation, http://www.w3.org/tr/owl-semantics/, november 2002.

[33] Peter F. Patel-Schneider, Deborah L. McGuiness, Ronald J. Brachman, Lori Alperin Resnick, and Alexander Borgida. The CLASSIC knowledge representation system: Guiding principles and implementation rational. *SIGART Bulletin*, 2(3):108–113, 1991.

[34] Riccardo Rosati. Towards expressive KR systems integrating Datalog and description logics: Preliminary report. In *Proceedings of the 1999 Description Logic Workshop (DL'99)*, pages 160–164. CEUR Electronic Workshop Proceedings, `http://ceur-ws.org/Vol-22/`, 1999.

[35] Stephan Tobies. *Complexity Results and Practical Algorithms for Logics in Knowledge Representation.* PhD thesis, LuFG Theoretical Computer Science, RWTH-Aachen, Germany, 2001.

[36] John Yen, Robert Neches, and Robert MacGregor. CLASP: Integrating term subsumption systems and production systems. *IEEE Transactions on Knowledge and Data Engineering*, 3(1):25–31, 1991.