# Quantified equilibrium logic and hybrid rules — **Source link** ↗

Jos de Bruijn, David Pearce, Axel Polleres, Agustín Valverde

**Institutions:** King Juan Carlos University, National University of Ireland, Galway, University of Málaga

Related papers:

- DL + log : tight integration of description logics and disjunctive datalog

- On the decidability and complexity of integrating ontologies and rules

- Combining answer set programming with description logics for the Semantic Web

- A faithful integration of description logics with logic programming

- The stable model semantics for logic programming

# Quantified Equilibrium Logic and Hybrid Rules [*]

Jos de Bruijn[3], David Pearce[1], Axel Polleres[1,4], and Agustín Valverde[2]

[1] Universidad Rey Juan Carlos, Madrid, Spain
[2] Universidad de Málaga, Málaga, Spain
[3] DERI Innsbruck, Innsbruck, Austria
[4] DERI Galway, National University of Ireland, Galway
jos.debruijn@deri.org, davidandrew.pearce@urjc.es,
axel@polleres.net, a_valverde@ctima.uma.es

**Abstract.** In the ongoing discussion about combining rules and Ontologies on the Semantic Web a recurring issue is how to combine first-order classical logic with nonmonotonic rule languages. Whereas several modular approaches to define a combined semantics for such hybrid knowledge bases focus mainly on decidability issues, we tackle the matter from a more general point of view. In this paper we show how Quantified Equilibrium Logic (QEL) can function as a unified framework which embraces classical logic as well as disjunctive logic programs under the (open) answer set semantics. In the proposed variant of QEL we relax the unique names assumption, which was present in earlier versions of QEL. Moreover, we show that this framework elegantly captures the existing modular approaches for hybrid knowledge bases in a unified way.

## 1  Introduction

In the current discussions on the Semantic Web architecture a recurring issue is how to combine a first-order classical theory formalising an ontology with a nonmonotonic rule base. In this context, nonmonotonic rule languages have received considerable attention and achieved maturity over the last few years due to the success of Answer Set Programming (ASP), a nonmonotonic, purely declarative logic programming and knowledge representation paradigm with many useful features such as aggregates, weak constraints and priorities, supported by efficient implementations (for an overview see [1]). As a logical foundation for the answer set semantics and a tool for logical analysis in ASP, the system of Equilibrium Logic was presented in [14] and further developed in subsequent works (see [15] for an overview and references). We will show how Equilibrium Logic can be used as a logical foundation for the combination of ASP and Ontologies.

In the quest to provide a formal underpinning for a nonmonotonic rules layer for the Semantic Web which can coexist in a semantically well-defined manner with the Ontology layer, various proposals for combining classical first-order logic with different variants of ASP have been presented in the literature.[5] We distinguish three kinds

---

[5] Most of these approaches focus on the Description Logics fragments of first-order logic underlying the Web Ontology Language OWL.

of approaches: At the one end of the spectrum there are approaches which provide an entailment-based query interface to the Ontology in the bodies of ASP rules, resulting in a loose integration (e.g. [5, 4]). At the other end there are approaches which use a unifying nonmonotonic formalism to embed both the Ontology and the rule base (e.g. [2, 13]), resulting in a tight coupling. Hybrid approaches (e.g. [18–20, 9]) fall between these extremes. Common to hybrid approaches is the definition of a modular semantics based on classical first-order models, on the one hand, and stable models, on the other hand. Additionally, they require several syntactical restrictions on the use of classical predicates within rules. With further restrictions of the classical part to decidable Description Logics (DLs), these semantics support straightforward implementation using existing DL reasoners and ASP engines, in a modular fashion. In this paper, we focus on such hybrid approaches.

*Example 1.* Consider a hybrid knowledge base consisting of a classical theory $\mathcal{T}$:

$$\forall x.PERSON(x) \rightarrow (AGENT(x) \wedge (\exists y.HAS\text{-}MOTHER(x,y)))$$
$$\forall x.(\exists y.HAS\text{-}MOTHER(x,y)) \rightarrow ANIMAL(x)$$

which says that every $PERSON$ is an $AGENT$ and has some (unknown) mother, and everyone who has a mother is an $ANIMAL$, and a nonmonotonic logic program $\mathcal{P}$:

$$PERSON(x) \leftarrow AGENT(x), \neg machine(x)$$
$$AGENT(DaveB)$$

which says that $AGENT$s are by default $PERSON$s, unless *known* to be $machine$s, and $DaveB$ is an $AGENT$.

Using a hybrid knowledge base which includes both $\mathcal{T}$ and $\mathcal{P}$, we intuitively would conclude $PERSON(DaveB)$ since he is not known to be a $machine$, further that $DaveB$ has some (unknown) mother, and thus $ANIMAL(DaveB)$.

We see two important shortcomings in current hybrid approaches:

(1) Current approaches to hybrid knowledge bases differ not only in terms of syntactic restrictions, motivated by decidability considerations, but also in the way they deal with more fundamental issues which arise when classical logic meets ASP, such as the domain closure and unique names assumptions.[6] In particular, current proposals implicitly deal with these issues by either restricting the allowed models of the classical theory, or by using variants of the traditional answer set semantics which cater for open domains and non-unique names. So far, little effort has been spent in a comparing the approaches from a more general perspective.

(2) The semantics of current hybrid knowledge bases is defined in a modular fashion. This has the important advantage that algorithms for reasoning with this combination can be based on existing algorithms for DL and ASP satisfiability. A single underlying logic for hybrid knowledge bases which, for example, allows to capture notions of equivalence between combined knowledge bases in a standard way, is lacking though.

Our main contribution with this paper is twofold. First, we survey and compare different (extensions of the) answer set semantics, as well as the existing approaches to hybrid knowledge bases. Second, we propose to use Quantified Equilibrium Logic (QEL) as a unified logical foundation for hybrid knowledge bases: As it turns out, the

---

[6] See [3] for a more in-depth discussion of these issues.

equilibrium models of the combined knowledge base coincide exactly with the modular nonmonotonic models for all approaches we are aware of [18–20, 9].

The remainder of this paper is structured as follows: Section 2 recalls some basics of classical first-order logic. Section 3 reformulates different variants of the answer set semantics introduced in the literature using a common notation and points out correspondences and discrepancies between these variants. Next, definitions of hybrid knowledge bases from the literature are compared and generalised in Section 4. QEL and its relation to the different variants of ASP are clarified in Section 5. Section 6 describes an embedding of hybrid knowledge bases into QEL and establishes the correspondence between equilibrium models and nonmonotonic models of hybrid KBs. Implications of our results and further work are discussed in the concluding Sections 6.1, 6.2, and 7.

## 2   First-Order Logic (FOL)

A *function-free first-order language* $\mathcal{L} = \langle C, P \rangle$ with equality consists of disjoint sets of constant and predicate symbols $C$ and $P$. Moreover, we assume a fixed countably infinite set of variables, the symbols '$\rightarrow$', '$\vee$', '$\wedge$', '$\neg$', '$\exists$', '$\forall$', and auxiliary parentheses '$($',' $)$'. Each predicate symbol $p \in P$ has an assigned arity $ar(p)$. Atoms and formulas are constructed as usual. Closed formulas, or *sentences*, are those where each variable is bound by some quantifier. A *theory* $\mathcal{T}$ is a set of sentences. Variable-free atoms, formulas, or theories are also called *ground*. If $D$ is a non-empty set, we denote by $At_D(C, P)$ the set of ground atoms constructible from $\mathcal{L}' = \langle C \cup D, P \rangle$.

Given a first-order language $\mathcal{L}$, an $\mathcal{L}$-structure consists of a pair $\mathcal{I} = \langle U, I \rangle$, where the *universe* $U = (D, \sigma)$ (sometimes called pre-interpretation) consists of a non-empty domain $D$ and a function $\sigma \colon C \cup D \rightarrow D$ which assigns a domain value to each constant such that $\sigma(d) = d$ for every $d \in D$. For tuples we write $\sigma(\boldsymbol{t}) = (\sigma(d_1), \ldots, \sigma(d_n))$. We call $d \in D$ an *unnamed* individual if there is no $c \in C$ such that $\sigma(c) = d$. The function $I$ assigns a relation $p^I \subseteq D^n$ to each $n$-ary predicate symbol $p \in P$ and is called the $\mathcal{L}$-*interpretation over* $D$ . The designated binary predicate symbol $eq$, occasionally written '$=$' in infix notation, is assumed to be associated with the fixed interpretation function $eq^I = \{(d, d) \mid d \in D\}$. If $\mathcal{I}$ is an $\mathcal{L}'$-structure we denote by $\mathcal{I}|_{\mathcal{L}}$ the restriction of $\mathcal{I}$ to a sublanguage $\mathcal{L} \subseteq \mathcal{L}'$.

An $\mathcal{L}$-structure $\mathcal{I} = \langle U, I \rangle$ *satisfies* an atom $p(d_1, \ldots, d_n)$ of $At_D(C, P)$, written $\mathcal{I} \models p(d_1, \ldots, d_n)$, iff $(\sigma(d_1), \ldots, \sigma(d_n)) \in p^I$. This is extended as usual to sentences and theories. $\mathcal{I}$ is a *model* of an atom (sentence, theory, respectively) $\varphi$, written $\mathcal{I} \models \varphi$, if it satisfies $\varphi$. A theory $\mathcal{T}$ *entails* a sentence $\varphi$, written $\mathcal{T} \models \varphi$, if every model of $\mathcal{T}$ is also a model of $\varphi$. A theory is *consistent* if it has a model.

In the context of logic programs, the following assumptions often play a role: We say that the *parameter names assumption (PNA)* applies in case $\sigma$ is surjective, i.e., there are no unnamed individuals in $D$; the *unique names assumption (UNA)* applies in case $\sigma$ is injective; in case both the PNA and UNA apply, the *standard names assumption (SNA)* applies, i.e. $\sigma$ is a bijection. In the following, we will speak about PNA-, UNA-, or SNA-structures, (or PNA-, UNA-, or SNA-models, respectively), depending on $\sigma$.

An $\mathcal{L}$-interpretation $I$ over $D$ can be seen as a subset of $At_D(C, P)$. So, we can define a subset relation for $\mathcal{L}$-structures $\mathcal{I}_1 = \langle (D, \sigma_1), I_1 \rangle$ and $\mathcal{I}_2 = \langle (D, \sigma_2), I_2 \rangle$

over the same domain by setting $\mathcal{I}_1 \subseteq \mathcal{I}_2$ if $I_1 \subseteq I_2$.[7] Whenever we speak about subset minimality of models/structures in the following, we thus mean minimality among all models/structures over the same domain.

## 3    Answer Set Semantics

In this paper we assume non-ground disjunctive logic programs with negation allowed in rule heads and bodies, interpreted under the answer set semantics [12].[8] A program $\mathcal{P}$ consists of a set of rules of the form

$$a_1 \lor a_2 \lor \ldots \lor a_k \lor \neg a_{k+1} \lor \ldots \lor \neg a_l \leftarrow b_1, \ldots, b_m, \neg b_{m+1}, \ldots, \neg b_n \qquad (1)$$

where $a_i$ ($i \in \{1, \ldots, l\}$) and $b_j$ ($j \in \{1, \ldots, n\}$) are atoms, called head (body, respectively) atoms of the rule, in a function-free first-order language $\mathcal{L} = \langle C, P \rangle$ without equality. By $C_\mathcal{P} \subseteq C$ we denote the set of constants which appear in $\mathcal{P}$. A rule with $k = l$ and $m = n$ is called *positive*. Rules where each variable appears in $b_1, \ldots, b_m$ are called *safe*. A program is *positive* (*safe*) if all its rules are positive (safe).

For the purposes of this paper, we give a slightly generalised definition of the common notion of the *grounding* of a program: The *grounding* $gr_U(\mathcal{P})$ of $\mathcal{P}$ wrt. a universe $U = (D, \sigma)$ denotes the set of all rules obtained as follows: For $r \in \mathcal{P}$, replace (i) each constant $c$ appearing in $r$ with $\sigma(c)$ and (ii) each variable with some element in $D$. Observe that thus $gr_U(\mathcal{P})$ is a ground program over the atoms in $At_D(C, P)$.

For a ground program $\mathcal{P}$ and first-order structure $\mathcal{I}$ the *reduct* $\mathcal{P}^\mathcal{I}$ consists of rules

$$a_1 \lor a_2 \lor \ldots \lor a_k \leftarrow b_1, \ldots, b_m$$

obtained from all rules of the form (1) in $\mathcal{P}$ for which hold that $\mathcal{I} \models a_i$ for all $k < i \le l$ and $\mathcal{I} \not\models b_j$ for all $m < j \le n$.

Answer set semantics is usually defined in terms of *Herbrand structures* over $\mathcal{L} = \langle C, P \rangle$. Herbrand structures have a fixed universe, the *Herbrand universe* $\mathcal{H} = (C, id)$, where $id$ is the identity function. For a Herbrand structure $\mathcal{I} = \langle \mathcal{H}, I \rangle$, $I$ can be viewed as a subset of the *Herbrand base*, $\mathcal{B}$, which consists of the ground atoms of $\mathcal{L}$. Note that by definition of $\mathcal{H}$, Herbrand structures are SNA-structures. A Herbrand structure $\mathcal{I}$ is an *answer set* [12] of $\mathcal{P}$ if $\mathcal{I}$ is subset minimal among the structures satisfying $gr_\mathcal{H}(\mathcal{P})^\mathcal{I}$. Two variations of this semantics, the open [8] and generalised open answer set [9] semantics, consider open domains, thereby relaxing the PNA. An *extended Herbrand structure* is a first-order structure based on a universe $U = (D, id)$, where $D \supseteq C$.

**Definition 1.** *A first-order $\mathcal{L}$-structure $\mathcal{I} = \langle U, I \rangle$ is called a* generalised open answer set *of $\mathcal{P}$ if $\mathcal{I}$ is subset minimal among the structures satisfying all rules in $gr_U(\mathcal{P})^\mathcal{I}$. If, additionally, $\mathcal{I}$ is an extended Herbrand structure, then $\mathcal{I}$ is an* open answer set *of $\mathcal{P}$.*

In the open answer set semantics the UNA applies. Note that every answer set of a program is also an open answer set [8], but the converse does not hold in general:

---

[7] Note that this is not the substructure or submodel relation in classical model theory, which holds between a structure and its restriction to a subdomain.

[8] By $\neg$ we mean negation as failure and not classical, or strong negation, which is also sometimes considered in ASP.

*Example 2.* Consider $\mathcal{P} = \{p(a);\ ok \leftarrow \neg p(x);\ \leftarrow \neg ok\}$ over $\mathcal{L} = \langle\{a\}, \{p, ok\}\rangle$. We leave it as an exercise to the reader to show that $\mathcal{P}$ is inconsistent under the answer set semantics, but $\mathcal{M} = \langle(\{a, c_1\}, id), \{p(a), ok\}\rangle$ is an open answer set of $\mathcal{P}$.

An alternative approach to relax the UNA has been presented by Rosati in [19]: Instead of grounding with respect to $U$, programs are grounded with respect to the Herbrand universe $\mathcal{H} = (C, id)$, and minimality of the models of $gr_{\mathcal{H}}(\mathcal{P})^{\mathcal{I}}$ wrt. $U$ is redefined: $\mathcal{I}{\upharpoonright}_{\mathcal{H}} = \{p(\sigma(c_1), \ldots, \sigma(c_n)) \mid p(c_1, \ldots, c_n) \in \mathcal{B}, \mathcal{I} \models p(c_1, \ldots, c_n)\}$, i.e., $\mathcal{I}{\upharpoonright}_{\mathcal{H}}$ is the restriction of $\mathcal{I}$ to ground atoms of $\mathcal{B}$. Given $\mathcal{L}$-structures $\mathcal{I}_1 = (U_1, I_1)$ and $\mathcal{I}_2 = (U_2, I_2)$, the relation $\mathcal{I}_1 \subseteq_{\mathcal{H}} \mathcal{I}_2$ holds if $\mathcal{I}_1{\upharpoonright}_{\mathcal{H}} \subseteq \mathcal{I}_2{\upharpoonright}_{\mathcal{H}}$.

**Definition 2.** *An $\mathcal{L}$-structure $\mathcal{I}$ is called a* generalised answer set *of $\mathcal{P}$ if $\mathcal{I}$ is $\subseteq_{\mathcal{H}}$-minimal among the structures satisfying all rules in $gr_{\mathcal{H}}(\mathcal{P})^{\mathcal{I}}$.*

The following Lemma establishes that, for safe programs, all atoms of $At_D(C, P)$ satisfied in an open answer set of a safe program are ground atoms over $C_{\mathcal{P}}$:

**Lemma 1.** *Let $\mathcal{P}$ be a safe program over $\mathcal{L} = \langle C, P \rangle$ with $\mathcal{M} = \langle U, I \rangle$ a (generalised) open answer set over universe $U = (D, \sigma)$. Then, for any atom from $At_D(C, P)$ such that $\mathcal{M} \models p(d_1, \ldots, d_n)$, there exist $c_i \in C_{\mathcal{P}}$ such that $\sigma(c_i) = d_i$ for each $1 \leq i \leq n$.*

From this Lemma, the following correspondence follows directly.

**Proposition 1.** *$\mathcal{M}$ is an (generalised) answer set of a* safe *program $\mathcal{P}$ if and only if $\mathcal{M}$ is an (generalised) open answer set of $\mathcal{P}$.*

If the SNA applies, consistency with respect to all semantics introduced so far boils down to consistency under the original definition of answer sets:

**Proposition 2.** *A program $\mathcal{P}$ has an answer set if and only if $\mathcal{P}$ has a generalised open answer under the SNA.*

Answer sets under SNA may differ from the original answer sets since also non-Herbrand structures are allowed. Further, we observe that there are programs which have generalised (open) answer sets but do not have (open) answer sets, even for safe programs:

*Example 3.* Consider $\mathcal{P} = \{p(a);\ \leftarrow \neg p(b)\}$ over $\mathcal{L} = \langle\{a, b\}, \{p\}\rangle$. $\mathcal{P}$ is ground, thus obviously safe. However, although $\mathcal{P}$ has a generalised (open) answer set – the reader may verify this by, for instance, considering the one-element universe $U = (\{d\}, \sigma)$, where $\sigma(a) = \sigma(b) = d$ – it is inconsistent under the open answer set semantics.

## 4 Hybrid Knowledge Bases

We now turn to the concept of hybrid knowledge bases, which combine classical theories with the various notions of answer sets. We define a notion of hybrid knowledge bases which generalizes definitions in the literature [18–20, 9]. We then compare and discuss the differences between the various definitions. It turns out that the differences are mainly concerned with the notion of answer sets, and syntactical restrictions, but do not change the general semantics. This will allow us to base our embedding into Quantified Equilibrium Logic on a unified definition.

A *hybrid knowledge base* $\mathcal{K} = (\mathcal{T}, \mathcal{P})$ over the function-free language $\mathcal{L} = \langle C, P_{\mathcal{T}} \cup P_{\mathcal{P}} \rangle$ consists of a classical first-order theory $\mathcal{T}$ (also called the *structural* part of $\mathcal{K}$) over the language $\mathcal{L}_{\mathcal{T}} = \langle C, P_{\mathcal{T}} \rangle$ and a program $\mathcal{P}$ (also called *rules* part of $\mathcal{K}$) over the language $\mathcal{L}$, where $P_{\mathcal{T}} \cap P_{\mathcal{P}} = \emptyset$, i.e. $\mathcal{T}$ and $\mathcal{P}$ share a single set of constants, and the predicate symbols allowed to be used in $\mathcal{P}$ are a superset of the predicate symbols in $\mathcal{L}_{\mathcal{T}}$. Intuitively, the predicates in $\mathcal{L}_{\mathcal{T}}$ are interpreted classically, whereas the predicates in $\mathcal{L}_{\mathcal{P}}$ are interpreted nonmonotonically under the (generalised open) answer set semantics. With $\mathcal{L}_{\mathcal{P}} = \langle C, P_{\mathcal{P}} \rangle$ we denote the restricted language of $\mathcal{P}$.

We define the *projection* of a ground program $\mathcal{P}$ with respect to an $\mathcal{L}$-structure $\mathcal{I} = \langle U, I \rangle$, denoted $\Pi(\mathcal{P}, \mathcal{I})$, as follows: for each rule $r \in \mathcal{P}$, $r^{\Pi}$ is defined as:

1. $r^{\Pi} = \emptyset$ if there is a literal over $At_D(C, P_T)$ in the head of $r$ of form $p(\boldsymbol{t})$ such that $p(\sigma(\boldsymbol{t})) \in I$ or of form $\neg p(\boldsymbol{t})$ with $p(\sigma(\boldsymbol{t})) \notin I$;
2. $r^{\Pi} = \emptyset$ if there is a literal over $At_D(C, P_T)$ in the body of $r$ of form $p(\boldsymbol{t})$ such that $p(\sigma(\boldsymbol{t})) \notin I$ or of form $\neg p(\boldsymbol{t})$ such that $p(\sigma(\boldsymbol{t})) \in I$;
3. otherwise $r^{\Pi}$ is the singleton set resulting from $r$ by deleting all occurrences of literals from $\mathcal{L}_{\mathcal{T}}$,

and $\Pi(\mathcal{P}, \mathcal{I}) = \bigcup \{r^{\Pi} : r \in \mathcal{P}\}$. Intuitively, the projection "evaluates" all classical literals in $\mathcal{P}$ with respect to $\mathcal{I}$.

**Definition 3.** *Let $\mathcal{K} = (\mathcal{T}, \mathcal{P})$ be a hybrid knowledge base over the language $\mathcal{L} = \langle C, P_{\mathcal{T}} \cup P_{\mathcal{P}} \rangle$. An NM-model $\mathcal{M} = \langle U, I \rangle$ (with $U = (D, \sigma)$) of $\mathcal{K}$ is a first-order $\mathcal{L}$-structure such that $\mathcal{M}|_{\mathcal{L}_{\mathcal{T}}}$ is a model of $\mathcal{T}$ and $\mathcal{M}|_{\mathcal{L}_{\mathcal{P}}}$ is a generalised open answer set of $\Pi(gr_U(\mathcal{P}), \mathcal{M})$.*

Analogous to first-order models, we speak about PNA-, UNA-, and SNA-NM-models.

*Example 4.* Consider the hybrid knowledge base $\mathcal{K} = (\mathcal{T}, \mathcal{P})$, with $\mathcal{T}$ and $\mathcal{P}$ as in Example 1, with the capitalised predicates being predicates in $P_{\mathcal{T}}$. Now consider the interpretation $\mathcal{I} = \langle U, I \rangle$ (with $U = (D, \sigma)$) with $D = \{DaveB, k\}$, $\sigma$ the identity function, and $I = \{AGENT(DaveB), HAS\text{-}MOTHER(DaveB, k), ANIMAL(DaveB), machine(DaveB)\}$. Clearly, $\mathcal{I}|_{\mathcal{L}_{\mathcal{T}}}$ is a model of $\mathcal{T}$. The projection $\Pi(gr_U(\mathcal{P}), \mathcal{I})$ is

$$\leftarrow \neg machine(DaveB),$$

which does not have a stable model, and thus $\mathcal{I}$ is not an NM-model of $\mathcal{K}$. In fact, the logic program $\mathcal{P}$ ensures that an interpretation cannot be an NM-model of $\mathcal{K}$ if there is an $AGENT$ which is neither a $PERSON$ nor known (by conclusions from $\mathcal{P}$) to be a $machine$. It is easy to verify that, for any NM-model of $\mathcal{K}$, the atoms $AGENT(DaveB), PERSON(DaveB)$, and $ANIMAL(DaveB)$ must be true, and are thus entailed by $\mathcal{K}$. The latter cannot be derived from $\mathcal{T}$ or $\mathcal{P}$ individually.

We now proceed to compare our definition of NM-models with the various definitions in the literature. The first kind of hybrid knowledge base we consider was introduced by Rosati in [18] (and extended in [20] under the name $\mathcal{DL}+log$), and was labeled *r-hybrid* knowledge base. Syntactically, *r-hybrid* KBs do not allow negated atoms in rule heads, i.e. for rules of the form (1) $l = k$, and do not allow atoms from

$\mathcal{L}_{\mathcal{T}}$ to occur negatively in the rule body.[9] Moreover, in [18], Rosati deploys a restriction which is stronger than standard safety: each variable must appear in at least one positive body atom with a predicate from $\mathcal{L}_{\mathcal{P}}$. We call this condition $\mathcal{L}_{\mathcal{P}}$-*safe* in the remainder. In [20] this condition is relaxed to *weak $\mathcal{L}_{\mathcal{P}}$-safety*: there is no special safety restriction for variables which occur only in body atoms from $P_{\mathcal{T}}$.

**Definition 4.** *Let $\mathcal{K} = (\mathcal{T}, \mathcal{P})$ be an r-hybrid knowledge base, over the language $\mathcal{L} = \langle C, P_{\mathcal{T}} \cup P_{\mathcal{P}} \rangle$, where $C$ is countably infinite, and $\mathcal{P}$ is a (weak) $\mathcal{L}_{\mathcal{P}}$-safe program. An r-NM-model $\mathcal{M} = \langle U, I \rangle$ of $\mathcal{K}$ is a first-order $\mathcal{L}$-SNA-structure such that $\mathcal{M}|_{\mathcal{L}_{\mathcal{T}}}$ is a model of $\mathcal{T}$ and $\mathcal{M}|_{\mathcal{L}_{\mathcal{P}}}$ is an answer set of $\Pi(gr_U(\mathcal{P}), \mathcal{M})$.*

In view of the (weak) $\mathcal{L}_{\mathcal{P}}$-safety condition, we observe that r-NM-model existence coincides with SNA-NM-model existence on r-hybrid knowledge bases, by Lemma 1 and Proposition 2. In [19], Rosati relaxes the UNA for what we will call here *$r^+$-hybrid* knowledge bases.

**Definition 5.** *Let $\mathcal{K} = (\mathcal{T}, \mathcal{P})$ be an $r^+$-hybrid knowledge base consisting of a theory $\mathcal{T}$ and an $\mathcal{L}_{\mathcal{P}}$-safe program $\mathcal{P}$. An $r^+$-NM-model, $\mathcal{M} = \langle U, I \rangle$ of $\mathcal{K}$ is a first-order $\mathcal{L}$-structure such that $\mathcal{M}|_{\mathcal{L}_{\mathcal{T}}}$ is a model of $\mathcal{T}$ and $\mathcal{M}|_{\mathcal{L}_{\mathcal{P}}}$ is a generalised answer set of $\Pi(gr_U(\mathcal{P}), \mathcal{M})$.*

$\mathcal{L}_{\mathcal{P}}$-safety guarantees safety of $\Pi(gr_U(\mathcal{P}), \mathcal{M})$. Thus, by Proposition 1, we can conclude that $r^+$-NM-models coincide with NM-models on r-hybrid knowledge bases.

*G-hybrid knowledge bases* [9] allow a different form of rules in the program. In order to regain decidability, rules are not required to be safe, but they are required to be *guarded* (hence the 'g' in g-hybrid): All variables in a rule are required to occur in a single positive body atom, the *guard*, with the exception of unsafe choice rules of the form

$$p(c_1, \ldots, c_n) \vee \neg p(c_1, \ldots, c_n) \leftarrow$$

are allowed. Moreover, disjunction in rule heads is limited to at most one positive atom, i.e. for rules of the form (1) we have that $k \leq 1$, but an arbitrary number of negated head atoms is allowed. The definition of NM-models in [9] coincides precisely with our Definition 3.

## 5 Quantified Equilibrium Logic (QEL)

Equilibrium logic for propositional theories and logic programs was presented in [14] as a foundation for answer set semantics, and extended to the first-order case in [16], as well as, in slightly more general, modified form, in [17]. For a survey of the main properties of equilibrium logic, see [15]. Usually in quantified equilibrium logic we consider a full first-order language allowing function symbols and we include a second, strong negation operator as occurs in several ASP dialects. For the present purpose of drawing comparisons with approaches to hybrid knowledge bases, it will suffice to

---

[9] Note that by projection, negation of predicates from $P_{\mathcal{T}}$ is treated classically, whereas negation of predicates from $P_{\mathcal{P}}$ is treated nonmonotonically. The negative occurrence of classical predicates in the body is equivalent to the positive occurrence of the predicate in the head.

consider the function-free language with a single negation symbol, '¬'. In particular, we shall work with a quantified version of the logic HT of *here-and-there*. In other respects we follow the treatment of [17].

## 5.1 General Structures for Quantified Here-and-There Logic

As before, we consider a function-free first order languages $\mathcal{L} = \langle C, P \rangle$ built over a set of *constant* symbols, $C$, and a set of *predicate* symbols, $P$. The sets of $\mathcal{L}$-formulas, $\mathcal{L}$-sentences and atomic $\mathcal{L}$-sentences are defined in the usual way.

Again, we only work with *sentences*, and, as in Section 2, by an $\mathcal{L}$-interpretation $I$ over a set $D$ we mean a subset $I$ of $At_D(C, P)$. A *here-and-there* $\mathcal{L}$-structure with static domains, or $\mathbf{QHT}^s(\mathcal{L})$-*structure*, is a tuple $\mathcal{M} = \langle (D, \sigma), I_h, I_t \rangle$ where
 – $D$ is a non-empty set, called the *domain* of $\mathcal{M}$.
 – $\sigma$ is a mapping: $C \cup D \to D$ called the *assignment* such that $\sigma(d) = d$ for all $d \in D$. If $D = C$ and $\sigma = id$, $\mathcal{M}$ is a *Herbrand structure*.
 – $I_h, I_t$ are $\mathcal{L}$-interpretations over $D$ such that $I_h \subseteq I_t$.
We can think of $\mathcal{M}$ as a structure similar to a first-order classical model, but having two parts, or components, $h$ and $t$ that correspond to two different points or "worlds", 'here' and 'there', in the sense of Kripke semantics for intuitionistic logic [22], where the worlds are ordered by $h \leq t$. At each world $w \in \{h, t\}$ one verifies a set of atoms $I_w$ in the expanded language for the domain $D$. We call the model static, since, in contrast to say intuitionistic logic, the same domain serves each of the worlds.[10] Since $h \leq t$, whatever is verified at $h$ remains true at $t$. The satisfaction relation for $\mathcal{M}$ is defined so as to reflect the two different components, so we write $\mathcal{M}, w \models \varphi$ to denote that $\varphi$ is true in $\mathcal{M}$ with respect to the $w$ component. Evidently we should require that an atomic sentence is true at $w$ just in case it belongs to the $w$-interpretation. Formally, if $p(t_1, \ldots, t_n) \in \mathrm{At}_D$ then

$$\mathcal{M}, w \models p(t_1, \ldots, t_n) \quad \text{iff} \quad p(\sigma(t_1), \ldots, \sigma(t_n)) \in I_w. \tag{2}$$

Then $\models$ is extended recursively as follows[11]:

 – $\mathcal{M}, w \models \varphi \wedge \psi$ iff $\mathcal{M}, w \models \varphi$ and $\mathcal{M}, w \models \psi$.
 – $\mathcal{M}, w \models \varphi \vee \psi$ iff $\mathcal{M}, w \models \varphi$ or $\mathcal{M}, w \models \psi$.
 – $\mathcal{M}, t \models \varphi \to \psi$ iff $\mathcal{M}, t \not\models \varphi$ or $\mathcal{M}, t \models \psi$.
 – $\mathcal{M}, h \models \varphi \to \psi$ iff $\mathcal{M}, t \models \varphi \to \psi$ and $\mathcal{M}, h \not\models \varphi$ or $\mathcal{M}, h \models \psi$.
 – $\mathcal{M}, w \models \neg \varphi$ iff $\mathcal{M}, t \not\models \varphi$.
 – $\mathcal{M}, t \models \forall x \varphi(x)$ iff $\mathcal{M}, t \models \varphi(d)$ for all $d \in D$.
 – $\mathcal{M}, h \models \forall x \varphi(x)$ iff $\mathcal{M}, t \models \forall x \varphi(x)$ and $\mathcal{M}, h \models \varphi(d)$ for all $d \in D$.
 – $\mathcal{M}, w \models \exists x \varphi(x)$ iff $\mathcal{M}, w \models \varphi(d)$ for some $d \in D$.

Truth of a sentence in a model is defined as follows: $\mathcal{M} \models \varphi$ iff $\mathcal{M}, w \models \varphi$ for each $w \in \{h, t\}$. A sentence $\varphi$ is valid if it is true in all models, denoted by $\models \varphi$. A sentence $\varphi$ is a consequence of a set of sentences $\Gamma$, denoted $\Gamma \models \varphi$, if every model

---

[10] Alternatively it is quite common to speak of a logic with *constant* domains.

[11] The reader may easily check that the following correspond exactly to the usual Kripke semantics for intuitionistic logic given our assumptions about the two worlds $h$ and $t$ and the single domain $D$, see e.g. [22]

of $\Gamma$ is a model of $\varphi$. In a model $\mathcal{M}$ we often use the symbols $H$ and $T$, possibly with subscripts, to denote the interpretations $I_h$ and $I_t$ respectively; so, an $\mathcal{L}$-structure may be written in the form $\langle U, H, T \rangle$, where $U = (D, \sigma)$.

The resulting logic is called *Quantified Here-and-There Logic with static domains*, denoted by $\mathbf{QHT}^s$, and can be axiomatised as follows.

Let $\mathbf{INT}^=$ denote first-order intuitionistic logic [22] with the usual axioms for equality:

$$x = x,$$
$$x = y \rightarrow (F(x) \rightarrow F(y))$$

for every formula $F(x)$ such that $y$ is substitutable for $x$ in $F(x)$. To this we add the axiom of Hosoi

$$\alpha \vee (\neg \beta \vee (\alpha \rightarrow \beta))$$

which determines 2-element here-and-there models in the propositional case, and the axiom:

SQHT  $\exists x (F(x) \rightarrow \forall x F(x))$.

The notation SQHT stands for "static quantified here-and-there". Lastly we add the "decidable equality" axiom

DE  $x = y \vee x \neq y$.

For a completeness proof for $\mathbf{QHT}^s$, see [11].

As usual in first order logic, satisfiability and validity are independent from the language. If $\mathcal{M} = \langle (D, \sigma), H, T \rangle$ is an $\mathbf{QHT}^s(\mathcal{L}')$-structure and $\mathcal{L} \subset \mathcal{L}'$, we denote by $\mathcal{M}|_{\mathcal{L}}$ the restriction of $\mathcal{M}$ to the sublanguage $\mathcal{L}$: $\mathcal{M}|_{\mathcal{L}} = \langle (D, \sigma|_{\mathcal{L}}), H|_{\mathcal{L}}, T|_{\mathcal{L}} \rangle$.

**Proposition 3.** *Suppose that $\mathcal{L}' \supset \mathcal{L}$, $\Gamma$ is a theory in $\mathcal{L}$ and $\mathcal{M}$ is an $\mathcal{L}'$-structure such $\mathcal{M} \models \Gamma$. Then $\mathcal{M}|_{\mathcal{L}}$ is a model of $\Gamma$ in $\mathbf{QHT}^s_=(\mathcal{L})$.*

**Proposition 4.** *Suppose that $\mathcal{L}' \supset \mathcal{L}$ and $\varphi \in \mathcal{L}$. Then $\varphi$ is valid (resp. satisfiable) in $\mathbf{QHT}^s_=(\mathcal{L})$ if and only if is valid (resp. satisfiable) in $\mathbf{QHT}^s_=(\mathcal{L}')$.*

Analogous to the case of classical models we can define special kinds of $\mathbf{QHT}^s$ (resp. $\mathbf{QHT}^s_=$) models. Let $\mathcal{M} = \langle (D, \sigma), H, T \rangle$ be an $\mathcal{L}$-structure that is a model of a universal theory $T$. Then, we call $\mathcal{M}$ a PNA-, UNA-, or SNA-model if the restriction of $\sigma$ to constants in $\mathcal{C}$ is surjective, injective or bijective, respectively.

### 5.2 Equilibrium Models and their Relation to Answer Sets

As in the propositional case, quantified equilibrium logic is based on a suitable notion of minimal model.

**Definition 6.** *Among $\mathbf{QHT}^s_=(\mathcal{L})$-structures we define the order $\trianglelefteq$ as: $\langle (D, \sigma), H, T \rangle \trianglelefteq \langle (D', \sigma'), H', T' \rangle$ if $D = D'$, $\sigma = \sigma'$, $T = T'$ and $H \subseteq H'$. If the subset relation is strict, we write '$\triangleleft$'.*

**Definition 7.** *Let $\Gamma$ be a set of sentences and $\mathcal{M} = \langle (D, \sigma), H, T \rangle$ a model of $\Gamma$.*
*1. $\mathcal{M}$ is said to be* total *if $H = T$.*
*2. $\mathcal{M}$ is said to be an* equilibrium *model of $\Gamma$ (or short, we say: "$\mathcal{M}$ is in equilibrium") if it is minimal under $\trianglelefteq$ among models of $\Gamma$, and it is total.*

Notice that a total $\mathbf{QHT}^s_{\underline{=}}$ model of a theory $\Gamma$ is equivalent to a classical first order model of $\Gamma$.

**Proposition 5.** *Let $\Gamma$ be a theory in $\mathcal{L}$ and $\mathcal{M}$ an equilibrium model of $\Gamma$ in $\mathbf{QHT}^s_{\underline{=}}(\mathcal{L}')$ with $\mathcal{L}' \supset \mathcal{L}$. Then $\mathcal{M}|_{\mathcal{L}}$ is an equilibrium model of $\Gamma$ in $\mathbf{QHT}^s_{\underline{=}}(\mathcal{L})$.*

The above version of QEL is described in more detail in [17]. If we assume all models are UNA-models, we obtain the version of QEL found in [16]. There, the relation of QEL to (ordinary) answer sets for logic programs with variables was established (in [16, Corollary 7.7]). For the present version of QEL the correspondence can be described as follows.

**Proposition 6 ([17]).** *Let $\Gamma$ be a universal theory in $\mathcal{L} = \langle C, P \rangle$. Let $\langle U, T, T \rangle$ be a total $\mathbf{QHT}^s_{\underline{=}}$ model of $\Gamma$. Then $\langle U, T, T \rangle$ is an equilibrium model of $\Gamma$ iff $\langle T, T \rangle$ is a propositional equilibrium model of $gr_U(\Gamma)$.*

By convention, when $\mathcal{P}$ is a logic program with variables we consider the models and equilibrium models of its universal closure expressed as a set of logical formulas. So, from Proposition 6 we obtain:

**Corollary 1.** *Let $\mathcal{P}$ be a logic program. A total $\mathbf{QHT}^s_{\underline{=}}$ model $\langle U, T, T \rangle$ of $\mathcal{P}$ is an equilibrium model of $\mathcal{P}$ iff it is a generalised open answer set of $\mathcal{P}$.*

## 6 Relation between Hybrid KBs and QEL

In this section we show how equilibrium models for hybrid knowledge bases relate to the NM models defined earlier and we show that QEL captures the various approaches to the semantics of hybrid KBs in the literature [18–20, 9].

Given a hybrid KB $\mathcal{K} = (\mathcal{T}, \mathcal{P})$ we call $\mathcal{T} \cup \mathcal{P} \cup st(\mathcal{T})$ the *stable closure* of $\mathcal{K}$, where $st(\mathcal{T}) = \{\forall x(p(x) \vee \neg p(x)) : p \in \mathcal{L}_{\mathcal{T}}\}$.[12] From now on, unless otherwise clear from context, the symbol '$\models$' denotes the truth relation for $\mathbf{QHT}^s_{\underline{=}}$. Given a ground program $\mathcal{P}$ and an $\mathcal{L}$-structure $\mathcal{M} = \langle U, H, T \rangle$, the *projection* $\Pi(\mathcal{P}, \mathcal{M})$ is understood to be defined relative to the component $T$ of $\mathcal{M}$.

**Lemma 2.** *Let $\mathcal{M} = \langle U, H, T \rangle$ be a $\mathbf{QHT}^s_{\underline{=}}$-model of $\mathcal{T} \cup st(\mathcal{T})$. Then $\mathcal{M} \models \mathcal{P}$ iff $\mathcal{M}|_{\mathcal{L}_{\mathcal{P}}} \models \Pi(gr_U(\mathcal{P}), \mathcal{M})$.*

*Proof.* By the hypothesis $\mathcal{M} \models \{\forall x(p(x) \vee \neg p(x)) : p \in \mathcal{L}_{\mathcal{T}}\}$. It follows that $H|_{\mathcal{L}_{\mathcal{T}}} = T|_{\mathcal{L}_{\mathcal{T}}}$. Consider any $r \in \mathcal{P}$, such that $r^\Pi \neq \emptyset$. Then there are four cases to consider. (i) $r$ has the form $\alpha \rightarrow \beta \vee p(t)$, $p(t) \in \mathcal{L}_{\mathcal{T}}$ and $p(\sigma(t)) \notin T$, so $\mathcal{M} \models \neg p(t)$. W.l.o.g. assume that $\alpha, \beta \in \mathcal{L}_{\mathcal{P}}$, so $r^\Pi = \alpha \rightarrow \beta$ and

$$\mathcal{M} \models r \Leftrightarrow \mathcal{M} \models r^\Pi \Leftrightarrow \mathcal{M}|_{\mathcal{L}_{\mathcal{P}}} \models r^\Pi \tag{3}$$

by the semantics for $\mathbf{QHT}^s_{\underline{=}}$ and Theorem 3. (ii) $r$ has the form $\alpha \rightarrow \beta \vee \neg p(t)$, where $p(\sigma(t)) \in T$; so $p(\sigma(t)) \in H$ and $\mathcal{M} \models p(t)$. Again it is easy to see that (3) holds. Case (iii): $r$ has the form $\alpha \wedge p(t) \rightarrow \beta$ and $p(\sigma(t)) \in H, T$, so $\mathcal{M} \models p(t)$. Case (iv):

---

[12] Evidently $\mathcal{T}$ becomes *stable* in $\mathcal{K}$ in the sense that $\forall \varphi \in \mathcal{T}, st(\mathcal{T}) \models \neg\neg\varphi \rightarrow \varphi$. The terminology is drawn from intuitionistic logic and mathematics.

$r$ has the form $\alpha \wedge \neg p(t) \to \beta$ and $\mathcal{M} \models \neg p(t)$. Clearly for these two cases (3) holds as well. It follows that if $\mathcal{M} \models \mathcal{P}$ then $\mathcal{M}|_{\mathcal{L}_{\mathcal{P}}} \models \Pi(gr_U(\mathcal{P}), \mathcal{M})$.

To check the converse condition we need now only examine the cases where $r^\Pi = \emptyset$. Suppose this arises because $p(\sigma(t)) \in H, T$, so $\mathcal{M} \models p(t)$. Now, if $p(t)$ is in the head of $r$, clearly $\mathcal{M} \models r$. Similarly if $\neg p(t)$ is in the body of $r$, by the semantics $\mathcal{M} \models r$. The cases where $p(\sigma(t)) \notin T$ are analogous and left to the reader. Consequently if $\mathcal{M}|_{\mathcal{L}_{\mathcal{P}}} \models \Pi(gr_U(\mathcal{P}), \mathcal{M})$, then $\mathcal{M} \models \mathcal{P}$. $\qquad\square$

We now state the relation between equilibrium models and NM-models.

**Theorem 1.** *Let $\mathcal{K} = (\mathcal{T}, \mathcal{P})$ be a hybrid knowledge base. Let $\mathcal{M} = \langle U, T, T \rangle$ be a total here-and-there model of the stable closure of $\mathcal{K}$. Then $\mathcal{M}$ is in equilibrium if and only if $\langle U, T \rangle$ is an NM-model of $\mathcal{K}$.*

*Proof.* Assume the hypothesis and suppose that $\mathcal{M}$ is in equilibrium. Since $\mathcal{T}$ contains only predicates from $\mathcal{L}_{\mathcal{T}}$ and $\mathcal{M} \models \mathcal{T} \cup st(\mathcal{T})$, evidently

$$\mathcal{M}|_{\mathcal{L}_{\mathcal{T}}} \models \mathcal{T} \cup st(\mathcal{T}) \tag{4}$$

and so in particular $(U, \mathcal{M}|_{\mathcal{L}_{\mathcal{T}}})$ is a model of $\mathcal{T}$. By Lemma 2,

$$\mathcal{M} \models \mathcal{P} \Leftrightarrow \mathcal{M}|_{\mathcal{L}_{\mathcal{P}}} \models \Pi(gr_U(\mathcal{P}), \mathcal{M}). \tag{5}$$

We claim (i) that $\mathcal{M}|_{\mathcal{L}_{\mathcal{P}}}$ is an equilibrium model of $\Pi(gr_U(\mathcal{P}), \mathcal{M})$. If not, there is a model $\mathcal{M}' = \langle H', T' \rangle$ with $H' \subset T' = T|_{\mathcal{L}_{\mathcal{P}}}$ and $\mathcal{M}' \models \Pi(gr_U(\mathcal{P}), \mathcal{M})$. Lift $(U, \mathcal{M}')$ to a (first order) $\mathcal{L}$-structure $\mathcal{N}$ by interpreting each $p \in \mathcal{L}_{\mathcal{T}}$ according to $\mathcal{M}$. So $\mathcal{N}|_{\mathcal{L}_{\mathcal{T}}} = \mathcal{M}|_{\mathcal{L}_{\mathcal{T}}}$ and by (4) clearly $\mathcal{N} \models \mathcal{T} \cup st(\mathcal{T})$. Moreover, by Lemma 2 $\mathcal{N} \models \mathcal{P}$ and by assumption $\mathcal{N} \lhd \mathcal{M}$, contradicting the assumption that $\mathcal{M}$ is an equilibrium model of $\mathcal{T} \cup st(\mathcal{T}) \cup \mathcal{P}$. This establishes (i). Lastly, we note that since $\langle T|_{\mathcal{L}_{\mathcal{P}}}, T|_{\mathcal{L}_{\mathcal{P}}} \rangle$ is an equilibrium model of $\Pi(gr_U(\mathcal{P}), \mathcal{M})$, $\mathcal{M}|_{\mathcal{L}_{\mathcal{P}}}$ is a generalised open answer set of $\Pi(gr_U(\mathcal{P}), \mathcal{M})$ by Corollary 1, so that $\mathcal{M} = \langle U, T, T \rangle$ is an NM-model of $\mathcal{K}$.

For the converse direction, assume the hypothesis but suppose that $\mathcal{M}$ is not in equilibrium. Then there is a model $\mathcal{M}' = \langle U, H, T \rangle$ of $\mathcal{T} \cup st(\mathcal{T}) \cup \mathcal{P}$, with $H \subset T$. Since $\mathcal{M}' \models \mathcal{P}$ we can apply Lemma 2 to conclude that $\mathcal{M}'|_{\mathcal{L}_{\mathcal{P}}} \models \Pi(gr_U(\mathcal{P}), \mathcal{M}')$. But clearly

$$\Pi(gr_U(\mathcal{P}), \mathcal{M}') = \Pi(gr_U(\mathcal{P}), \mathcal{M}).$$

However, since evidently $\mathcal{M}'|_{\mathcal{L}_{\mathcal{T}}} = \mathcal{M}|_{\mathcal{L}_{\mathcal{T}}}$, thus $\mathcal{M}'|_{\mathcal{L}_{\mathcal{P}}} \lhd \mathcal{M}|_{\mathcal{L}_{\mathcal{P}}}$, so this shows that $\mathcal{M}|_{\mathcal{L}_{\mathcal{P}}}$ is not an equilibrium model of $\Pi(gr_U(\mathcal{P}), \mathcal{M})$ and therefore $T|_{\mathcal{L}_{\mathcal{P}}}$ is not an answer set of $\Pi(gr_U(\mathcal{P}), \mathcal{M})$ and $\mathcal{M}$ is not an NM- model of $\mathcal{K}$. $\qquad\square$

This establishes the main theorem relating to the various special types of hybrid KBs discussed earlier.

**Theorem 2 (Main Theorem).** *(i) Let $\mathcal{K} = (\mathcal{T}, \mathcal{P})$ be a g-hybrid (resp. an $r^+$-hybrid) knowledge base. Let $\mathcal{M} = \langle U, T, T \rangle$ be a total here-and-there model of the stable closure of $\mathcal{K}$. Then $\mathcal{M}$ is in equilibrium if and only if $\langle U, T \rangle$ is an NM-model (resp. $r^+$-NM-model) of $\mathcal{K}$.*

*(ii) Let $\mathcal{K} = (\mathcal{T}, \mathcal{P})$ be an r-hybrid knowledge base. Let $\mathcal{M} = \langle U, T, T \rangle$ be an Herbrand model of the stable closure of $\mathcal{K}$. Then $\mathcal{M}$ is in equilibrium in the sense of [16] if and only if $\langle U, T \rangle$ is an r-NM-model of $\mathcal{K}$.*

*Example 5.* Consider again the hybrid knowledge base $\mathcal{K} = (\mathcal{T}, \mathcal{P})$, with $\mathcal{T}$ and $\mathcal{P}$ as in Example 1. The stable closure of $\mathcal{K}$, $st(\mathcal{K}) = \mathcal{T} \cup st(\mathcal{T}) \cup \mathcal{P}$ is

$$\forall x.PERSON(x) \rightarrow (AGENT(x) \wedge (\exists y.HAS\text{-}MOTHER(x,y)))$$
$$\forall x.(\exists y.HAS\text{-}MOTHER(x,y)) \rightarrow ANIMAL(x)$$
$$\forall x.PERSON(x) \vee \neg PERSON(x)$$
$$\forall x.AGENT(x) \vee \neg AGENT(x)$$
$$\forall x.ANIMAL(x) \vee \neg ANIMAL(x)$$
$$\forall x,y.HAS\text{-}MOTHER(x,y) \vee \neg HAS\text{-}MOTHER(x,y)$$
$$\forall x.AGENT(x) \wedge \neg machine(x) \rightarrow PERSON(x)$$
$$AGENT(DaveB)$$

Consider the total HT-model $\mathcal{M}_{HT} = \langle U, I, I \rangle$ of $st(\mathcal{K})$, with $U, I$ as in Example 4. $\mathcal{M}_{HT}$ is *not* an equilibrium model of $st(\mathcal{K})$, since $\mathcal{M}_{HT}$ is not minimal among all models: $\langle U, I', I \rangle$, with $I' = I \backslash \{machine(DaveB)\}$, is a model of $st(\mathcal{K})$. Furthermore, it is easy to verify that $\langle U, I', I' \rangle$ is not a model of $st(\mathcal{K})$.

Now, consider the total HT-model $\mathcal{M}'_{HT} = \langle U, M, M \rangle$, with $U$ as before, and

$$M = \{AGENT(DaveB), PERSON(DaveB),$$
$$ANIMAL(DaveB), HAS\text{-}NAME(DaveB, k)\}.$$

$\mathcal{M}'_{HT}$ is an equilibrium model of $st(\mathcal{K})$. Indeed, consider any $M' \subset M$. It is easy to verify that $\langle U, M', M \rangle$ is not a model of $st(\mathcal{K})$.

## 6.1 Discussion

We have seen that quantified equilibrium logic captures three of the main approaches to integrating classical, first-order or DL knowledge bases with nonmonotonic rules under the answer set semantics, in a modular, hybrid approach. However, QEL has a quite distinct flavor from those of r-hybrid, $r^+$-hybrid and g-hybrid KBs. Each of these hybrid approaches has a semantics composed of two different components: a classical model on the one hand and an answer set on the other. The style of QEL is different. There is one semantics and one kind of model that covers both types of knowledge. The only distinction we make is that for that part of the knowledge base considered to be classical and monotonic we add a stability condition to obtain the intended interpretation.

There are other features of the approach using QEL that are worth highlighting. First, it is based on a simple minimal model semantics in a known non-classical logic. No reducts are involved and, consequently, the equilibrium construction applies directly to arbitrary first-order theories. The rule part $\mathcal{P}$ of a knowledge base might therefore comprise, say, a nested logic program, where the heads and bodies of rules may be arbitrary boolean formulas, or perhaps rules permitting nestings of the implication connective. While answer sets have recently been defined for such general formulas, more work would be needed to provide integration in a hybrid KB setting.[13] Evidently QEL in the general case is undecidable, so for extensions of the rule language syntax for practical applications one may wish to study restrictions analogous to safety or guardedness. Second, the logic $\mathbf{QHT}^s_=$ can be applied to characterise properties such as the strong equivalence of programs and theories [11, 17]. While strong equivalence and related

---

[13] For a recent extension of answer sets to first-order formulas, see [6].

concepts have been much studied recently in ASP, their characterisation in the case of hybrid KBs remains uncharted territory. The fact that QEL provides a single semantics for hybrid KBs means that a simple concept of strong equivalence is applicable to such KBs and characterisable using the underlying logic, $\mathbf{QHT}^s_{=}$. We now describe briefly how $\mathbf{QHT}^s_{=}$ can be applied in this context.

### 6.2 An application to the strong equivalence of knowledge bases

Generally speaking it is important to know when different reconstructions of a given body of knowledge or state of affairs are equivalent and lead to essentially the same solutions. In the case of knowledge reconstructed in classical logic, ordinary logical equivalence can serve as a suitable concept when applied to theories formulated in the same vocabulary. In the case where nonmonotonic rules are present, however, one would like to know that equivalence between KBs is also robust, since two sets of rules may have the same answer sets yet behave very differently once they are embedded in some larger context. A robust or modular notion of equivalence for logic programs should therefore require that programs behave similarly when extended by any further programs. This leads to the following concept of *strong* equivalence: programs $\Pi_1$ and $\Pi_2$ are strongly equivalent if and only if for any set of rules $\Sigma$, $\Pi_1 \cup \Sigma$ and $\Pi_2 \cup \Sigma$ have the same answer sets. This concept of strong equivalence for logic programs in ASP was introduced and studied in [10] and has given rise to a substantial body of further work looking at different characterisations, new variations and applications of the idea, as well as the development of systems to test for strong equivalence.

In the case of hybrid knowledge bases $\mathcal{K} = (\mathcal{T}, \mathcal{P})$, various kinds of equivalence can be specified, according to whether one or other or both of the components $\mathcal{T}$ and $\mathcal{P}$ are allowed to vary. Let us illustrate for simplicity the case where $\mathcal{T}$ is fixed and $\mathcal{P}$ may vary; the extension to other cases is straightforward.

**Definition 8.** *Let $\mathcal{K}_1 = (\mathcal{T}, \mathcal{P}_1)$ and $\mathcal{K}_2 = (\mathcal{T}, \mathcal{P}_2)$ be two hybrid KBs based on the same classical theory $\mathcal{T}$. $\mathcal{K}_1$ and $\mathcal{K}_2$ are said to be* strongly equivalent *if for any set of rules $\mathcal{P}$, $(\mathcal{T}, \mathcal{P}_1 \cup \mathcal{P})$ and $(\mathcal{T}, \mathcal{P}_2 \cup \mathcal{P})$ have the same NM-models.*

The following characterisation of strong equivalence is an immediate consequence of Theorem 1 and the main theorem of [11].

**Proposition 7.** *Hybrid KBs $\mathcal{K}_1 = (\mathcal{T}, \mathcal{P}_1)$ and $\mathcal{K}_2 = (\mathcal{T}, \mathcal{P}_2)$ are strongly equivalent if and only if $\mathcal{P}_1$ and $\mathcal{P}_2$ are logically equivalent in $\mathbf{QHT}^s_{=}$.*

In other words, although we consider the effect of adding arbitrary nonmonotonic rules to a knowledge base, ordinary logical equivalence in $\mathbf{QHT}^s_{=}$ is a necessary and sufficient condition for strong equivalence.

It is interesting to note here that meaning-preserving relations among ontologies have recently become a topic of interest in the DL community where logical concepts such as that of conservative extension are currently being studied and applied [7]. A unified, logical approach to hybrid KBs such as that developed here should lend itself well to the application of such concepts.

# 7  Related Work and Conclusions

We have provided a general notion of hybrid knowledge base, combining first-order theories with nonmonotonic rules, with the aim of comparing and contrasting some of the different variants of hybrid KBs found in the literature [18–20, 9]. We presented a version of quantified equilibrium logic, QEL, without the unique names assumption, as a unified logical foundation for hybrid knowledge bases. We showed how for a hybrid knowledge base $\mathcal{K}$ there is a natural correspondence between the nonmonotonic models of $\mathcal{K}$ and the equilibrium models of what we call the *stable closure* of $\mathcal{K}$. This yields a way to capture in QEL the semantics of the g-hybrid KBs of Heymans et al. [9] and the r-hybrid KBs of Rosati [19], where the latter is defined without the UNA but for safe programs. Similarly, the version of QEL with UNA captures the semantics of r-hybrid KBs as defined in [18, 20]. It is important to note that the aim of this paper was not that of providing new kinds of safety conditions or decidability results; these issues are ably dealt with in the literature reviewed here. Rather our objective has been to show how classical and nonmonotonic theories might be unified under a single semantical model. In part, as [9] show with their reduction of DL knowledge bases to open answer set programs, this can also be achieved (at some cost of translation) in other approaches. What distinguishes QEL is the fact that it is based on a standard, nonclassical logic, $\mathbf{QHT}^s_=$, which can therefore provide a unified logical foundation for such extensions of (open) ASP. To illustrate the usefulness of our framework we showed how the logic $\mathbf{QHT}^s_=$ also captures a natural concept of strong equivalence between hybrid knowledge bases.

There are several other approaches to combining languages for Ontologies with nonmonotonic rules which can be divided into two main streams [3]: approaches which define integration of rules and ontologies (a) by entailment, ie. querying classical knowledge bases through special predicates the rules body, and (b) on the basis of single models, ie. defining a common notion of combined model.

The most prominent of the former kind of approaches are dl-programs [5] and their generalization, HEX-programs [4]. Although these approaches both are based on Answer Set programming like our approach, the orthogonal view of integration by entailment can probably not be captured by a simple embedding in QEL. Another such approach which allows querying classical KBs from a nonmonotonic rules language is based on Defeasible Logic [21].

As for the second stream, variants of Autoepistemic Logic [2], and the logic of minimal knowledge and negation as failure (MKNF) [13] have been recently proposed in the literature. Similar to our approach, both these approaches embed a combined knowledge base in a unifying logic. Remarkably however, both [2] and [13] use modal logics which syntactically and semantically extend first-order logics. Thus, in these approaches, embedding of the classical part of the theory is trivial, whereas the nonmonotonic rules part needs to be rewritten in terms of modal formulae. Our approach is orthogonal, as we base on a non-classical logic where the nonmonotonic rules are trivially embedded, but the stable closure guarantees classical behavior of certain predicates.

In future work we hope to consider further aspects of applying QEL to the domain of hybrid knowledge systems. Extending the language with functions symbols and with strong negation is a routine task, since QEL includes these items already. We also plan

to consider in the future how QEL can be used to define a catalogue of logical relations between hybrid KBs.

## References

1. C. Baral. *Knowledge Representation, Reasoning and Declarative Problem Solving*. CUP, 2002.
2. J. de Bruijn, T. Eiter, A. Polleres, H. Tompits. Embedding Non-Ground Logic Programs into Autoepistemic Logic for Knowledge-Base Combination. *IJCAI 2007*.
3. J. de Bruijn, T. Eiter, A. Polleres, H. Tompits. On Representational Issues about Combinations of Classical Theories with Nonmonotonic Rules. *KSEM 2006*.
4. T. Eiter, G. Ianni, R. Schindlauer, H. Tompits. A Uniform Integration of Higher-Order Reasoning and External Evaluations in Answer-Set Programming. *IJCAI 2005*.
5. T. Eiter, T. Lukasiewicz, R. Schindlauer, H. Tompits. Combining Answer Set Programming with Description Logics for the Semantic Web. *KR 2004*.
6. Paolo Ferraris, Joohyung Lee, and Vladimir Lifschitz. A new perspective on stable models. IJCAI 2007.
7. S. Ghilardi, C. Lutz, F. Wolter. Did I damage my ontology: A Case for Conservative Extensions of Description Logics. *KR 2006*.
8. S. Heymans, D. van Nieuwenborgh, D. Vermeir. Open answer set programming with guarded programs. *ACM Transactions on Computational Logic*. Accepted for Publication.
9. S. Heymans, L. Predoiu, C. Feier, J. de Bruijn, D. van Nieuwenborgh. G-hybrid Knowledge Bases. *ALPSWS 2006*.
10. V. Lifschitz, D. Pearce, and A. Valverde. Strongly equivalent logic programs. *ACM Transactions on Computational Logic*, 2:526–541, 2001.
11. V. Lifschitz, D. Pearce, and A. Valverde. A Characterization of Strong Equivalence for Logic Programs with Variables. *Proceedings LPNMR 2007*, Springer, to appear.
12. V. Lifschitz, T. Woo. Answer Sets in General Nonmonotonic Reasoning (preliminary report). *KR 1992*.
13. B. Motik, R. Rosati. A faithful integration of description logics with logic programming. IJCAI 2007.
14. D. Pearce. A new Logical Characterization of Stable Models and Answer Sets. *NMELP 96*, 1997.
15. D. Pearce. Equilibrium logic. *Annals of Mathematics and Artificial Intelligence*, 2006. In press, available at `http://dx.doi.org/10.1007/s10472-006-9028-z`.
16. D. Pearce, A. Valverde. A First-Order Nonmonotonic Extension of Constructive Logic. *Studia Logica*, 80, 2005.
17. D. Pearce, A. Valverde. Quantified Equilibrium Logic. *Tech. report, Univ. Rey Juan Carlos*, 2006. `http://www.satd.uma.es/matap/investig/tr/ma06_02.pdf`.
18. R. Rosati. On the Decidability and Complexity of Integrating Ontologies and Rules. *Journal of Web Semantics*, 3(1), 2005.
19. R. Rosati. Semantic and Computational Advantages of the Safe Integration of Ontologies and Rules. *PPSWR 2005*.
20. R. Rosati. $\mathcal{DL} + log$: Tight Integration of Description Logics and Disjunctive Datalog. *KR 2006*.
21. K. Wang, D. Billington, J. Blee, G. Antoniou. Combining Description Logic and Defeasible Logic for the Semantic Web. *RuleML-2004*.
22. D. van Dalen. *Logic and Structure*. Springer, 1983.