A Logic for Hybrid Rules

Axel Polleres

Universidad Rey Juan Carlos

Universidad Rey Juan Carlos, Madrid

Joint work with: **Jos de Bruijn**, DERI Innsbruck,
David Pearce, Univ. Rey Juan Carlos,
Agustín Valverde, Univ. de Málaga

RuleML Workshop 2006

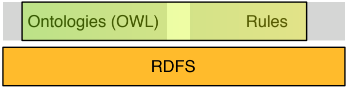
Outline

Motivation – Hybrid Knowledge Bases

Combine **rules with negation as failure** with **classical theories**:

- **Hyrid KB** approaches rely on (variants of) the Answer Set Semantics. [Rosati,2005/2005b/2006, Heymans, et al. 2006]
- Defined for **syntactically limited** programs/FOL theories
- All give a **modular definition** of models by projection+reduct.

Questions:



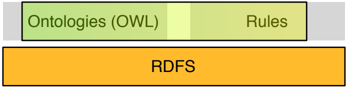
- Can we generalize these combinations in a (non-classical) logic, i.e. with a non-modular model definition?
- Does this provide us with notions of equivalence commonly used (strong equivalence, uniform equivalence, etc.)?

Motivation – Hybrid Knowledge Bases

Combine **rules with negation as failure** with **classical theories**:

- **Hyrid KB** approaches rely on (variants of) the Answer Set Semantics. [Rosati,2005/2005b/2006, Heymans, et al. 2006]
- Defined for **syntactically limited** programs/FOL theories
- All give a **modular definition** of models by projection+reduct.

Questions:



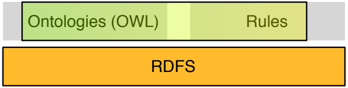
- Can we generalize these combinations in a (non-classical) logic, i.e. with a non-modular model definition?
- Does this provide us with notions of equivalence commonly used (strong equivalence, uniform equivalence, etc.)?

Motivation – Hybrid Knowledge Bases

Combine **rules with negation as failure** with **classical theories**:

- **Hyrid KB** approaches rely on (variants of) the Answer Set Semantics. [Rosati,2005/2005b/2006, Heymans, et al. 2006]
- Defined for **syntactically limited** programs/FOL theories
- All give a **modular definition** of models by projection+reduct.

Questions:



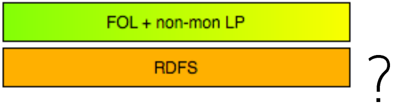
- Can we generalize these combinations in a (non-classical) logic, i.e. with a non-modular model definition?
- Does this provide us with notions of equivalence commonly used (strong equivalence, uniform equivalence, etc.)?

Motivation – Hybrid Knowledge Bases

Combine **rules with negation as failure** with **classical theories**:

- **Hyrid KB** approaches rely on (variants of) the Answer Set Semantics. [Rosati,2005/2005b/2006, Heymans, et al. 2006]
- Defined for **syntactically limited** programs/FOL theories
- All give a **modular definition** of models by projection+reduct.

Questions:



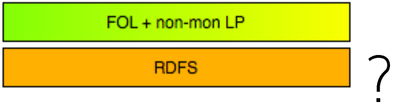
- Can we generalize these combinations in a (non-classical) logic, i.e. with a non-modular model definition?
- Does this provide us with notions of equivalence commonly used (strong equivalence, uniform equivalence, etc.)?

Motivation – Hybrid Knowledge Bases

Combine **rules with negation as failure** with **classical theories**:

- **Hyrid KB** approaches rely on (variants of) the Answer Set Semantics. [Rosati,2005/2005b/2006, Heymans, et al. 2006]
- Defined for **syntactically limited** programs/FOL theories
- All give a **modular definition** of models by projection+reduct.

Questions:



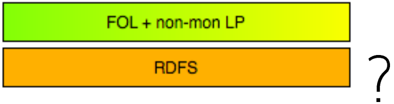
- Can we generalize these combinations in a (non-classical) logic, i.e. with a non-modular model definition?
- Does this provide us with notions of equivalence commonly used (strong equivalence, uniform equivalence, etc.)?

Motivation – Hybrid Knowledge Bases

Combine **rules with negation as failure** with **classical theories**:

- **Hyrid KB** approaches rely on (variants of) the Answer Set Semantics. [Rosati,2005/2005b/2006, Heymans, et al. 2006]
- Defined for **syntactically limited** programs/FOL theories
- All give a **modular definition** of models by projection+reduct.

Questions:



- Can we generalize these combinations in a (non-classical) logic, i.e. with a non-modular model definition?
- Does this provide us with notions of equivalence commonly used (strong equivalence, uniform equivalence, etc.)?

Hybrid Knowledge Bases – Generalized Definition

$\mathcal{K} = (\mathcal{T}, \mathcal{P})$ *hybrid knowledge base*:

- classical first-order theory $\mathcal{T}$ over function-free language $\mathcal{L}_\mathcal{T} = \langle C, P_\mathcal{T} \rangle$
- a logic program $\mathcal{P}$ over function-free language $\mathcal{L} = \langle C, P_\mathcal{T} \cup P_\mathcal{P} \rangle$, i.e. a set of rules:

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

  where $P_\mathcal{T} \cap P_\mathcal{P} = \emptyset$

Note:

- $\mathcal{T}$ and $\mathcal{P}$ talk about the same constants, and
- allowed predicate symbols in $\mathcal{P}$ are a superset of the predicate symbols in $\mathcal{L}_\mathcal{T}$.

Hybrid Knowledge Bases – Generalized Definition

$\mathcal{K} = (\mathcal{T}, \mathcal{P})$ *hybrid knowledge base*:

- classical first-order theory $\mathcal{T}$ over function-free language $\mathcal{L}_\mathcal{T} = \langle C, P_\mathcal{T} \rangle$
- a logic program $\mathcal{P}$ over function-free language $\mathcal{L} = \langle C, P_\mathcal{T} \cup P_\mathcal{P} \rangle$, i.e. a set of rules:

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

  where $P_\mathcal{T} \cap P_\mathcal{P} = \emptyset$

Note:

- $\mathcal{T}$ and $\mathcal{P}$ talk about the same constants, and
- allowed predicate symbols in $\mathcal{P}$ are a superset of the predicate symbols in $\mathcal{L}_\mathcal{T}$.

Hybrid Knowledge Bases – Generalized Definition

$\mathcal{K} = (\mathcal{T}, \mathcal{P})$ *hybrid knowledge base*:

- classical first-order theory $\mathcal{T}$ over function-free language $\mathcal{L}_\mathcal{T} = \langle C, P_\mathcal{T} \rangle$
- a logic program $\mathcal{P}$ over function-free language $\mathcal{L} = \langle C, P_\mathcal{T} \cup P_\mathcal{P} \rangle$, i.e. a set of rules:

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

where $P_\mathcal{T} \cap P_\mathcal{P} = \emptyset$

Note:

- $\mathcal{T}$ and $\mathcal{P}$ talk about the same constants, and
- allowed predicate symbols in $\mathcal{P}$ are a superset of the predicate symbols in $\mathcal{L}_\mathcal{T}$.

Hybrid Knowledge Bases – Generalized Definition

$\mathcal{K} = (\mathcal{T}, \mathcal{P})$ *hybrid knowledge base*:

- classical first-order theory $\mathcal{T}$ over function-free language $\mathcal{L}_\mathcal{T} = \langle C, P_\mathcal{T} \rangle$
- a logic program $\mathcal{P}$ over function-free language $\mathcal{L} = \langle C, P_\mathcal{T} \cup P_\mathcal{P} \rangle$, i.e. a set of rules:

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

where $P_\mathcal{T} \cap P_\mathcal{P} = \emptyset$

Note:

- $\mathcal{T}$ and $\mathcal{P}$ talk about the same constants, and
- allowed predicate symbols in $\mathcal{P}$ are a superset of the predicate symbols in $\mathcal{L}_\mathcal{T}$.

Hybrid Knowledge Bases – Generalized Definition

$\mathcal{K} = (\mathcal{T}, \mathcal{P})$ *hybrid knowledge base*:

- classical first-order theory $\mathcal{T}$ over function-free language $\mathcal{L}_\mathcal{T} = \langle C, P_\mathcal{T} \rangle$
- a logic program $\mathcal{P}$ over function-free language $\mathcal{L} = \langle C, P_\mathcal{T} \cup P_\mathcal{P} \rangle$, i.e. a set of rules:

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

where $P_\mathcal{T} \cap P_\mathcal{P} = \emptyset$

Note:

- $\mathcal{T}$ and $\mathcal{P}$ talk about the same constants, and
- allowed predicate symbols in $\mathcal{P}$ are a superset of the predicate symbols in $\mathcal{L}_\mathcal{T}$.

Hybrid Knowledge Bases – Projection:

Overall idea for a nonmonotonic semantics: "evaluate" $\mathcal{P}$ wrt a classical model of the theory and then compute stable models.

Let $\mathcal{P}$ be a *ground* program an $\mathcal{I} = \langle U, I \rangle$ an $\mathcal{L}$-structure, with $U = (D, \sigma)$.

$\Pi(\mathcal{P}, \mathcal{I})$, the projection of $\mathcal{P}$ wrt $\mathcal{I}$, obtained by

1. deleting each rule with head literal $p(t)$ (or $\neg p(t)$) over $At_D(C, P_T)$ such that $p(\sigma(t)) \in I$ (or $p(\sigma(t)) \notin I$)
2. deleting each rule with body literal $p(t)$ (or $\neg p(t)$) over $At_D(C, P_T)$ such that $p(\sigma(t)) \notin I$ (or $p(\sigma(t)) \in I$);

and deleting occurrences of literals from $\mathcal{L}_T$ from remaining rules.

Hybrid Knowledge Bases – Projection:

Overall idea for a nonmonotonic semantics: "evaluate" $\mathcal{P}$ wrt a classical model of the theory and then compute stable models.

Let $\mathcal{P}$ be a *ground* program an $\mathcal{I} = \langle U, I \rangle$ an $\mathcal{L}$-structure, with $U = (D, \sigma)$.

$\Pi(\mathcal{P}, \mathcal{I})$, the projection of $\mathcal{P}$ wrt $\mathcal{I}$, obtained by

1. deleting each rule with head literal $p(t)$ (or $\neg p(t)$) over $At_D(C, P_\mathcal{T})$ such that $p(\sigma(t)) \in I$ (or $p(\sigma(t)) \notin I$)
2. deleting each rule with body literal $p(t)$ (or $\neg p(t)$) over $At_D(C, P_\mathcal{T})$ such that $p(\sigma(t)) \notin I$ (or $p(\sigma(t)) \in I$);

and deleting occurrences of literals from $\mathcal{L}_\mathcal{T}$ from remaining rules.

Hybrid Knowledge Bases – Projection:

Overall idea for a nonmonotonic semantics: "evaluate" $\mathcal{P}$ wrt a classical model of the theory and then compute stable models.

Let $\mathcal{P}$ be a *ground* program an $\mathcal{I} = \langle U, I \rangle$ an $\mathcal{L}$-structure, with $U = (D, \sigma)$.

$\Pi(\mathcal{P}, \mathcal{I})$, the projection of $\mathcal{P}$ wrt $\mathcal{I}$, obtained by

1. deleting each rule with head literal $p(t)$ (or $\neg p(t)$) over $At_D(C, P_\mathcal{T})$ such that $p(\sigma(t)) \in I$ (or $p(\sigma(t)) \notin I$)
2. deleting each rule with body literal $p(t)$ (or $\neg p(t)$) over $At_D(C, P_\mathcal{T})$ such that $p(\sigma(t)) \notin I$ (or $p(\sigma(t)) \in I$);

and deleting occurrences of literals from $\mathcal{L}_\mathcal{T}$ from remaining rules.

Hybrid Knowledge Bases – Projection:

Overall idea for a nonmonotonic semantics: "evaluate" $\mathcal{P}$ wrt a classical model of the theory and then compute stable models.

Let $\mathcal{P}$ be a *ground* program an $\mathcal{I} = \langle U, I \rangle$ an $\mathcal{L}$-structure, with $U = (D, \sigma)$.

$\Pi(\mathcal{P}, \mathcal{I})$, the projection of $\mathcal{P}$ wrt $\mathcal{I}$, obtained by

1. deleting each rule with head literal $p(t)$ (or $\neg p(t)$) over $At_D(C, P_\mathcal{T})$ such that $p(\sigma(t)) \in I$ (or $p(\sigma(t)) \notin I$)
2. deleting each rule with body literal $p(t)$ (or $\neg p(t)$) over $At_D(C, P_\mathcal{T})$ such that $p(\sigma(t)) \notin I$ (or $p(\sigma(t)) \in I$);

and deleting occurrences of literals from $\mathcal{L}_\mathcal{T}$ from remaining rules.

Hybrid Knowledge Bases – Projection:

Overall idea for a nonmonotonic semantics: "evaluate" $\mathcal{P}$ wrt a classical model of the theory and then compute stable models.

Let $\mathcal{P}$ be a *ground* program an $\mathcal{I} = \langle U, I \rangle$ an $\mathcal{L}$-structure, with $U = (D, \sigma)$.

$\Pi(\mathcal{P}, \mathcal{I})$, the projection of $\mathcal{P}$ wrt $\mathcal{I}$, obtained by

1. deleting each rule with head literal $p(t)$ (or $\neg p(t)$) over $At_D(C, P_\mathcal{T})$ such that $p(\sigma(t)) \in I$ (or $p(\sigma(t)) \notin I$)
2. deleting each rule with body literal $p(t)$ (or $\neg p(t)$) over $At_D(C, P_\mathcal{T})$ such that $p(\sigma(t)) \notin I$ (or $p(\sigma(t)) \in I$);

and deleting occurrences of literals from $\mathcal{L}_\mathcal{T}$ from remaining rules.

Hybrid Knowledge Bases – NM-model:

Overall idea for a nonmonotonic semantics: "evaluate" $\mathcal{P}$ wrt a classical model of the theory and then compute stable models.

Let $\mathcal{K} = (\mathcal{T}, \mathcal{P})$ be a hybrid knowledge base. An **NM-model** $\mathcal{M} = \langle U, I \rangle$ of a hybrid knowledge base $\mathcal{K}$ is a first-order $\mathcal{L}$-structure such that

1. $\mathcal{M}|_{\mathcal{L}_\mathcal{T}}$ is a model of $\mathcal{T}$ and
2. $\mathcal{M}|_{\mathcal{L}_\mathcal{P}}$ is a stable model set of $\Pi(gr_U(\mathcal{P}), \mathcal{M}|_{\mathcal{L}_\mathcal{T}})$, i.e. $\mathcal{M}|_{\mathcal{L}_\mathcal{P}}$ is a minimal Herbrand Model of the reduct $\Pi(gr_U(\mathcal{P}), \mathcal{M})^{\mathcal{M}|_{\mathcal{L}_\mathcal{P}}}$, obtained by taking all rules:
   - such that $\mathcal{M}|_{\mathcal{L}_\mathcal{P}} \models a_i$ for negative head atoms a_i and
   - $\mathcal{I} \not\models b_j$ for all negative body atoms b_j.

Hybrid Knowledge Bases – NM-model:

Overall idea for a nonmonotonic semantics: "evaluate" $\mathcal{P}$ wrt a classical model of the theory and then compute stable models.

Let $\mathcal{K} = (\mathcal{T}, \mathcal{P})$ be a hybrid knowledge base. An **NM-model** $\mathcal{M} = \langle U, I \rangle$ of a hybrid knowledge base $\mathcal{K}$ is a first-order $\mathcal{L}$-structure such that

1. $\mathcal{M}|_{\mathcal{L}_\mathcal{T}}$ is a model of $\mathcal{T}$ and
2. $\mathcal{M}|_{\mathcal{L}_\mathcal{P}}$ is a stable model set of $\Pi(gr_U(\mathcal{P}), \mathcal{M}|_{\mathcal{L}_\mathcal{T}})$, i.e. $\mathcal{M}|_{\mathcal{L}_\mathcal{P}}$ is a minimal Herbrand Model of the reduct $\Pi(gr_U(\mathcal{P}), \mathcal{M})^{\mathcal{M}|_{\mathcal{L}_\mathcal{P}}}$, obtained by taking all rules:
   - such that $\mathcal{M}|_{\mathcal{L}_\mathcal{P}} \models a_i$ for negative head atoms a_i and
   - $\mathcal{I} \not\models b_j$ for all negative body atoms b_j.

Hybrid Knowledge Bases – NM-model:

Overall idea for a nonmonotonic semantics: "evaluate" $\mathcal{P}$ wrt a classical model of the theory and then compute stable models.

Let $\mathcal{K} = (\mathcal{T}, \mathcal{P})$ be a hybrid knowledge base. An **NM-model** $\mathcal{M} = \langle U, I \rangle$ of a hybrid knowledge base $\mathcal{K}$ is a first-order $\mathcal{L}$-structure such that

1. $\mathcal{M}|_{\mathcal{L}_\mathcal{T}}$ is a model of $\mathcal{T}$ and
2. $\mathcal{M}|_{\mathcal{L}_\mathcal{P}}$ is a stable model set of $\Pi(gr_U(\mathcal{P}), \mathcal{M}|_{\mathcal{L}_\mathcal{T}})$, i.e. $\mathcal{M}|_{\mathcal{L}_\mathcal{P}}$ is a minimal Herbrand Model of the reduct $\Pi(gr_U(\mathcal{P}), \mathcal{M})^{\mathcal{M}|_{\mathcal{L}_\mathcal{P}}}$, obtained by taking all rules:
   - such that $\mathcal{M}|_{\mathcal{L}_\mathcal{P}} \models a_i$ for negative head atoms a_i and
   - $\mathcal{I} \not\models b_j$ for all negative body atoms b_j.

Hybrid Knowledge Bases – NM-model:

Overall idea for a nonmonotonic semantics: "evaluate" $\mathcal{P}$ wrt a classical model of the theory and then compute stable models.

Let $\mathcal{K} = (\mathcal{T}, \mathcal{P})$ be a hybrid knowledge base. An **NM-model** $\mathcal{M} = \langle U, I \rangle$ of a hybrid knowledge base $\mathcal{K}$ is a first-order $\mathcal{L}$-structure such that

1. $\mathcal{M}|_{\mathcal{L}_\mathcal{T}}$ is a model of $\mathcal{T}$ and
2. $\mathcal{M}|_{\mathcal{L}_\mathcal{P}}$ is a stable model set of $\Pi(gr_U(\mathcal{P}), \mathcal{M}|_{\mathcal{L}_\mathcal{T}})$, i.e. $\mathcal{M}|_{\mathcal{L}_\mathcal{P}}$ is a minimal Herbrand Model of the reduct $\Pi(gr_U(\mathcal{P}), \mathcal{M})^{\mathcal{M}|_{\mathcal{L}_\mathcal{P}}}$, obtained by taking all rules:
   - such that $\mathcal{M}|_{\mathcal{L}_\mathcal{P}} \models a_i$ for negative head atoms a_i and
   - $\mathcal{I} \not\models b_j$ for all negative body atoms b_j.

Hybrid Knowledge Bases – NM-model:

Overall idea for a nonmonotonic semantics: "evaluate" $\mathcal{P}$ wrt a classical model of the theory and then compute stable models.

Let $\mathcal{K} = (\mathcal{T}, \mathcal{P})$ be a hybrid knowledge base. An **NM-model** $\mathcal{M} = \langle U, I \rangle$ of a hybrid knowledge base $\mathcal{K}$ is a first-order $\mathcal{L}$-structure such that

1. $\mathcal{M}|_{\mathcal{L}_\mathcal{T}}$ is a model of $\mathcal{T}$ and
2. $\mathcal{M}|_{\mathcal{L}_\mathcal{P}}$ is a stable model set of $\Pi(gr_U(\mathcal{P}), \mathcal{M}|_{\mathcal{L}_\mathcal{T}})$, i.e. $\mathcal{M}|_{\mathcal{L}_\mathcal{P}}$ is a minimal Herbrand Model of the reduct $\Pi(gr_U(\mathcal{P}), \mathcal{M})^{\mathcal{M}|_{\mathcal{L}_\mathcal{P}}}$, obtained by taking all rules:
   - such that $\mathcal{M}|_{\mathcal{L}_\mathcal{P}} \models a_i$ for negative head atoms a_i and
   - $\mathcal{I} \not\models b_j$ for all negative body atoms b_j.

Example – a small Hybrid KB:

Let $\mathcal{K} = (\mathcal{T}, \mathcal{P})$ with

$\mathcal{T}$: Each foaf:Person is a foaf:Agent:

$\forall x.PERSON(x) \rightarrow AGENT(x)$
$AGENT(David)$

$\mathcal{P}$: Some nonmonotonic rule on top

$PERSON(x) \leftarrow pcmember(x, LPNMR), AGENT(x), \neg machine(x)$
$pcmember(David, LPNMR)$

Is David a PERSON?

Recall: $\neg$ is "negation as failure" here!

Example – a small Hybrid KB:

Let $\mathcal{K} = (\mathcal{T}, \mathcal{P})$ with

$\mathcal{T}$: Each foaf:Person is a foaf:Agent:

$\forall x. PERSON(x) \rightarrow AGENT(x)$
$AGENT(David)$

$\mathcal{P}$: Some nonmonotonic rule on top

$PERSON(x) \leftarrow pcmember(x, LPNMR), AGENT(x), \neg machine(x)$
$pcmember(David, LPNMR)$

Is David a PERSON?

Recall: $\neg$ is "negation as failure" here!

Example - NM-models:

Classical models of $\mathcal{T}$:

$\forall x.PERSON(x) \rightarrow AGENT(x)$
$AGENT(David)$

$\mathcal{M}_1|_{\mathcal{L}_\mathcal{T}} = \{AGENT(David), \neg PERSON(David), \neg AGENT(LPNMR), \ldots\}$
$\mathcal{M}_2|_{\mathcal{L}_\mathcal{T}} = \{AGENT(David), PERSON(David), \neg AGENT(LPNMR), \ldots\}$

$gr_U(\mathcal{P})$

$PERSON(David) \leftarrow$
$pcmember(David, LPNMR), AGENT(David), \neg machine(David)$
$PERSON(LPNMR) \leftarrow$
$pcmember(LPNMR, LPNMR), AGENT(LPNMR), \neg machine(LPNMR)$

$pcmember(David, LPNMR)$

Is David a PERSON?

$\Pi(gr_U(\mathcal{P}), \mathcal{M}_2|_{\mathcal{L}_\mathcal{T}})$

Example - NM-models:

Classical models of $\mathcal{T}$:

$\forall x.PERSON(x) \rightarrow AGENT(x)$
$AGENT(David)$

$\mathcal{M}_1|_{\mathcal{L}_\mathcal{T}} = \{AGENT(David), \neg PERSON(David), \neg AGENT(LPNMR), \ldots\}$

$\mathcal{M}_2|_{\mathcal{L}_\mathcal{T}} = \{AGENT(David), PERSON(David), \neg AGENT(LPNMR), \ldots\}$

$\Pi(gr_U(\mathcal{P}), \mathcal{M}_1|_{\mathcal{L}_\mathcal{T}})$

$\leftarrow pcmember(David, LPNMR), \neg machine(David).$

$pcmember(David, LPNMR)$

No stable models!

Is David a PERSON?

$\Pi(gr_U(\mathcal{P}), \mathcal{M}_2|_{\mathcal{L}_\mathcal{T}})$

Example - NM-models:

Classical models of $\mathcal{T}$:

$\forall x.PERSON(x) \rightarrow AGENT(x)$
$AGENT(David)$

$\mathcal{M}_1|_{\mathcal{L}_\mathcal{T}} = \{AGENT(David), \neg PERSON(David), \neg AGENT(LPNMR), \ldots\}$
$\mathcal{M}_2|_{\mathcal{L}_\mathcal{T}} = \{AGENT(David), PERSON(David), \neg AGENT(LPNMR), \ldots\}$

$\Pi(gr_U(\mathcal{P}), \mathcal{M}_2|_{\mathcal{L}_\mathcal{T}})$

$pcmember(David, LPNMR)$

One stable model...

Is David a PERSON? Yes!

$PERSON(David)$ in all NM-models, i.e. $\mathcal{K} \models_{NM} PERSON(David)$

Example - NM-models:

Classical models of $\mathcal{T}$:

$\forall x.PERSON(x) \rightarrow AGENT(x)$
$AGENT(David)$

$\mathcal{M}_1|_{\mathcal{L}_\mathcal{T}} = \{AGENT(David), \neg PERSON(David), \neg AGENT(LPNMR), \ldots\}$
$\mathcal{M}_2|_{\mathcal{L}_\mathcal{T}} = \{AGENT(David), PERSON(David), \neg AGENT(LPNMR), \ldots\}$

$\Pi(gr_U(\mathcal{P}), \mathcal{M}_2|_{\mathcal{L}_\mathcal{T}})$

$pcmember(David, LPNMR)$

One stable model...

Is David a PERSON? Yes!
$PERSON(David)$ in all NM-models, i.e. $\mathcal{K} \models_{NM} PERSON(David)$

Equilibrium Logic

Back to our question for a non-classical logic which covers this. . .

- *Equilibrium logic* (Pearce, 1997) generalizes stable model semantics and answer set semantics for logic programs to arbitrary propositional theories.
- It is a nonmonotonic extension of the logic of Here-and-there (with strong negation).
- Model theory based on Kripke semantics for intuitionistic logic
- We need a first-order verstion here...

Equilibrium Logic

Back to our question for a non-classical logic which covers this...

- *Equilibrium logic* (Pearce, 1997) generalizes stable model semantics and answer set semantics for logic programs to arbitrary propositional theories.
- It is a nonmonotonic extension of the logic of Here-and-there (with strong negation).
- Model theory based on Kripke semantics for intuitionistic logic
- We need a first-order verstion here...

Quantified Here-and-there Logic

- $\mathbf{QHT^s}$ is complete for linear Kripke frames with two worlds "here" and "there" with a "static" domain over both worlds: $h \leq t$.
- here-and-there structures: $\mathcal{M} = \langle (D, \sigma), I_h, I_t \rangle$
- I_h, I_t are first-order-interpretations over D such that $I_h \subseteq I_t$.

The models are extended to all formulas via the rules known in intuitionistic logic, notions of validity and logical consequence relation are the ones for (intuitionistic) Kripke semantics.

Quantified Here-and-there Logic

- $\mathbf{QHT}^{s}$ is complete for linear Kripke frames with two worlds "here" and "there" with a "static" domain over both worlds: $h \leq t$.
- here-and-there structures: $\mathcal{M} = \langle (D, \sigma), I_h, I_t \rangle$
- I_h, I_t are first-order-interpretations over D such that $I_h \subseteq I_t$.

The models are extended to all formulas via the rules known in intuitionistic logic, notions of validity and logical consequence relation are the ones for (intuitionistic) Kripke semantics.

Quantified Here-and-there Logic

For $w \in \{h, t\}$:

- $\mathcal{M}, w \models \varphi \wedge \psi$ 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 $\mathcal{M}, w \models \psi$.
- $\mathcal{M}, t \models \varphi \rightarrow \psi$ iff $\mathcal{M}, t \not\models \varphi$ or $\mathcal{M}, t \models \psi$.
- $\mathcal{M}, h \models \varphi \rightarrow \psi$ iff $\mathcal{M}, t \models \varphi \rightarrow \psi$ 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 $d \in D$.

Quantified Equilibrium Logic (QEL)

- We write QHT^s-structures more briefly as ordered pairs of atoms $\langle H, T\rangle$, with $H \subseteq T$.
- An QHT^s-Structure $\langle H, T\rangle$ is said to be total if $H = T$
- Order relation: $\langle H, T\rangle \trianglelefteq \langle H', T'\rangle$ if $T = T'$ and $H \subseteq H'$
- $\langle H, T\rangle$ is an equilibrium model of Π if is
  - (i) $\langle H, T\rangle$minimal under $\trianglelefteq$, and
  - (ii) $\langle H, T\rangle$ is total.

QEL is determined by the equilibrium models of a theory.

Quantified Equilibrium Logic (QEL)

- We write QHT^s-structures more briefly as ordered pairs of atoms $\langle H, T \rangle$, with $H \subseteq T$.
- An QHT^s-Structure $\langle H, T \rangle$ is said to be total if $H = T$
- Order relation: $\langle H, T \rangle \trianglelefteq \langle H', T' \rangle$ if $T = T'$ and $H \subseteq H'$
- $\langle H, T \rangle$ is an equilibrium model of Π if is
  (i) $\langle H, T \rangle$minimal under $\trianglelefteq$, and
  (ii) $\langle H, T \rangle$ is total.

QEL is determined by the equilibrium models of a theory.

Quantified Equilibrium Logic (QEL)

- We write QHT^s-structures more briefly as ordered pairs of atoms $\langle H, T \rangle$, with $H \subseteq T$.
- An QHT^s-Structure $\langle H, T \rangle$ is said to be total if $H = T$
- Order relation: $\langle H, T \rangle \trianglelefteq \langle H', T' \rangle$ if $T = T'$ and $H \subseteq H'$
- $\langle H, T \rangle$ is an equilibrium model of Π if is
  (i) $\langle H, T \rangle$minimal under $\trianglelefteq$, and
  (ii) $\langle H, T \rangle$ is total.

QEL is determined by the equilibrium models of a theory.

Quantified Equilibrium Logic (QEL)

- We write QHT^s-structures more briefly as ordered pairs of atoms $\langle H, T\rangle$, with $H \subseteq T$.

- An QHT^s-Structure $\langle H, T\rangle$ is said to be total if $H = T$
- Order relation: $\langle H, T\rangle \trianglelefteq \langle H', T'\rangle$ if $T = T'$ and $H \subseteq H'$
- $\langle H, T\rangle$ is an equilibrium model of Π if is
  (i) $\langle H, T\rangle$minimal under $\trianglelefteq$, and
  (ii) $\langle H, T\rangle$ is total.

QEL is determined by the equilibrium models of a theory.

Quantified Equilibrium Logic (QEL)

- We write QHT^s-structures more briefly as ordered pairs of atoms $\langle H, T\rangle$, with $H \subseteq T$.
- An QHT^s-Structure $\langle H, T\rangle$ is said to be total if $H = T$
- Order relation: $\langle H, T\rangle \trianglelefteq \langle H', T'\rangle$ if $T = T'$ and $H \subseteq H'$
- $\langle H, T\rangle$ is an equilibrium model of Π if is
  - (i) $\langle H, T\rangle$minimal under $\trianglelefteq$, and
  - (ii) $\langle H, T\rangle$ is total.

QEL is determined by the equilibrium models of a theory.

Quantified Equilibrium Logic and Answer Set Semantics

- Equilibrium Logic generalizes Answer Set semantics for arbitrary formulae (including disjunctive and nested programs)
- Any rule

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

  is just treated as (universally closed) formula in QEL:

  $(\forall) a_1 \vee a_2 \vee \ldots \vee a_k \vee \neg a_{k+1} \vee \ldots \vee \neg a_l \leftarrow b_1 \wedge \ldots \wedge b_m \wedge \neg b_{m+1} \wedge \ldots \wedge \neg b_n$

- Equilibrium models correspond to (open) answer sets: $\langle T, T \rangle$ is a equilibrium model of $\mathcal{P}$ iff T is an answer set of Π.
- So, for $\mathcal{K} = (\emptyset, P)$ answer sets and QEL-models correspond!

UNA [Pearce&Valverde, 2005] and non-UNA [Pearce&Valverde, 2006] versions of QEL available.

Quantified Equilibrium Logic and Answer Set Semantics

- Equilibrium Logic generalizes Answer Set semantics for arbitrary formulae (including disjunctive and nested programs)
- Any rule

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

  is just treated as (universally closed) formula in QEL:

  $(\forall) a_1 \vee a_2 \vee \ldots \vee a_k \vee \neg a_{k+1} \vee \ldots \vee \neg a_l \leftarrow b_1 \wedge \ldots \wedge b_m \wedge \neg b_{m+1} \wedge \ldots \wedge \neg b_n$

- Equilibrium models correspond to (open) answer sets: $\langle T, T \rangle$ is a equilibrium model of $\mathcal{P}$ iff T is an answer set of Π.
- So, for $\mathcal{K} = (\emptyset, P)$ answer sets and QEL-models correspond!

UNA [Pearce&Valverde, 2005] and non-UNA [Pearce&Valverde, 2006] versions of QEL available.

Quantified Equilibrium Logic and Answer Set Semantics

- Equilibrium Logic generalizes Answer Set semantics for arbitrary formulae (including disjunctive and nested programs)
- Any rule

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

is just treated as (universally closed) formula in QEL:

$$(\forall) a_1 \vee a_2 \vee \ldots \vee a_k \vee \neg a_{k+1} \vee \ldots \vee \neg a_l \leftarrow b_1 \wedge \ldots \wedge b_m \wedge \neg b_{m+1} \wedge \ldots \wedge \neg b_n$$

- Equilibrium models correspond to (open) answer sets: $\langle T, T \rangle$ is a equilibrium model of $\mathcal{P}$ iff T is an answer set of Π.
- So, for $\mathcal{K} = (\emptyset, P)$ answer sets and QEL-models correspond!

UNA [Pearce&Valverde, 2005] and non-UNA [Pearce&Valverde, 2006] versions of QEL available.

Quantified Equilibrium Logic and Answer Set Semantics

- Equilibrium Logic generalizes Answer Set semantics for arbitrary formulae (including disjunctive and nested programs)
- Any rule

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

  is just treated as (universally closed) formula in QEL:

  $$(\forall) a_1 \vee a_2 \vee \ldots \vee a_k \vee \neg a_{k+1} \vee \ldots \vee \neg a_l \leftarrow b_1 \wedge \ldots \wedge b_m \wedge \neg b_{m+1} \wedge \ldots \wedge \neg b_n$$

- Equilibrium models correspond to (open) answer sets: $\langle T, T \rangle$ is a equilibrium model of $\mathcal{P}$ iff T is an answer set of Π.
- So, for $\mathcal{K} = (\emptyset, P)$ answer sets and QEL-models correspond!

UNA [Pearce&Valverde, 2005] and non-UNA [Pearce&Valverde, 2006] versions of QEL available.

Quantified Equilibrium Logic and Answer Set Semantics

- Equilibrium Logic generalizes Answer Set semantics for arbitrary formulae (including disjunctive and nested programs)
- Any rule

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

  is just treated as (universally closed) formula in QEL:

  $$(\forall) a_1 \vee a_2 \vee \ldots \vee a_k \vee \neg a_{k+1} \vee \ldots \vee \neg a_l \leftarrow b_1 \wedge \ldots \wedge b_m \wedge \neg b_{m+1} \wedge \ldots \wedge \neg b_n$$

- Equilibrium models correspond to (open) answer sets: $\langle T, T \rangle$ is a equilibrium model of $\mathcal{P}$ iff T is an answer set of Π.
- So, for $\mathcal{K} = (\emptyset, P)$ answer sets and QEL-models correspond!

UNA [Pearce&Valverde, 2005] and non-UNA [Pearce&Valverde, 2006] versions of QEL available.

Quantified Equilibrium Logic and Answer Set Semantics

- Equilibrium Logic generalizes Answer Set semantics for arbitrary formulae (including disjunctive and nested programs)
- Any rule

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

  is just treated as (universally closed) formula in QEL:

  $$(\forall) a_1 \vee a_2 \vee \ldots \vee a_k \vee \neg a_{k+1} \vee \ldots \vee \neg a_l \leftarrow b_1 \wedge \ldots \wedge b_m \wedge \neg b_{m+1} \wedge \ldots \wedge \neg b_n$$

- Equilibrium models correspond to (open) answer sets: $\langle T, T \rangle$ is a equilibrium model of $\mathcal{P}$ iff T is an answer set of Π.
- So, for $\mathcal{K} = (\emptyset, P)$ answer sets and QEL-models correspond!

UNA [Pearce&Valverde, 2005] and non-UNA [Pearce&Valverde, 2006] versions of QEL available.

Quantified Equilibrium Logic and Answer Set Semantics

- Equilibrium Logic generalizes Answer Set semantics for arbitrary formulae (including disjunctive and nested programs)
- Any rule

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

  is just treated as (universally closed) formula in QEL:

  $$(\forall) a_1 \vee a_2 \vee \ldots \vee a_k \vee \neg a_{k+1} \vee \ldots \vee \neg a_l \leftarrow b_1 \wedge \ldots \wedge b_m \wedge \neg b_{m+1} \wedge \ldots \wedge \neg b_n$$

- Equilibrium models correspond to (open) answer sets: $\langle T, T \rangle$ is a equilibrium model of $\mathcal{P}$ iff T is an answer set of Π.
- So, for $\mathcal{K} = (\emptyset, P)$ answer sets and QEL-models correspond!

UNA [Pearce&Valverde, 2005] and non-UNA [Pearce&Valverde, 2006] versions of QEL available.

Embedding Hybrid Knowledge Bases

Q: Does the correspondence extend to hybrid KBs? Yes!

Idea: define embedding based on the observation that adding LEM makes intuitionistic logic classical!

Given a hybrid KB $\mathcal{K} = (\mathcal{T}, \mathcal{P})$ we call $\mathcal{T} \cup st(\mathcal{T}) \cup \mathcal{P}$ the stable closure of $\mathcal{K}$, where $st(\mathcal{T}) = \{\forall x(p(x) \vee \neg p(x)) : p \in \mathcal{L}_\mathcal{T}\}$.

Wake up! Main theorem of the paper!!! ;-)

Theorem

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 an equilibrium model if and only if it is an NM-model of $\mathcal{K}$.

Embedding Hybrid Knowledge Bases

Q: Does the correspondence extend to hybrid KBs? Yes!

Idea: define embedding based on the observation that adding LEM makes intuitionistic logic classical!

Given a hybrid KB $\mathcal{K} = (\mathcal{T}, \mathcal{P})$ we call $\mathcal{T} \cup st(\mathcal{T}) \cup \mathcal{P}$ the stable closure of $\mathcal{K}$, where $st(\mathcal{T}) = \{\forall x(p(x) \vee \neg p(x)) : p \in \mathcal{L}_{\mathcal{T}}\}$.

Wake up! Main theorem of the paper!!! ;-)

Theorem

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 an equilibrium model if and only if it is an NM-model of $\mathcal{K}$.

Embedding Hybrid Knowledge Bases

Q: Does the correspondence extend to hybrid KBs? Yes!

Idea: define embedding based on the observation that adding LEM makes intuitionistic logic classical!

Given a hybrid KB $\mathcal{K} = (\mathcal{T}, \mathcal{P})$ we call $\mathcal{T} \cup st(\mathcal{T}) \cup \mathcal{P}$ the stable closure of $\mathcal{K}$, where $st(\mathcal{T}) = \{\forall x(p(x) \vee \neg p(x)) : p \in \mathcal{L}_{\mathcal{T}}\}$.

Wake up! Main theorem of the paper!!! ;-)

Theorem

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 an equilibrium model if and only if it is an NM-model of $\mathcal{K}$.

Embedding Hybrid Knowledge Bases

Q: Does the correspondence extend to hybrid KBs? Yes!

Idea: define embedding based on the observation that adding LEM makes intuitionistic logic classical!

Given a hybrid KB $\mathcal{K} = (\mathcal{T}, \mathcal{P})$ we call $\mathcal{T} \cup st(\mathcal{T}) \cup \mathcal{P}$ the stable closure of $\mathcal{K}$, where $st(\mathcal{T}) = \{\forall x(p(x) \vee \neg p(x)) : p \in \mathcal{L}_\mathcal{T}\}$.

Wake up! Main theorem of the paper!!! ;-)

Theorem

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 an equilibrium model if and only if it is an NM-model of $\mathcal{K}$.

Example - *stable closure* of $\mathcal{K}$:

$st(\mathcal{K}) = \mathcal{T} \cup st(\mathcal{T}) \cup \mathcal{P}$

$\forall x.PERSON(x) \rightarrow AGENT(x)$
$AGENT(David)$
$\forall x.PERSON(x) \vee \neg PERSON(x)$
$\forall x.AGENT(x) \vee \neg AGENT(x)$
$\forall x.PERSON(x) \leftarrow pcmember(x, LPNMR) \wedge AGENT(x) \wedge \neg machine(x)$
$pcmember(David, LPNMR)$

There IS a classical model of this theory
$\mathcal{M} = \{\neg PERSON(David), machine(David), \ldots\}$

Thus:
$K \not\models_{FOL} PERSON(David)$

Example - *stable closure* of $\mathcal{K}$:

$st(\mathcal{K}) = \mathcal{T} \cup st(\mathcal{T}) \cup \mathcal{P}$

$\forall x.PERSON(x) \rightarrow AGENT(x)$
$AGENT(David)$
$\forall x.PERSON(x) \vee \neg PERSON(x)$
$\forall x.AGENT(x) \vee \neg AGENT(x)$
$\forall x.PERSON(x) \leftarrow pcmember(x, LPNMR) \wedge AGENT(x) \wedge \neg machine(x)$
$pcmember(David, LPNMR)$

There IS a classical model of this theory
$\mathcal{M} = \{\neg PERSON(David), machine(David), \ldots\}$

Thus:
$K \not\models_{FOL} PERSON(David)$

Example - *stable closure* of $\mathcal{K}$:

$st(\mathcal{K}) = \mathcal{T} \cup st(\mathcal{T}) \cup \mathcal{P}$

$\forall x.PERSON(x) \rightarrow AGENT(x)$
$AGENT(David)$
$\forall x.PERSON(x) \vee \neg PERSON(x)$
$\forall x.AGENT(x) \vee \neg AGENT(x)$
$\forall x.PERSON(x) \leftarrow pcmember(x, LPNMR) \wedge AGENT(x) \wedge \neg machine(x)$
$pcmember(David, LPNMR)$

There IS a classical model of this theory
$\mathcal{M} = \{\neg PERSON(David), machine(David), \ldots\}$

Thus:
$K \not\models_{FOL} PERSON(David)$

Example - *stable closure* of $\mathcal{K}$:

$st(\mathcal{K}) = \mathcal{T} \cup st(\mathcal{T}) \cup \mathcal{P}$

$\forall x.PERSON(x) \rightarrow AGENT(x)$
$AGENT(David)$
$\forall x.PERSON(x) \vee \neg PERSON(x)$
$\forall x.AGENT(x) \vee \neg AGENT(x)$
$\forall x.PERSON(x) \leftarrow pcmember(x, LPNMR), AGENT(x) \wedge \neg machine(x)$

$pcmember(David, LPNMR)$

The total HT-model $\mathcal{M}_{HT} = \langle H, T \rangle$ corresponding to $\mathcal{M}$ with:
$H = T = \{machine(David), \ldots\}$

is NO Equilibrium model, since there is a model $\mathcal{M}'_{HT} \lhd \mathcal{M}_{HT}$:

with: $H' = \{\ldots\}$
$T = \{machine(David), \ldots\}$

All Equilibrium models include $PERSON(David)$, thus:
$st(\mathcal{K}) \models_{QEL} PERSON(David)$ $\surd$

Example - *stable closure* of $\mathcal{K}$:

$st(\mathcal{K}) = \mathcal{T} \cup st(\mathcal{T}) \cup \mathcal{P}$

$\forall x.PERSON(x) \rightarrow AGENT(x)$
$AGENT(David)$
$\forall x.PERSON(x) \vee \neg PERSON(x)$
$\forall x.AGENT(x) \vee \neg AGENT(x)$
$\forall x.PERSON(x) \leftarrow pcmember(x, LPNMR), AGENT(x) \wedge \neg machine(x)$

$pcmember(David, LPNMR)$

The total HT-model $\mathcal{M}_{HT} = \langle H, T \rangle$ corresponding to $\mathcal{M}$ with:
$H = T = \{machine(David), \ldots\}$

is NO Equilibrium model, since there is a model $\mathcal{M}'_{HT} \lhd \mathcal{M}_{HT}$:

with: $H' = \{\ldots\}$
$T = \{machine(David), \ldots\}$

All Equilibrium models include $PERSON(David)$, thus:
$st(\mathcal{K}) \models_{QEL} PERSON(David)$ ✓

Example - *stable closure* of $\mathcal{K}$:

$st(\mathcal{K}) = \mathcal{T} \cup st(\mathcal{T}) \cup \mathcal{P}$

$\forall x.PERSON(x) \rightarrow AGENT(x)$
$AGENT(David)$
$\forall x.PERSON(x) \vee \neg PERSON(x)$
$\forall x.AGENT(x) \vee \neg AGENT(x)$
$\forall x.PERSON(x) \leftarrow pcmember(x, LPNMR), AGENT(x) \wedge \neg machine(x)$
$pcmember(David, LPNMR)$

The total HT-model $\mathcal{M}_{HT} = \langle H, T \rangle$ corresponding to $\mathcal{M}$ with:
$H = T = \{machine(David), \ldots\}$

is NO Equilibrium model, since there is a model $\mathcal{M}'_{HT} \lhd \mathcal{M}_{HT}$:

with: $H' = \{\ldots\}$
$T = \{machine(David), \ldots\}$

All Equilibrium models include $PERSON(David)$, thus:
$st(\mathcal{K}) \models_{QEL} PERSON(David)$ √

Example - *stable closure* of $\mathcal{K}$:

$st(\mathcal{K}) = \mathcal{T} \cup st(\mathcal{T}) \cup \mathcal{P}$

$\forall x.PERSON(x) \rightarrow AGENT(x)$
$AGENT(David)$
$\forall x.PERSON(x) \vee \neg PERSON(x)$
$\forall x.AGENT(x) \vee \neg AGENT(x)$
$\forall x.PERSON(x) \leftarrow pcmember(x, LPNMR), AGENT(x) \wedge \neg machine(x)$

$pcmember(David, LPNMR)$

The total HT-model $\mathcal{M}_{HT} = \langle H, T \rangle$ corresponding to $\mathcal{M}$ with:
$H = T = \{machine(David), \ldots\}$

is NO Equilibrium model, since there is a model $\mathcal{M}'_{HT} \lhd \mathcal{M}_{HT}$:

with: $H' = \{\ldots\}$
$T = \{machine(David), \ldots\}$

All Equilibrium models include $PERSON(David)$, thus:
$st(\mathcal{K}) \models_{QEL} PERSON(David)$ $\surd$

Conclusions/Observations:

- Quantified Equilibrium Logic provides a powerful and intuitive tool as a "carrier" logic for Hybrid KBs
- Embedding is simple: add LEM for classical predicates.
- Why this works is not so surprising: QHT^s based on intuitionistic logic, adding LEM enforces totalization of HT models on the respective predicates, i.e. make them "classical".
- No reducts/grounding involved, this gives us:
  - a semantics for nested logic programs. Well-investigated for propositional LPs, first-order case needs more investigation, respective results on QEL relatively new.
  - results on strong equivalence for Equilibrium Logic carry over: HT-model equivalence amounts to strong equivalence! Important notion for nonmonotonic logic programs, program optimization, meaning preserving transformations of hybrid KB's, etc.

Conclusions/Observations:

- Quantified Equilibrium Logic provides a powerful and intuitive tool as a “carrier” logic for Hybrid KBs
- Embedding is simple: add LEM for classical predicates.
- Why this works is not so surprising: QHT^s based on intuitionistic logic, adding LEM enforces totalization of HT models on the respective predicates, i.e. make them “classical”.
- No reducts/grounding involved, this gives us:
  - a semantics for nested logic programs. Well-investigated for propositional LPs, first-order case needs more investigation, respective results on QEL relatively new.
  - results on strong equivalence for Equilibrium Logic carry over: HT-model equivalence amounts to strong equivalence! Important notion for nonmonotonic logic programs, program optimization, meaning preserving transformations of hybrid KB’s, etc.

Conclusions/Observations:

- Quantified Equilibrium Logic provides a powerful and intuitive tool as a "carrier" logic for Hybrid KBs
- Embedding is simple: add LEM for classical predicates.
- Why this works is not so surprising: QHT^s based on intuitionistic logic, adding LEM enforces totalization of HT models on the respective predicates, i.e. make them "classical".
- No reducts/grounding involved, this gives us:
  - a semantics for nested logic programs. Well-investigated for propositional LPs, first-order case needs more investigation, respective results on QEL relatively new.
  - results on strong equivalence for Equilibrium Logic carry over: HT-model equivalence amounts to strong equivalence! Important notion for nonmonotonic logic programs, program optimization, meaning preserving transformations of hybrid KB's, etc.

Conclusions/Observations:

- Quantified Equilibrium Logic provides a powerful and intuitive tool as a "carrier" logic for Hybrid KBs
- Embedding is simple: add LEM for classical predicates.
- Why this works is not so surprising: QHT^s based on intuitionistic logic, adding LEM enforces totalization of HT models on the respective predicates, i.e. make them "classical".
- No reducts/grounding involved, this gives us:
  - a semantics for nested logic programs. Well-investigated for propositional LPs, first-order case needs more investigation, respective results on QEL relatively new.
  - results on strong equivalence for Equilibrium Logic carry over: HT-model equivalence amounts to strong equivalence! Important notion for nonmonotonic logic programs, program optimization, meaning preserving transformations of hybrid KB's, etc.

Conclusions/Observations:

- Quantified Equilibrium Logic provides a powerful and intuitive tool as a "carrier" logic for Hybrid KBs
- Embedding is simple: add LEM for classical predicates.
- Why this works is not so surprising: QHT^s based on intuitionistic logic, adding LEM enforces totalization of HT models on the respective predicates, i.e. make them "classical".
- No reducts/grounding involved, this gives us:
  - a semantics for nested logic programs. Well-investigated for propositional LPs, first-order case needs more investigation, respective results on QEL relatively new.
  - results on strong equivalence for Equilibrium Logic carry over: HT-model equivalence amounts to strong equivalence! Important notion for nonmonotonic logic programs, program optimization, meaning preserving transformations of hybrid KB's, etc.

Conclusions/Observations:

- Quantified Equilibrium Logic provides a powerful and intuitive tool as a "carrier" logic for Hybrid KBs
- Embedding is simple: add LEM for classical predicates.
- Why this works is not so surprising: QHT^s based on intuitionistic logic, adding LEM enforces totalization of HT models on the respective predicates, i.e. make them "classical".
- No reducts/grounding involved, this gives us:
  - a semantics for nested logic programs. Well-investigated for propositional LPs, first-order case needs more investigation, respective results on QEL relatively new.
  - results on strong equivalence for Equilibrium Logic carry over: HT-model equivalence amounts to strong equivalence! Important notion for nonmonotonic logic programs, program optimization, meaning preserving transformations of hybrid KB's, etc.

Conclusions/Observations:

- Quantified Equilibrium Logic provides a powerful and intuitive tool as a "carrier" logic for Hybrid KBs
- Embedding is simple: add LEM for classical predicates.
- Why this works is not so surprising: QHT^s based on intuitionistic logic, adding LEM enforces totalization of HT models on the respective predicates, i.e. make them "classical".
- No reducts/grounding involved, this gives us:
  - a semantics for nested logic programs. Well-investigated for propositional LPs, first-order case needs more investigation, respective results on QEL relatively new.
  - results on strong equivalence for Equilibrium Logic carry over: HT-model equivalence amounts to strong equivalence! Important notion for nonmonotonic logic programs, program optimization, meaning preserving transformations of hybrid KB's, etc.

Future Work:

- Paper covers only equality-free FOL embedding (results already there in [Pearce&Valverde, 2006])
- Investigation on related (IJCAI) approaches:
  - Logic of minimal knowledge and negation as failure (MKNF) [Motik & Rosati, 2007]
  - First-Order Autoepistemic Logic [de Bruijn et al., 2007]
  - Circumscription [Ferraris,Lee,Lifschitz, 2007]

Get the paper at: `http://polleres.net/publications.html`

Variants discussed in the paper:

	UNA	variables	disj.	neg. $\mathcal{L}_\mathcal{T}$ atoms
r-hybrid [Rosati,2005]	yes	$\mathcal{L}_\mathcal{P}$-safe	pos.	no
r^+-hybrid [Rosati,2005b]	no	$\mathcal{L}_\mathcal{P}$-safe	pos.	no
r_w-hybrid [Rosati, 2006]	yes	weak $\mathcal{L}_\mathcal{P}$-safe	pos.	no
g-hybrid [Heymans, et al. 2006]	no	guarded	neg.*	yes

* g-hybrid allows negation in the head but at most one positive head atom

Table: Different variants of hybrid KBs