Second-Order Characterizations of Definientia in Formula Classes

Christoph Wernhard
Technische Universität Dresden

Automated Reasoning Workshop/Deduktionstreffen (ARW-DT 2014)
Vienna, July 24 2014
1. Introduction
Definientia

- A **definition of** $G$ **in terms of** $S$ **within** $F$ is a formula $(G \leftrightarrow X)$ s.t.
  1. $F \models (G \leftrightarrow X)$, and
  2. $X$ contains only symbols from $S$

- $G$ is the **definiendum**
  
  $X$ is the **definiens**

- This applies also to first-order logic:
  
  If there are no free variables in $F$, then
  
  $F \models \forall x (G(x) \leftrightarrow X(x))$ iff
  
  $F \models G(x) \leftrightarrow X(x)$

- We are interested in **computing definientia** $X$ for given $F$, $G$ and $S$
An Application: Definientia as Exact View-Based Query Rewritings

[Halevy 01, Calvanese* 07, Marx 07, Nash* 10, Bárány* 13, W 14a]

- Given:
  - $DBSymbols = \{a\}$
  - $ViewSymbols = \{p, q\}$
  - $ViewSpec$ in terms of $DBSymbols \cup ViewSymbols$
  - $Query$ in terms of $DBSymbols$

- Compute a **Rewriting** of $Query$ in terms of $ViewSymbols$ s.t. for all $DB$:
  \[
  DB \land ViewSpec \models Rewriting \iff DB \models Query
  \]

- (Under certain assumptions on $ViewSpec$), the **Rewritings** are the **definientia** of $Query$ in terms of $ViewSymbols$ within $ViewSpec$

\[
\begin{align*}
(p \leftrightarrow a) \land (q \leftrightarrow a) & \models a \leftrightarrow (p \land q) \\
(p \leftrightarrow a) \land (q \leftrightarrow a) & \models a \leftrightarrow p \\
(p \leftrightarrow a) \land (q \leftrightarrow a) & \models a \leftrightarrow q \\
(p \leftrightarrow a) \land (q \leftrightarrow a) & \models a \leftrightarrow (p \lor q) \\
\end{align*}
\]

\[
\begin{align*}
a \land (p \leftrightarrow a) \land (q \leftrightarrow a) & \models p \text{ iff } a \models a \\
\neg a \land (p \leftrightarrow a) \land (q \leftrightarrow a) & \models p \text{ iff } \neg a \models a
\end{align*}
\]
Addressed Question

- Definientia in terms of a given set of predicates can be characterized semantically by second-order formulas
  - They can be computed by second-order quantifier elimination
    aka computation of forgetting and uniform interpolants
    [Doherty* 97, Gabbay and Ohlbach 92, Gabbay* 08, Ghilardi* 06, Konev* 09, Koopmann and Schmidt 13]

- It seems useful to compute definientia that are in a given formula class
  (like Horn or Krom formulas)

- “Determinacy” is investigated in database research
  [Segoufin and Vianu 05, Marx 07, Nash* 10, Bárány* 13]

For Query, ViewSpec in particular formula classes:
- is the existence of an exact rewriting (definiens) decidable?
- what formula class contains all exact rewritings (definientia)?

Can definientia in given formula classes be characterized by second-order formulas?
Two Basic Approaches

1. Characterizations based on semantic properties, such as the model intersection property for Horn formulas

2. Modeling syntactic characterizations by an auxiliary “meta-level” vocabulary

Can definientia in given formula classes be characterized by second-order formulas?
2. Toolkit: Classical Logic

+ Second-Order Operators
Classical Logic + Second-Order Operators

• We start with an **underlying classical logic**, e.g., first-order or propositional

• It is extended by **second-order operators**, e.g., predicate quantification or Boolean quantification

  \[ \exists q (p \rightarrow q) \land (q \rightarrow r) \]

For propositional logic:  
\[ \exists p F \equiv F[p \mapsto \text{TRUE}] \lor F[p \mapsto \text{FALSE}] \]

• The associated computation is **second-order operator elimination**: computing an equivalent formula without second-order operators

  \[ \exists q (p \rightarrow q) \land (q \rightarrow r) \equiv p \rightarrow r. \]
Forgetting, Projection, Uniform Interpolation

<table>
<thead>
<tr>
<th>Second-order operator</th>
<th>Elimination aka</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\exists q , F$</td>
<td>Predicate quantifier elimination</td>
</tr>
<tr>
<td></td>
<td>Boolean variable elimination</td>
</tr>
<tr>
<td>$\equiv$ forget$_{{q}}(F)$</td>
<td>Computation of forgetting</td>
</tr>
<tr>
<td>$\equiv$ project$_{{p,r}}(F)$</td>
<td>Computation of projection</td>
</tr>
<tr>
<td>$\equiv$ forget$_{\text{ALLPREDICATES} - {p,r}}(F)$</td>
<td>Uniform interpolation</td>
</tr>
<tr>
<td>$\equiv$ project$_{\text{ALLPREDICATES} - {q}}(F)$</td>
<td></td>
</tr>
</tbody>
</table>


Considering Polarity: Literal Forgetting, Literal Projection

[Lang* 03, W 08]

• We generalize the first argument of forgetting and projection to a set of ground literals, called scope

Effects on just positive or negative predicate occurrences can be expressed

**Literal forgetting** and **literal projection** are now our basic operators

Let \( F = (p \rightarrow q) \land (q \rightarrow r) \)

\[
\begin{align*}
\text{forget}_{\{-q\}}(F) & \equiv \text{project}_{\{p,q,r,\neg p,\neg r\}}(F) \\
& \equiv (p \rightarrow q) \land (p \rightarrow r)
\end{align*}
\]

An interpretation is a set of ground literals, containing each ground atom either positively or negatively.

\( I \models \text{project}_S(F) \quad \text{iff def} \quad \text{There exists a } J \text{ s.t. } J \models F \text{ and } J \cap S \subseteq I. \)

\( \text{forget}_S(F) \quad \text{def} \quad \text{project}_{\text{ALLGROUNDLITERALS} \setminus S}(F). \)
Scope-Determined Circumscription

- Interpretations can be **partially ordered** according to the subset relationship between the set of ground atoms that they satisfy

\[
\{p(a), \neg p(b), \neg q(a), \neg q(b)\} \leq \{p(a), p(b), q(a), \neg q(b)\}
\]

- **Predicate circumscription** allows to characterize the set of models of a formula that are **minimal** w.r.t. this ordering and generalizations where
  
  - only extensions of specified predicates are compared
  - comparison requires that extensions of specified predicates are equal

  [McCarthy 80, Lifschitz 94, Doherty* 97]

- The **second-order operator** \(\text{circ}_S(F)\) can express these variations, generalized to model maximization [W 12]

\[
\text{circ}_{\{p,q\}}(p \lor q) \equiv (p \land \neg q) \lor (q \land \neg p)
\]

\[
I \models \text{project}_S(F) \iff \text{def} \quad \text{There exists a } J \text{ s.t. } J \models F \text{ and } J \cap S \subseteq I.
\]

\[
I \models \text{raise}_S(F) \iff \text{def} \quad \text{There exists a } J \text{ s.t. } J \models F \text{ and } J \cap S \subset I \cap S.
\]

\[
\text{circ}_S(F) \quad \text{def} \quad F \land \neg \text{raise}_S(F).
\]
Notation for Aboutness

- That $F$ is "about" $S$, or "in scope" $S$ is written

$$ F \subseteq S $$

Let $F = p \lor \neg q \lor (r \land \neg r)$

$$
F \subseteq \{p, \neg q\} \\
F \subseteq \{p, q, r, s, \neg p, \neg q, \neg r, \neg s\} \\
F \not\subseteq \{p\}
$$

$$ F \subseteq S \iff_{def} F \equiv \text{project}_S(F). $$
Globally Strongest Necessary and Weakest Sufficient Condition

- The **globally strongest necessary condition** of $G$ on $S$ within $F$ is the strongest $X \subseteq S$ s.th. $(F \land G) \models X$
  It can be expressed by a second-order operator

\[
\text{gsnc}_{\{p\}}((q \rightarrow p), q) \equiv p
\]

- The **globally weakest sufficient condition** of $G$ on $S$ within $F$ is the weakest $X \subseteq S$ s.th. $(F \land X) \models G$
  It can be expressed by a second-order operator

\[
\text{gwsc}_{\{p\}}((p \rightarrow q), q) \equiv p
\]

- The analog concepts in [Lin 01] are not unique modulo equivalence. See also [Doherty* 01, W 12]

Let $\overline{S}$ denote the set of the complements of the members of scope $S$.

\[
\text{gsnc}_S(F, G) \overset{\text{def}}{=} \text{project}_S(F \land G).
\]

\[
\text{gwsc}_S(F, G) \overset{\text{def}}{=} \neg \text{project}_{\overline{S}}(F \land \neg G).
\]
Definientia, Definability in Terms of Second-Order Operators

Recall: A definition of $G$ in terms of $S$ within $F$ is a formula $(G \leftrightarrow X)$ s.t.
(1.) $X \in S$, and (2.) $F \models G \leftrightarrow X$. $G$ is the definiendum, $X$ is the definiens.

- **Definientia** are exactly those formulas in the scope that are between the GSNC and the GWSC.

Let $F = (p \leftrightarrow a) \land (q \leftrightarrow a)$, let $S = \{p, q, \neg p, \neg q\}$

$$\text{gsnc}_S(F, a) \equiv (p \land q) \models p \models q \models (p \lor q) \equiv \text{gwsc}_S(F, a)$$

- **Definability** (existence of a definiens) holds iff the GSNC entails the GWSC.

**Definitions:**

- $\text{ISDEFINIENS}(X, G, S, F') \iff \text{def } X \in S \text{ and } \text{gsnc}_S(F, G) \models X \models \text{gwsc}_S(F, G)$.
- $\text{ISDEFINABLE}(G, S, F') \iff \text{def } \text{gsnc}_S(F, G) \models \text{gwsc}_S(F, G)$.
- $\text{gsnc}_S(F, G) \overset{\text{def}}{=} \text{project}_S(F \land G)$.
- $\text{gwsc}_S(F, G) \overset{\text{def}}{=} \neg \text{project}_S(F \land \neg G)$. 

14
So Far we Have:

- Second-order operators for
  
  literal forgetting \( \text{forget}_S(F) \)
  
  literal projection \( \text{project}_S(F) \)
  
  predicate circumscription \( \text{circ}_S(F) \)
  
  globally strongest necessary condition \( \text{gsnc}_S(F, G) \)
  
  globally weakest sufficient condition \( \text{gwsc}_S(F, G) \)

- Characterizations in terms of second-order operations for
  
  aboutness \( F \subseteq S \)
  
  definiens \( \text{ISDEFINIENS}(X, G, S, F) \)
  
  definability \( \text{ISDEFINABLE}(G, S, F) \)
3. Horn Formulas and Horn Upper Bounds
Horn Formulas and the Least Horn Upper Bound

- The least Horn upper bound of a given formula is the strongest Horn formula that is weaker than or equivalent to the given formula [Selman and Kautz 91, Kautz* 95]

- It is unique up to equivalence

- It is equivalent to the conjunction of all prime implicates that are Horn [Selman and Kautz 91]

- Let $\text{lhub}(F)$ denote the least Horn upper bound of $F$

- $\text{lhub}(F)$ can characterized semantically as the strongest formula that is weaker than or equivalent to $F$ and closed under “model intersection” [McKinsey 43, Dechter and Pearl 92]

Closure under “model intersection” can be characterized by predicate quantification, but $\text{lhub}(F)$ seems to require further means
“Filled” Horn Upper Bound

[W 14b]

- Let $f_{hub}(F)$ denote the “filled” Horn upper bound of $F$, another (possibly weaker) unique Horn upper bound.

Let $F = p \land (q \rightarrow r) \land (s \lor t) \land \neg u$

$l_{hub}(F) \equiv p \land (q \rightarrow r) \land \neg u$

$f_{hub}(F) \equiv p \land \neg u$

- It can be characterized just in terms of predicate quantification, involving a second-order operator $\text{diff}_S(F)$.

- The set of models of $f_{hub}(F)$, so-to-speak, completely “fills” the space “between” the greatest lower bound and the models of $F$.

$I \models \text{project}_S(F)$ \iff_{def} \text{There exists a } J \text{ s.t. } J \models F \text{ and } J \cap S \subseteq I.$

$I \models \text{diff}_S(F)$ \iff_{def} \text{There exists a } J \text{ s.t. } J \models F \text{ and } J \cap S \not\subseteq I.$

$\text{glb}(F)$ \iff_{def} \text{circ}_{\neg \text{diff}_{\neg \text{neg}}}(F)$.\ 

$f_{hub}(F)$ \iff_{def} \text{project}_{\text{pos}}(\text{glb}(F)) \land \text{project}_{\text{neg}}(F)$.\ 

18
Illustration: Horn Upper Bounds

Formula $F$
Illustration: Horn Upper Bounds

- formula $F$
- model intersection step $\text{im}(F)$
- greatest lower bound $\text{glb}(F)$
- least Horn upper bound $\text{lhub}(F)$
- filled Horn upper bound $\text{fhub}(F)$
Illustration: Horn Upper Bounds

- Formula $F$
- Model intersection step $\text{im}(F)$
- Least Horn upper bound $\text{im} (\text{im}(F)) = \text{lhub}(F)$
Illustration: Horn Upper Bounds

- Formula
- Model intersection step
- Least Horn upper bound
- Greatest lower bound

\[ F \cup \text{im}(F) = \text{lhub}(F) \]
\[ \text{glb}(F') \]
Illustration: Horn Upper Bounds

Formula

- Model intersection step: \( \text{im}(F) \)
- Least Horn upper bound: \( \text{im}(\text{im}(F)) = \text{lhub}(F) \)
- Greatest lower bound: \( \text{glb}(F') \)
- Filled Horn upper bound: \( \text{fhub}(F') \)
4. Expressing Definientia in Formula Classes
Considered Formula Classes and Shown Properties

- We consider the following formula classes:
  - **S**H**O**R**N** Formulas equivalent to a **Horn** formula
  - **S**C**O**N**A**T**M** Formulas equivalent to a **conjunction of atoms**
  - **S**K**R**OM Formulas equivalent to a **Krom** formula

- A definiens in such a class $C$ is called $C$-definiens

- For each of these formula classes $C$ we show
  - a **characterization of $C$-definability**, that is, existence of a $C$-definiens
  - a **representative $C$-definiens**, that is, a (second-order) formula which is a $C$-definiens under the sole precondition of $C$-definability

[W 14b]
Definability and Definientia: SHORN

- Definability: *The least Horn upper bound of the GSNC entails the GWSC*
- Representative definiens: *The least Horn upper bound of the GSNC*

**Theorem:** $G$ is $\mathcal{C}$-definable in terms of $S$ within $F$ iff

For $\mathcal{C} = \text{SHORN}$: $\text{lhub}(\text{gsnc}_S(F, G)) \models \text{gwsc}_S(F, G)$.

For $\mathcal{C} = \text{SCONATM}$: $\text{glb}(\text{gsnc}_{S \cap \text{POS}}(F, G)) \models \text{gwsc}_{S \cap \text{POS}}(F, G)$.

For $\mathcal{C} = \text{SKROM}$: $G$ is SCONATM-definable in terms of $\text{KS}(S)$ within $(F \land \text{KD}(S))$.

**Theorem:** If $G$ is $\mathcal{C}$-definable in terms of $S$ within $F$, then the following formula is a $\mathcal{C}$-definiens:

For $\mathcal{C} = \text{SHORN}$: $\text{lhub}(\text{gsnc}_S(F, G))$.

For $\mathcal{C} = \text{SCONATM}$: $\text{fhub}(\text{gsnc}_{S \cap \text{POS}}(F, G))$.

For $\mathcal{C} = \text{SKROM}$: $\text{project}_S(\text{fhub}(\text{gsnc}_{\text{KS}(S)}(F \land \text{KD}(S), G)) \land \text{KD}(S))$.
Definability and Definientia: SConATM

- Here we consider the GSNC and the GWSC on the set of the positive literals in the specified scope
- Definability: The greatest lower bound of the GSNC entails the GWSC
- Representative definiens: The filled Horn upper bound of the GSNC
- Expressed just by the introduced second-order operators, which in turn are reducible to predicate quantification

**Theorem:** $G$ is $\mathcal{C}$-definable in terms of $S$ within $F$ iff
- For $\mathcal{C} = \text{SHORN}$: $\text{lhub}(\text{gsnc}_S(F, G)) \models \text{gwsc}_S(F, G)$.
- For $\mathcal{C} = \text{SConATM}$: $\text{glb}(\text{gsnc}_{S \cap \text{POS}}(F, G)) \models \text{gwsc}_{S \cap \text{POS}}(F, G)$.
- For $\mathcal{C} = \text{SKROM}$: $G$ is SConATM-definable in terms of $\text{KS}(S)$ within $(F \land \text{KD}(S))$.

**Theorem:** If $G$ is $\mathcal{C}$-definable in terms of $S$ within $F$, then the following formula is a $\mathcal{C}$-definiens:
- For $\mathcal{C} = \text{SHORN}$: $\text{lhub}(\text{gsnc}_S(F, G))$.
- For $\mathcal{C} = \text{SConATM}$: $\text{fhub}(\text{gsnc}_{S \cap \text{POS}}(F, G))$.
- For $\mathcal{C} = \text{SKROM}$: $\text{project}_S(\text{fhub}(\text{gsnc}_{\text{KS}(S)}(F \land \text{KD}(S), G)) \land \text{KD}(S)))$
Modeling Syntactic Characterizations by a Meta-Level Vocabulary

- Idea:
  1. define **“meta-level” symbols** for expressions
  2. restrict the “meta-level” symbols allowed in definientia

  Problem: Arbitrary combinations of **disjunctions** and **negations** of formulas would meet such restrictions

- **Negation and disjunction can be excluded with SCONATM-definientia**

**Theorem:** $G$ is $\mathcal{C}$-definable in terms of $S$ within $F$ iff

For $\mathcal{C} = \text{SHORN}$: $\text{lhub}(\text{gsnc}_S(F, G)) \models \text{gwsc}_S(F, G)$.

For $\mathcal{C} = \text{SCONATM}$: $\text{glb}(\text{gsnc}_{S\cap\text{POS}}(F, G)) \models \text{gwsc}_{S\cap\text{POS}}(F, G)$.

For $\mathcal{C} = \text{SKROM}$: $G$ is SCONATM-definable in terms of $\text{KS}(S)$ within $(F \land \text{KD}(S))$.

**Theorem:** If $G$ is $\mathcal{C}$-definable in terms of $S$ within $F$, then the following formula is a $\mathcal{C}$-definiens:

For $\mathcal{C} = \text{SHORN}$: $\text{lhub}(\text{gsnc}_S(F, G))$.

For $\mathcal{C} = \text{SCONATM}$: $\text{fhub}(\text{gsnc}_{S\cap\text{POS}}(F, G))$.

For $\mathcal{C} = \text{SKROM}$: $\text{project}_S(\text{fhub}(\text{gsnc}_{\text{KS}(S)}(F \land \text{KD}(S), G)) \land \text{KD}(S))$
Definability and Definientia: $SKROM$ – Auxiliary Formula

- $KD(S)$ is the conjunction of the definitions of the “meta-level” atoms
  - $empty$, representing the empty clause, and
  - $clause(L, M)$, representing nonempty Krom clauses
- $KS(S)$ are the positive literals with the “meta-level” atoms

Assume a fixed total order $\leq$ on literals. Define:

$$KD(S) \overset{\text{def}}{=} (\text{empty} \leftrightarrow \bot) \land \bigwedge_{L, M \in S, \; L \leq M, \; L \neq \overline{M}} (\text{clause}(L, M) \leftrightarrow L \lor M).$$

$$KS(S) \overset{\text{def}}{=} \{\text{empty}\} \cup \{\text{clause}(L, M) \mid L, M \in S, \; L \leq M, \; L \neq \overline{M}\}.$$

**Theorem:** $G$ is $C$-definable in terms of $S$ within $F$ iff

For $C = SHORN$: $\text{lhub}(gsnc}_S(F, G)) \models gwsc}_S(F, G)$.

For $C = SCONATM$: $\text{glb}(gsnc}_{S \cap \text{POS}}(F, G)) \models gwsc}_{S \cap \text{POS}}(F, G)$.

For $C = SKROM$: $G$ is SCONATM-definable in terms of $KS(S)$ within $(F \land KD(S))$.

**Theorem:** If $G$ is $C$-definable in terms of $S$ within $F$, then the following formula is a $C$-definiens:

For $C = SHORN$: $\text{lhub}(gsnc}_S(F, G))$.

For $C = SCONATM$: $\text{fhub}(gsnc}_{S \cap \text{POS}}(F, G))$.

For $C = SKROM$: $\text{project}_S(\text{fhub}(gsnc}_{KS(S)}(F \land KD(S), G)) \land KD(S))$
Definability and Definientia: SKROM

- Definability: *Definable by a conjunction of the “meta-level” atoms*

- Representative definiens: *Take the representative definiens as conjunction of the “meta-level” atoms and convert it by projection to the original vocabulary*

**Theorem:** $G$ is $\mathcal{C}$-definable in terms of $S$ within $F$ iff

For $\mathcal{C} = \text{SHORN}$: $\text{lhub}(\text{gsnc}_S(F, G)) \models \text{gwsc}_S(F, G)$.

For $\mathcal{C} = \text{SCONATM}$: $\text{glb}(\text{gsnc}_{S \cap \text{POS}}(F, G)) \models \text{gwsc}_{S \cap \text{POS}}(F, G)$.

For $\mathcal{C} = \text{SKROM}$: $G$ is SCONATM-definable in terms of $\text{KS}(S)$ within $(F \land \text{KD}(S))$.

**Theorem:** If $G$ is $\mathcal{C}$-definable in terms of $S$ within $F$, then the following formula is a $\mathcal{C}$-definiens:

For $\mathcal{C} = \text{SHORN}$: $\text{lhub}(\text{gsnc}_S(F, G))$.

For $\mathcal{C} = \text{SCONATM}$: $\text{fhub}(\text{gsnc}_{S \cap \text{POS}}(F, G))$.

For $\mathcal{C} = \text{SKROM}$: $\text{project}_S(\text{fhub}(\text{gsnc}_{\text{KS}(S)}(F \land \text{KD}(S), G)) \land \text{KD}(S))$.
5. Conclusion
Open(ed) Issues

Q1: How to **express the least Horn upper bound** – fixpoint extension?

Q2: What **first-order formula classes** correspond to semantic properties like closure under model intersection?

Q3: What further **formula classes and properties** can be handled?

Q4: Are there useful **properties of the second-order expressions** characterizing definability and definientia, also with respect to arguments from specific classes?

Q5: Are there relationships to works on **non-uniform interpolation**?

Q6: Can the characterizations be applied with **approximations** like GSNC and GWSC instead of definitions?

Q7: What about **computation** of definientia? Layers involved:
   - manipulation on the operator level
   - eliminating the second-order operators
   - conversion to the actual syntactic form

Q8: Can the approach be **practically** used?
   - implemented with *ToyElim* [W 13], suitable for tiny experiments
   - support for nested forgetting missing in current DL systems
Summary

• Steps towards a **formalized** and **mechanizable** bridge between
  • **expressibility in formula classes** and
  • **expressibility in restricted vocabularies**, formulated essentially by predicate quantification

• Demonstration with propositional logic as basis for **conjunctions of atoms**, **Krom** formulas, and to some degree for **Horn** formulas
Appendix
Example: $S\text{CONATM-Definientia}$

- Let $F = (q \rightarrow r \lor s) \land (t \rightarrow q) \land ((r \lor s) \land u \rightarrow p) \land (p \rightarrow t \land u)$

- Consider finding definientia of $p$ within $F$, in terms of positive occurrences of the other atoms $S = S \cap \text{POS} = \{q, r, s, t, u\}$

- Then:
  \[ \text{gsnc}_S(F, p) \equiv q \land t \land u \land (r \lor s). \]
  \[ \text{gwsc}_S(F, p) \equiv u \land (q \lor r \lor s \lor t). \]

- None of both is equivalent to a conjunction of atoms

- By the theorem, there must exist a $S\text{CONATM}$-definiens:
  \[ \text{glb}(\text{gsnc}_S(F, p)) \equiv (q \land t \land u \land \neg p \land \neg r \land \neg s) \models \text{gwsc}_S(F, p) \]

- By the theorem, $\text{fhub}(\text{gsnc}_S(F, p))$ is a $S\text{CONATM}$-definiens:
  \[ \text{gsnc}_S(F, p) \models \text{fhub}(\text{gsnc}_S(F, p)) \equiv (q \land t \land u) \models \text{gwsc}_S(F, p) \]
References
[Bárány* 13] Bárány, V., Benedikt, M., and ten Cate, B.
Rewriting guarded negation queries.

View-based query processing: On the relationship between rewriting, answering and losslessness.

[Dechter and Pearl 92] Dechter, R. and Pearl, J.
Structure identification in relational data.

[Doherty* 97] Doherty, P., Łukaszewicz, W., and Szałas, A.
Computing circumscription revisited: A reduction algorithm.

[Doherty* 01] Doherty, P., Łukaszewicz, W., and Szałas, A.
Computing strongest necessary and weakest sufficient conditions of first-order formulas.
Quantifier elimination in second-order predicate logic.

*Second-Order Quantifier Elimination: Foundations, Computational Aspects and Applications.*

Did I damage my ontology? A case for conservative extensions in description logics.

[Halevy 01] Halevy, A. Y.
Answering queries using views: a survey.

[Kautz* 95] Kautz, H., Kearns, M., and Selman, B.
Horn approximations of empirical data.
Forgetting and uniform interpolation in large-scale description logic terminologies.

Uniform interpolation of \(\mathcal{ALC}\)-ontologies using fixpoints.

[Lang* 03] Lang, J., Liberatore, P., and Marquis, P.
Propositional independence – formula-variable independence and forgetting.

[Lifschitz 94] Lifschitz, V.
Circumscription.

[Lin 01] Lin, F.
On strongest necessary and weakest sufficient conditions.
[Marx 07] Marx, M.
Queries determined by views: pack your views.

[McCarthy 80] McCarthy, J.
Circumscription – a form of non-monotonic reasoning.

The decision problem for some classes of sentences without quantifiers.
*JSL*, 8:61–76 (1943)

Views and queries: Determinacy and rewriting.
*TODS*, 35(3) (2010)

[Segoufin and Vianu 05] Segoufin, L. and Vianu, V.
Views and queries: Determinacy and rewriting.
[Selman and Kautz 91] Selman, B. and Kautz, H. A.
Knowledge compilation using Horn approximations.

[Tarski 35] Tarski, A.
Einige methologische Untersuchungen zur Definierbarkeit der Begriffe.
Erkenntnis, 5:80–100 (1935)

[W 08] Wernhard, C.
Literal projection for first-order logic.

[W 12] Wernhard, C.
Projection and scope-determined circumscription.
JSC, 47(9):1089–1108 (2012)

Computing with logic as operator elimination: The ToyElim system.
[W 14a] Wernhard, C.
Expressing view-based query processing and related approaches with second-order operators.

[W 14b] Wernhard, C.
Second-order characterizations of definientia in formula classes.