Rewriting the Description Logic $\mathcal{ALCHIQ}$ to Disjunctive Existential Rules

David Carral and Markus Krötzsch

Knowledge-Based Systems Group

TECHNISCHE
UNIVERSITÄT
DRESDEN

February 10, 2021

Link to the paper:
iccl.inf.tu-dresden.de/web/
Inproceedings3244/en

Rewritings

Definition

Consider fragments $\mathcal{L}$ and $\mathcal{L}'$ of FOL. An $\mathcal{L}'$-theory $\mathcal{T}'$ is a *rewriting* of an $\mathcal{L}$-theory $\mathcal{T}$ if, for all fact sets $\mathcal{F}$ over the signature of $\mathcal{T}$, we have that $\mathcal{T} \cup \mathcal{F}$ and $\mathcal{T}' \cup \mathcal{F}$ are equisatisfiable.
If we can always compute such a rewriting, $\mathcal{L}$ is *rewritable* to $\mathcal{L}'$.

Definition

Consider fragments $\mathcal{L}$ and $\mathcal{L}'$ of FOL. An $\mathcal{L}'$-theory $\mathcal{T}'$ is a *rewriting* of an $\mathcal{L}$-theory $\mathcal{T}$ if, for all fact sets $\mathcal{F}$ over the signature of $\mathcal{T}$, we have that $\mathcal{T} \cup \mathcal{F}$ and $\mathcal{T}' \cup \mathcal{F}$ are equisatisfiable.
If we can always compute such a rewriting, $\mathcal{L}$ is *rewritable* to $\mathcal{L}'$.

Motivation

Rewritings

Definition

Consider fragments $\mathcal{L}$ and $\mathcal{L}'$ of FOL. An $\mathcal{L}'$-theory $\mathcal{T}'$ is a *rewriting* of an $\mathcal{L}$-theory $\mathcal{T}$ if, for all fact sets $\mathcal{F}$ over the signature of $\mathcal{T}$, we have that $\mathcal{T} \cup \mathcal{F}$ and $\mathcal{T}' \cup \mathcal{F}$ are equisatisfiable.
If we can always compute such a rewriting, $\mathcal{L}$ is *rewritable* to $\mathcal{L}'$.

Motivation

- Theoretical: understand the expressivity of FOL fragments.

Rewritings

Definition

Consider fragments $\mathcal{L}$ and $\mathcal{L}'$ of FOL. An $\mathcal{L}'$-theory $\mathcal{T}'$ is a *rewriting* of an $\mathcal{L}$-theory $\mathcal{T}$ if, for all fact sets $\mathcal{F}$ over the signature of $\mathcal{T}$, we have that $\mathcal{T} \cup \mathcal{F}$ and $\mathcal{T}' \cup \mathcal{F}$ are equisatisfiable.
If we can always compute such a rewriting, $\mathcal{L}$ is *rewritable* to $\mathcal{L}'$.

Motivation

- Theoretical: understand the expressivity of FOL fragments.
- Practical: reuse existing reasoners across FOL fragments.

Definition

Consider fragments $\mathcal{L}$ and $\mathcal{L}'$ of FOL. An $\mathcal{L}'$-theory $\mathcal{T}'$ is a *rewriting* of an $\mathcal{L}$-theory $\mathcal{T}$ if, for all fact sets $\mathcal{F}$ over the signature of $\mathcal{T}$, we have that $\mathcal{T} \cup \mathcal{F}$ and $\mathcal{T}' \cup \mathcal{F}$ are equisatisfiable.
If we can always compute such a rewriting, $\mathcal{L}$ is *rewritable* to $\mathcal{L}'$.

Motivation

- Theoretical: understand the expressivity of FOL fragments.
- Practical: reuse existing reasoners across FOL fragments.
  - Assume that $\mathcal{L}$ is rewritable to $\mathcal{L}'$.

Definition

Consider fragments $\mathcal{L}$ and $\mathcal{L}'$ of FOL. An $\mathcal{L}'$-theory $\mathcal{T}'$ is a *rewriting* of an $\mathcal{L}$-theory $\mathcal{T}$ if, for all fact sets $\mathcal{F}$ over the signature of $\mathcal{T}$, we have that $\mathcal{T} \cup \mathcal{F}$ and $\mathcal{T}' \cup \mathcal{F}$ are equisatisfiable.
If we can always compute such a rewriting, $\mathcal{L}$ is *rewritable* to $\mathcal{L}'$.

Motivation

- Theoretical: understand the expressivity of FOL fragments.
- Practical: reuse existing reasoners across FOL fragments.
  - Assume that $\mathcal{L}$ is rewritable to $\mathcal{L}'$.
  - Consider $\mathcal{T} \cup \mathcal{F}$ with $\mathcal{T}$ an $\mathcal{L}$-theory and $\mathcal{F}$ a fact set.

Rewritings

Definition

Consider fragments $\mathcal{L}$ and $\mathcal{L}'$ of FOL. An $\mathcal{L}'$-theory $\mathcal{T}'$ is a *rewriting* of an $\mathcal{L}$-theory $\mathcal{T}$ if, for all fact sets $\mathcal{F}$ over the signature of $\mathcal{T}$, we have that $\mathcal{T} \cup \mathcal{F}$ and $\mathcal{T}' \cup \mathcal{F}$ are equisatisfiable.
If we can always compute such a rewriting, $\mathcal{L}$ is *rewritable* to $\mathcal{L}'$.

Motivation

- Theoretical: understand the expressivity of FOL fragments.
- Practical: reuse existing reasoners across FOL fragments.
  - Assume that $\mathcal{L}$ is rewritable to $\mathcal{L}'$.
  - Consider $\mathcal{T} \cup \mathcal{F}$ with $\mathcal{T}$ an $\mathcal{L}$-theory and $\mathcal{F}$ a fact set.
  - Compute an equisatisfiable theory $\mathcal{T}' \cup \mathcal{F}$ with $\mathcal{T}' \in \mathcal{L}'$.

Rewritings

Definition

Consider fragments $\mathcal{L}$ and $\mathcal{L}'$ of FOL. An $\mathcal{L}'$-theory $\mathcal{T}'$ is a *rewriting* of an $\mathcal{L}$-theory $\mathcal{T}$ if, for all fact sets $\mathcal{F}$ over the signature of $\mathcal{T}$, we have that $\mathcal{T} \cup \mathcal{F}$ and $\mathcal{T}' \cup \mathcal{F}$ are equisatisfiable.
If we can always compute such a rewriting, $\mathcal{L}$ is *rewritable* to $\mathcal{L}'$.

Motivation

- Theoretical: understand the expressivity of FOL fragments.
- Practical: reuse existing reasoners across FOL fragments.
  - Assume that $\mathcal{L}$ is rewritable to $\mathcal{L}'$.
  - Consider $\mathcal{T} \cup \mathcal{F}$ with $\mathcal{T}$ an $\mathcal{L}$-theory and $\mathcal{F}$ a fact set.
  - Compute an equisatisfiable theory $\mathcal{T}' \cup \mathcal{F}$ with $\mathcal{T}' \in \mathcal{L}'$.
  - Use an $\mathcal{L}'$-reasoner to decide if $\mathcal{T}' \cup \mathcal{F}$ is satisfiable.

Rewritings

Definition

Consider fragments $\mathcal{L}$ and $\mathcal{L}'$ of FOL. An $\mathcal{L}'$-theory $\mathcal{T}'$ is a *rewriting* of an $\mathcal{L}$-theory $\mathcal{T}$ if, for all fact sets $\mathcal{F}$ over the signature of $\mathcal{T}$, we have that $\mathcal{T} \cup \mathcal{F}$ and $\mathcal{T}' \cup \mathcal{F}$ are equisatisfiable.
If we can always compute such a rewriting, $\mathcal{L}$ is *rewritable* to $\mathcal{L}'$.

Motivation

- Theoretical: understand the expressivity of FOL fragments.
- Practical: reuse existing reasoners across FOL fragments.
  - Assume that $\mathcal{L}$ is rewritable to $\mathcal{L}'$.
  - Consider $\mathcal{T} \cup \mathcal{F}$ with $\mathcal{T}$ an $\mathcal{L}$-theory and $\mathcal{F}$ a fact set.
  - Compute an equisatisfiable theory $\mathcal{T}' \cup \mathcal{F}$ with $\mathcal{T}' \in \mathcal{L}'$.
  - Use an $\mathcal{L}'$-reasoner to decide if $\mathcal{T}' \cup \mathcal{F}$ is satisfiable.
  - The result determines whether $\mathcal{T} \cup \mathcal{F}$ is satisfiable.

Rewritings

Definition

Consider fragments $\mathcal{L}$ and $\mathcal{L}'$ of FOL. An $\mathcal{L}'$-theory $\mathcal{T}'$ is a *rewriting* of an $\mathcal{L}$-theory $\mathcal{T}$ if, for all fact sets $\mathcal{F}$ over the signature of $\mathcal{T}$, we have that $\mathcal{T} \cup \mathcal{F}$ and $\mathcal{T}' \cup \mathcal{F}$ are equisatisfiable.
If we can always compute such a rewriting, $\mathcal{L}$ is *rewritable* to $\mathcal{L}'$.

Motivation

- Theoretical: understand the expressivity of FOL fragments.
- Practical: reuse existing reasoners across FOL fragments.

Contribution

Rewritings

Definition

Consider fragments $\mathcal{L}$ and $\mathcal{L}'$ of FOL. An $\mathcal{L}'$-theory $\mathcal{T}'$ is a *rewriting* of an $\mathcal{L}$-theory $\mathcal{T}$ if, for all fact sets $\mathcal{F}$ over the signature of $\mathcal{T}$, we have that $\mathcal{T} \cup \mathcal{F}$ and $\mathcal{T}' \cup \mathcal{F}$ are equisatisfiable.
If we can always compute such a rewriting, $\mathcal{L}$ is *rewritable* to $\mathcal{L}'$.

Motivation

- Theoretical: understand the expressivity of FOL fragments.
- Practical: reuse existing reasoners across FOL fragments.

Contribution

Establish that $\mathcal{ALCHIQ}$ is rewritable into rule-based languages.

The DL $\mathcal{ALCHIQ}$: Syntax and Semantics

Definition: $\mathcal{ALCHIQ}$

$$A \sqcap B \sqsubseteq C \qquad A(x) \wedge B(x) \rightarrow C(x)$$

$$A \sqsubseteq B \sqcup C \qquad A(x) \rightarrow B(x) \vee C(x)$$

$$A \sqsubseteq \forall R.B \qquad A(x) \wedge R(x,y) \rightarrow B(y)$$

$$A \sqsubseteq \exists R.B \qquad A(x) \rightarrow \exists y.R(x,y) \wedge B(y)$$

$$A \sqsubseteq \leqslant 1\, R.B \qquad A(x) \wedge R(x,y) \wedge B(y) \wedge R(x,z) \wedge B(z) \rightarrow y \approx z$$

$$R \sqcap S \sqsubseteq V \qquad R(x,y) \wedge S(x,y) \rightarrow V(x,y)$$

$$R \sqsubseteq S \sqcup V \qquad R(x,y) \rightarrow S(x,y) \vee V(x,y)$$

$$R^{-} \sqsubseteq S \qquad R(y,x) \rightarrow S(x,y)$$

In the above, A, B, and C are unary predicates (i.e., concept names) and R, S, and V are binary predicates (i.e., role names)

Datalog$^{\vee\exists}$: Syntax and Semantics

Definition

A *disjunctive existential rule* is a FOL formula of the form

$$\forall\vec{x}.\Big(\beta[\vec{x}] \to \bigvee_{i=1}^{n} \exists\vec{y}_i.\eta_i[\vec{x}_i, \vec{y}_i]\Big).$$

where $\beta[\vec{x}]$ and $\eta_i[\vec{x}_i, \vec{y}_i]$ are atom conjunctions using variables in the lists $\vec{x}_{(i)}$ and $\vec{y}_i$, such that $\vec{x}_i \subseteq \vec{x}$ and $\vec{x} \cap \vec{y}_i = \emptyset$ for all $1 \leq i \leq n$.

Datalog$^{\vee\exists}$: Syntax and Semantics

Definition

A *disjunctive existential rule* is a FOL formula of the form

$$\forall \vec{x}. \Big(\beta[\vec{x}] \to \bigvee\nolimits_{i=1}^{n} \exists \vec{y}_i . \eta_i[\vec{x}_i, \vec{y}_i] \Big).$$

where $\beta[\vec{x}]$ and $\eta_i[\vec{x}_i, \vec{y}_i]$ are atom conjunctions using variables in the lists $\vec{x}_{(i)}$ and $\vec{y}_i$, such that $\vec{x}_i \subseteq \vec{x}$ and $\vec{x} \cap \vec{y}_i = \emptyset$ for all $1 \leq i \leq n$.

Definition

- Datalog$^{\vee\exists}$: all sets of disjunctive existential rules.
- *Datalog*$^{\exists}$: Datalog$^{\vee\exists}$ without disjunction.
- *Datalog*$^{\vee}$: Datalog$^{\vee\exists}$ without existential quantifiers.
- *Datalog*: Datalog$^{\vee}$ without disjunctions.

[Hustadt et al., 2007]	$\mathcal{ALCHIQ}$	Datalog$^\vee$	exp.	†
[Eiter et al., 2012]	Horn-$\mathcal{SHIQ}$	Datalog	exp.	†
[Rudolph et al., 2012]	$\mathcal{SHIQ}b_s$	Datalog$^\vee$	exp.	†
[Bienvenu et al., 2014]	$\mathcal{SHI}$	Datalog$^\vee$	exp.	†
[Carral et al., 2018]	Horn-$\mathcal{ALCHOIQ}$	Datalog	exp.	†
[Carral et al., 2019b]	Horn-$\mathcal{SHIQ}$	Datalog	exp.	†
[Ortiz et al., 2010]	Horn-$\mathcal{ALCHOIQ}$	Datalog	poly.	
[Ahmetaj et al., 2016]	$\mathcal{ALCHIO}$	Datalog$^\vee$	poly.	
[Krötzsch, 2011]	$\mathcal{EL}^{++}$	Datalog	poly.	†
[Carral et al., 2019a]	Horn-$\mathcal{ALC}$	Datalog$^\exists$	poly.	†

†: rules of bounded size that does not depend on input

Rewritings of DL-Type Logics to Rule Languages

[Hustadt et al., 2007]	$\mathcal{ALCHIQ}$	Datalog$^{\vee}$	exp.	†
[Eiter et al., 2012]	Horn-$\mathcal{SHIQ}$	Datalog	exp.	†
[Rudolph et al., 2012]	$\mathcal{SHIQ}b_s$	Datalog$^{\vee}$	exp.	†
[Bienvenu et al., 2014]	$\mathcal{SHI}$	Datalog$^{\vee}$	exp.	†
[Carral et al., 2018]	Horn-$\mathcal{ALCHOIQ}$	Datalog	exp.	†
[Carral et al., 2019b]	Horn-$\mathcal{SHIQ}$	Datalog	exp.	†
[Ortiz et al., 2010]	Horn-$\mathcal{ALCHOIQ}$	Datalog	poly.	
[Ahmetaj et al., 2016]	$\mathcal{ALCHIO}$	Datalog$^{\vee}$	poly.	
[Krötzsch, 2011]	$\mathcal{EL}^{++}$	Datalog	poly.	†
[Carral et al., 2019a]	Horn-$\mathcal{ALC}$	Datalog$^{\exists}$	poly.	†

†: rules of bounded size that does not depend on input

Remark

All rewriting techniques from expressive DLs to *Datalog*$^{(\vee)}$ produce rule sets of exponential size or unbounded arity.

Theorem 1

$\mathcal{ALCHIQ}$ is poly-time rewritable into terminating Datalog$^{\vee\exists}$ rules **of bounded size**.

Theorem 1

$\mathcal{ALCHIQ}$ is poly-time rewritable into terminating Datalog$^{\vee\exists}$ rules **of bounded size**.

Definition: Terminating Datalog$^{\vee\exists}$

Language of all sets $\mathcal{R}$ of disjunctive existential rules that terminate with respect to the Datalog-first restricted chase.

Theorem 1

$\mathcal{ALCHIQ}$ is poly-time rewritable into terminating Datalog$^{\vee\exists}$ rules **of bounded size**.

Definition: Terminating Datalog$^{\vee\exists}$

Language of all sets $\mathcal{R}$ of disjunctive existential rules that terminate with respect to the Datalog-first restricted chase.

Theorem 2

$\mathcal{ALCHIQ}$ is poly-time rewritable to Datalog$^{\vee}$ rules (of unbounded size).

Simplified Theorem

$\mathcal{ALC}$ is poly-time rewritable into terminating Datalog$^{\vee\exists}$ rules of bounded size.

Simplified Theorem

$\mathcal{ALC}$ is poly-time rewritable into terminating Datalog$^{\vee\exists}$ rules of bounded size.

Definition: $\mathcal{ALC}$

$$
\begin{aligned}
A \sqcap B \sqsubseteq C & \qquad & A(x) \wedge B(x) \rightarrow C(x) \\
A \sqsubseteq B \sqcup C & \qquad & A(x) \rightarrow B(x) \vee C(x) \\
A \sqsubseteq \forall R.B & \qquad & A(x) \wedge R(x,y) \rightarrow B(y) \\
A \sqsubseteq \exists R.B & \qquad & A(x) \rightarrow \exists y.R(x,y) \wedge B(y)
\end{aligned}
$$

In the above, A, B, and C are unary predicates (i.e., concept names) and R is a binary predicate (i.e., role name)

Definition: $\mathcal{ALC}$ Rewritings

Consider a theory $\mathcal{T}$ of $\mathcal{ALC}$ axioms and a sequence $A_1, \ldots, A_n$ containing all of the classes in $\mathcal{T}$. Then, the following set of Datalog$^{\vee\exists}$ rules is a terminating rewriting for $\mathcal{T}$:

$$\{A(x) \wedge B(x) \rightarrow C(x) \mid A \sqcap B \sqsubseteq C \in \mathcal{T}\} \cup \{A(x) \rightarrow B(x) \vee C(x) \mid A \sqsubseteq B \sqcup C \in \mathcal{T}\} \cup$$
$$\{A(x) \wedge R(x,y) \rightarrow B(y) \mid A \sqsubseteq \forall R.B \in \mathcal{T}\} \cup$$
$$\{A(x) \rightarrow \exists y.R(x,y) \wedge B(y) \wedge \mathit{Succ}(x,y) \mid A \sqsubseteq \exists R.B \in \mathcal{T}\} \cup$$

$$\{ \rightarrow A(x) \vee A^{\neg}(x), A(x) \wedge A^{\neg}(x) \rightarrow \bot \mid A \in \textbf{Classes}(\mathcal{T})\} \cup$$
$$\{A_1(x) \wedge A_1(z) \rightarrow \mathit{SameClasses}_1(x,z), A_1^{\neg}(x) \wedge A_1^{\neg}(z) \rightarrow \mathit{SameClasses}_1(x,z)\} \cup$$
$$\{\mathit{SameClasses}_{i-1}(x,z) \wedge A_i(x) \wedge A_i(z) \rightarrow \mathit{SameClasses}_i(x,z),$$
$$\mathit{SameClasses}_{i-1}(x,z) \wedge A_i^{\neg}(x) \wedge A_i^{\neg}(z) \rightarrow \mathit{SameClasses}_i(x,z) \mid 2 \leq i \leq n\} \cup$$
$$\{\mathit{SameClasses}_n(x,y) \rightarrow \mathit{SameType}(x,y)\} \cup$$

$$\{\mathit{SameType}(x,z) \wedge \mathit{Succ}(x,y) \wedge R(x,y) \rightarrow \mathit{Succ}(z,y) \wedge R(z,y) \mid R \in \textbf{Roles}(\mathcal{T})\}$$

Thank you for your attention!

Link to the paper:
iccl.inf.tu-dresden.de/web/Inproceedings3244/en

References I

Ahmetaj, S., Ortiz, M., and Simkus, M. (2016).
Polynomial Datalog rewritings for expressive Description Logics with closed predicates.
In Kambhampati, S., editor, *Proc. 25th Int. Joint Conf. on Artif. Intell. (IJCAI 2016)*, pages 878–885. IJCAI/AAAI Press.

Bienvenu, M., ten Cate, B., Lutz, C., and Wolter, F. (2014).
Ontology-based data access: A study through disjunctive Datalog, CSP, and MMSNP.
ACM Transactions of Database Systems, 39(4):33:1–33:44.

Carral, D., Dragoste, I., and Krötzsch, M. (2018).
The combined approach to query answering in Horn-$\mathcal{ALCHOIQ}$.
In Thielscher, M., Toni, F., and Wolter, F., editors, *Proc. 16th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR 2018)*, pages 339–348. AAAI Press.

Carral, D., Dragoste, I., Krötzsch, M., and Lewe, C. (2019a).
Chasing sets: How to use existential rules for expressive reasoning.
In Kraus, S., editor, *Proc. 28th Int. Joint Conf. on Artif. Intell. (IJCAI 2019)*, pages 1624–1631. ijcai.org.

Carral, D., González, L., and Koopmann, P. (2019b).
From Horn-$\mathcal{SRIQ}$ to Datalog: A data-independent transformation that preserves assertion entailment.
In *Proc. 33rd AAAI Conf. on Artificial Intelligence (AAAI 2019)*, pages 2736–2743. AAAI Press.

Eiter, T., Ortiz, M., Simkus, M., Tran, T.-K., and Xiao, G. (2012).
Query rewriting for Horn-$\mathcal{SHIQ}$ plus rules.
In Hoffmann, J. and Selman, B., editors, *Proc. 26th AAAI Conf. on Artificial Intelligence (AAAI 2012)*. AAAI Press.

Hustadt, U., Motik, B., and Sattler, U. (2007).
Reasoning in Description Logics by a reduction to disjunctive Datalog.
J. Automated Reasoning, 39(3):351–384.

Krötzsch, M. (2011).
Efficient rule-based inferencing for OWL EL.
In Walsh, T., editor, *Proc. 22nd Int. Joint Conf. on Artif. Intell. (IJCAI 2011)*, pages 2668–2673. IJCAI/AAAI.

Ortiz, M., Rudolph, S., and Simkus, M. (2010).
Worst-case optimal reasoning for the Horn-DL fragments of OWL 1 and 2.
In Lin, F., Sattler, U., and Truszczynski, M., editors, *Proc. 12th Int. Conf. on Principles of Knowledge Representation and Reasoning (KR 2010)*. AAAI Press.

References IV

Rudolph, S., Krötzsch, M., and Hitzler, P. (2012).
Type-elimination-based reasoning for the Description Logic $\mathcal{SHIQ}b_s$ using decision diagrams and disjunctive Datalog.
Logical Methods in Computer Science, 8(1).