Department of Decision Sciences

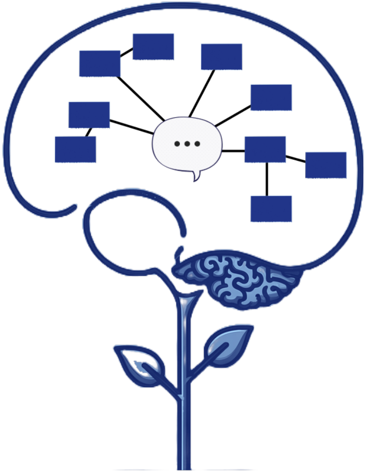
Large-scale Optimization Methods for Logical Reasoning: A Novel Perspective

Maryam Daryalal

MIP Workshop
June 2024

Outline

- Introduction
- Problem Description
- Solution Methodology
- Results & Conclusion



Introduction

Description Logic

- Description Logics ($\mathcal{DL}$) are a family of formal knowledge representation languages.
- Used to represent the knowledge of an application domain in a structured and formal way.
- Provides a mechanism for encoding semantics of a domain and reasoning about it.

Description Logic

- Description Logics ($\mathcal{DL}$) are a family of formal knowledge representation languages.
- Used to represent the knowledge of an application domain in a structured and formal way.
- Provides a mechanism for encoding semantics of a domain and reasoning about it.

Checking satisfiability; classification; answering queries

Description Logic

- Description Logics ($\mathcal{DL}$) are a family of formal knowledge representation languages.
- Used to represent the knowledge of an application domain in a structured and formal way.
- Provides a mechanism for encoding semantics of a domain and reasoning about it.

  Checking satisfiability; classification; answering queries

- It serves as a foundation for implementing ontologies and semantic web technologies

Description Logic

- Description Logics ($\mathcal{DL}$) are a family of formal knowledge representation languages.
- Used to represent the knowledge of an application domain in a structured and formal way.
- Provides a mechanism for encoding semantics of a domain and reasoning about it.

  Checking satisfiability; classification; answering queries

- It serves as a foundation for implementing ontologies and semantic web technologies

  Used to define a formal representation of a domain using concepts and relationships.

Description Logic

- Description Logics ($\mathcal{DL}$) are a family of formal knowledge representation languages.
- Used to represent the knowledge of an application domain in a structured and formal way.
- Provides a mechanism for encoding semantics of a domain and reasoning about it.

  Checking satisfiability; classification; answering queries

- It serves as a foundation for implementing ontologies and semantic web technologies

  Used to define a formal representation of a domain using concepts and relationships.

  An extension of the traditional Web, enables computers to understand web data, using ontologies.

The Semantic Web & Description Logic

- OWL (Web Ontology Language): A language for defining and instantiating Web ontologies.
  - OWL uses $\mathcal{DL}$ to provide semantics for complex ontologies.
  - OWL $\mathcal{DL}$ is compatible with existing Web standards, e.g., HTTP, XML, RDF, RDFS

The Semantic Web & Description Logic

- OWL (Web Ontology Language): A language for defining and instantiating Web ontologies.
  - OWL uses $\mathcal{DL}$ to provide semantics for complex ontologies.
  - OWL $\mathcal{DL}$ is compatible with existing Web standards, e.g., HTTP, XML, RDF, RDFS

$\rightarrow$ $\mathcal{DL}$ provides a theoretical basis for semantic reasoning on the Web.

The Semantic Web & Description Logic

- OWL (Web Ontology Language): A language for defining and instantiating Web ontologies.
  - OWL uses $\mathcal{DL}$ to provide semantics for complex ontologies.
  - OWL $\mathcal{DL}$ is compatible with existing Web standards, e.g., HTTP, XML, RDF, RDFS

$\rightarrow$ $\mathcal{DL}$ provides a theoretical basis for semantic reasoning on the Web.

Semantic Web: Data is not just structured but also meaningful and machine-understandable.

The Semantic Web & Description Logic

- OWL (Web Ontology Language): A language for defining and instantiating Web ontologies.
  - OWL uses $\mathcal{DL}$ to provide semantics for complex ontologies.
  - OWL $\mathcal{DL}$ is compatible with existing Web standards, e.g., HTTP, XML, RDF, RDFS

$\rightarrow$ $\mathcal{DL}$ provides a theoretical basis for semantic reasoning on the Web.

Semantic Web: Data is not just structured but also meaningful and machine-understandable.

- Why not the existing Web data models? XML? RDF?
  - XML: syntax ✓, semantics ×
  - RDF: syntax ✓, (basic) semantics ✓, reasoning ×
  - OWL $\mathcal{DL}$: syntax ✓, (rich) semantics ✓, reasoning ✓

A Simple Ontology: Description

HEC MONTRÉAL

Modeling a University domain including entities like Professors, Students, and Courses.

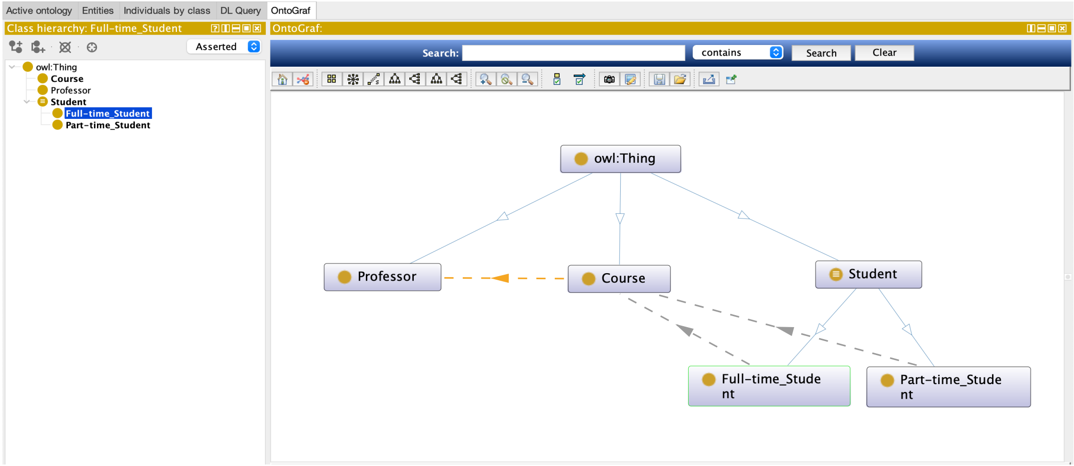
- **Concepts:** Professor, Student, Full-time Student, Part-time Student, Course
- **Roles:** `teaches`(Professor, Course), `enrolled`(Student, Course)
- **Axioms:**
  - Every Full-time Student is a Student. Every Part-time Student is a Student.
  - A Student is either Full-time Student, or Part-time Student. They cannot be both.
  - Every Full-time Student is enrolled in at least 3 Courses.
  - Every Part-time Student is enrolled in at most 2 Courses.
  - For every Course there exists some Professor teaching it.

A Simple Ontology: $\mathcal{DL}$ Syntax

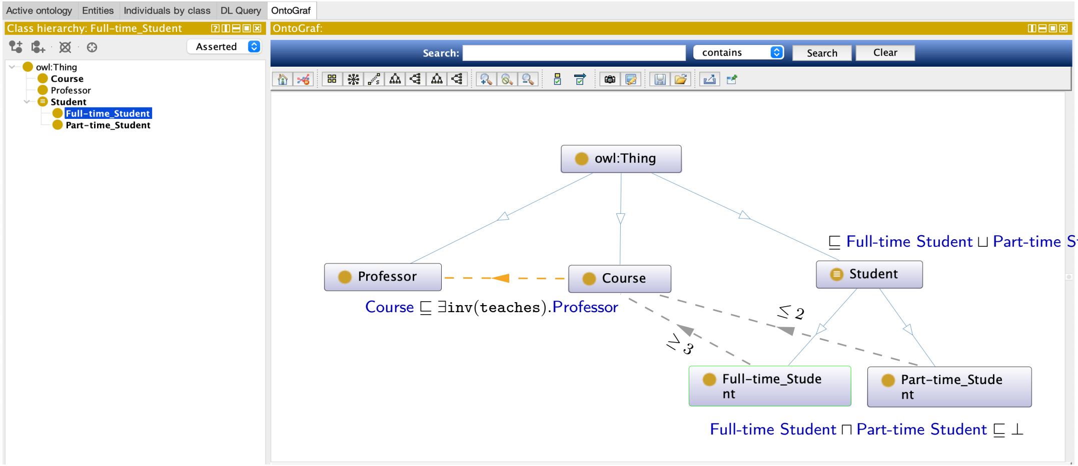
Modeling a University domain including entities like Professors, Students, and Courses.

- **Concepts:** Professor, Student, Full-time Student, Part-time Student, Course
- **Roles:** `teaches`(Professor, Course), `enrolled`(Student, Course)
- **Axioms:**
  - Full-time Student $\sqsubseteq$ Student, Part-time Student $\sqsubseteq$ Student
  - Student $\sqsubseteq$ Full-time Student $\sqcup$ Part-time Student, Full-time Student $\sqcap$ Part-time Student $\sqsubseteq \bot$
  - Full-time Student $\sqsubseteq \geq 3\, \texttt{enrolled}.$Course
  - Part-time Student $\sqsubseteq \leq 2\, \texttt{enrolled}.$Course
  - Course $\sqsubseteq \exists \texttt{inv}(\texttt{teaches}).$Professor

A Simple Ontology: Knowledge Graph



A Simple Ontology: Knowledge Graph



$\sqsubseteq$ Full-time Student $\sqcup$ Part-time Student

Course $\sqsubseteq \exists \texttt{inv(teaches)}.$Professor

Full-time Student $\sqcap$ Part-time Student $\sqsubseteq \bot$

A Simple Ontology: Reasoning

- OWL $\mathcal{DL}$ can also infer new knowledge $\rightarrow$ reasoning

A Simple Ontology: Reasoning

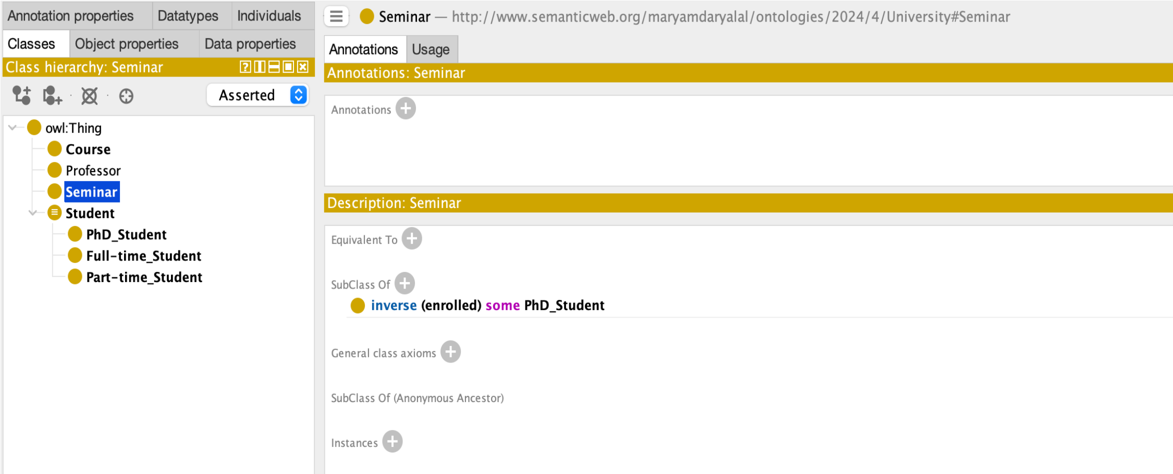
- OWL $\mathcal{DL}$ can also infer new knowledge $\rightarrow$ reasoning
- Let's add two new concepts to our ontology:
  - PhD-Student $\sqsubseteq$ Student
  - Seminar $\sqsubseteq \exists \texttt{inv}(\texttt{enrolled})$.PhD-Student

A Simple Ontology: Reasoning

- OWL $\mathcal{DL}$ can also infer new knowledge $\rightarrow$ reasoning
- Let's add two new concepts to our ontology:
  - PhD-Student $\sqsubseteq$ Student
  - Seminar $\sqsubseteq \exists \texttt{inv}(\texttt{enrolled})$.PhD-Student
- We haven't explicitly told the reasoner that a Seminar is a Course. It will infer this.
  - RDF cannot represent the semantics of our ontology. It lacks the vocabulary for disjointedness, cardinality, etc.
  - RDF cannot infer new knowledge.

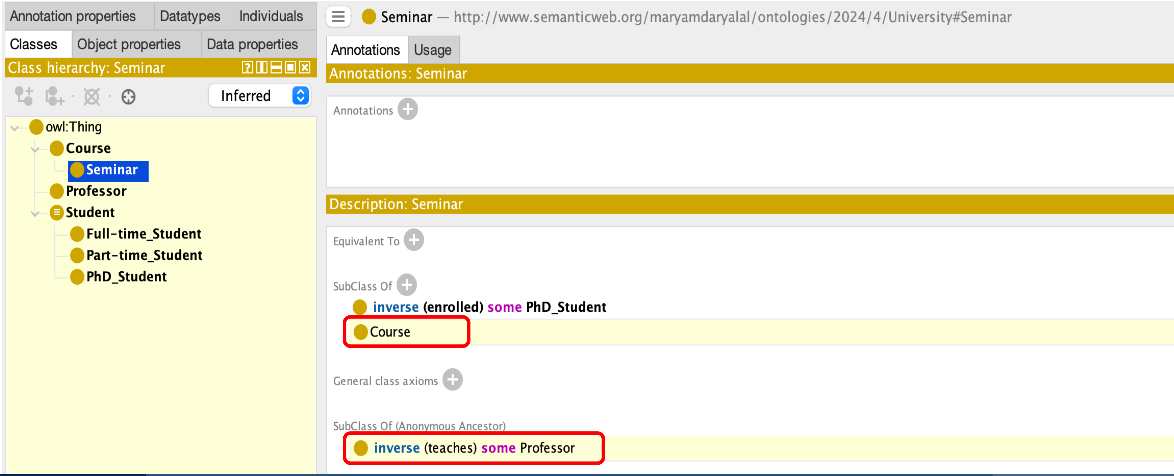
A Simple Ontology: Reasoning

- Added concepts and axiom



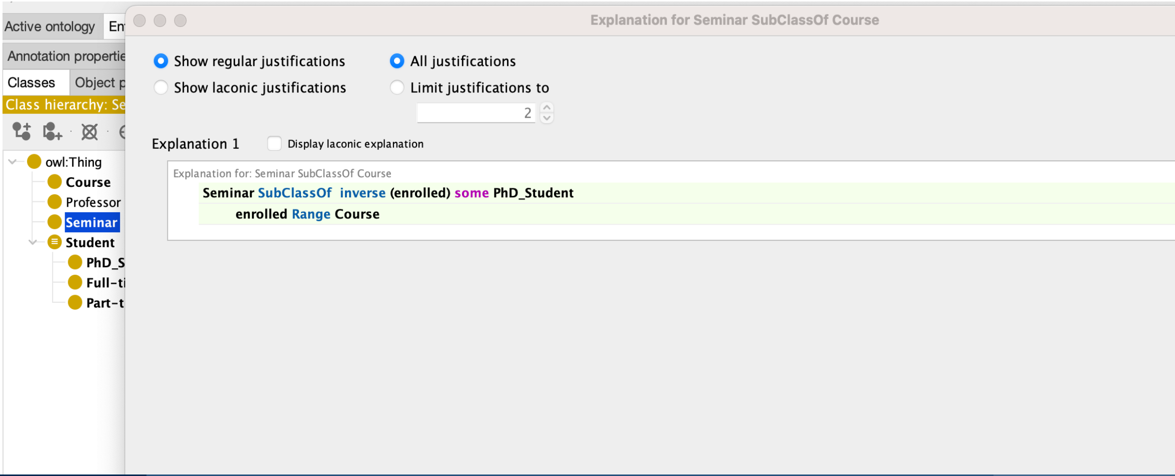
A Simple Ontology: Reasoning

- Inferences made by the reasoner

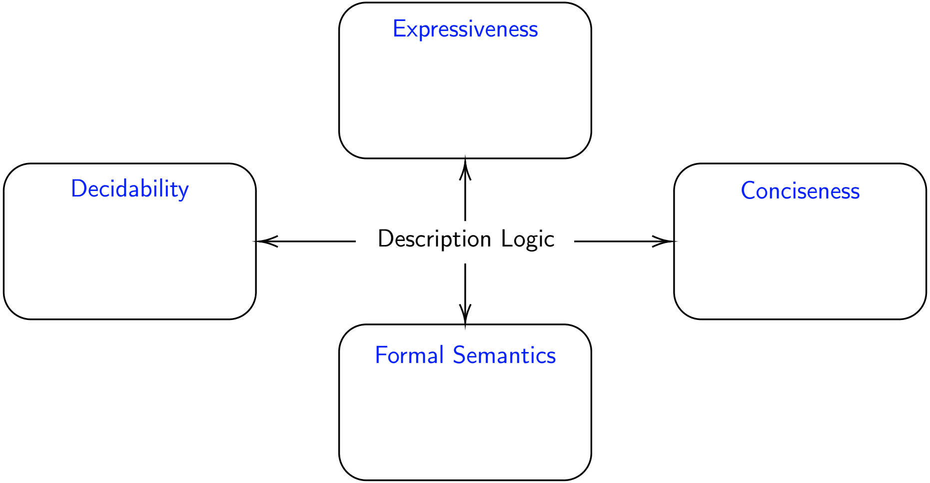


A Simple Ontology: Reasoning

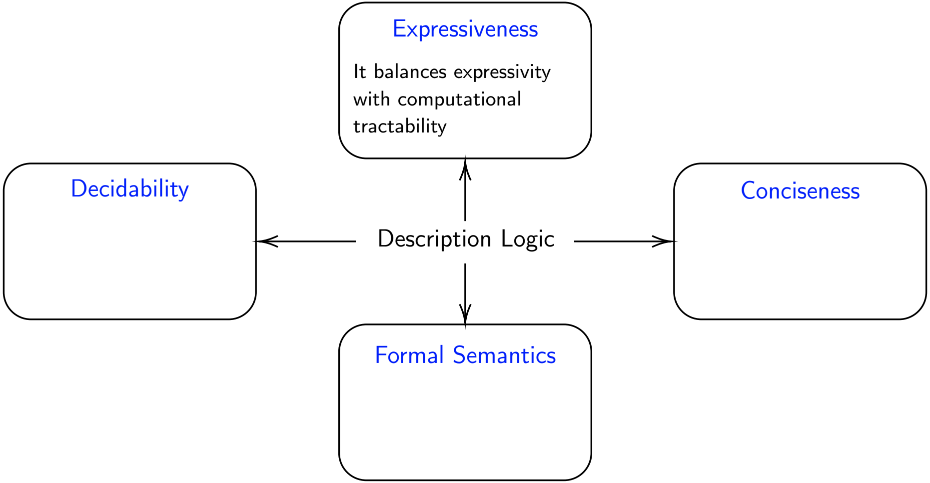
- Explanations provided by the reasoner



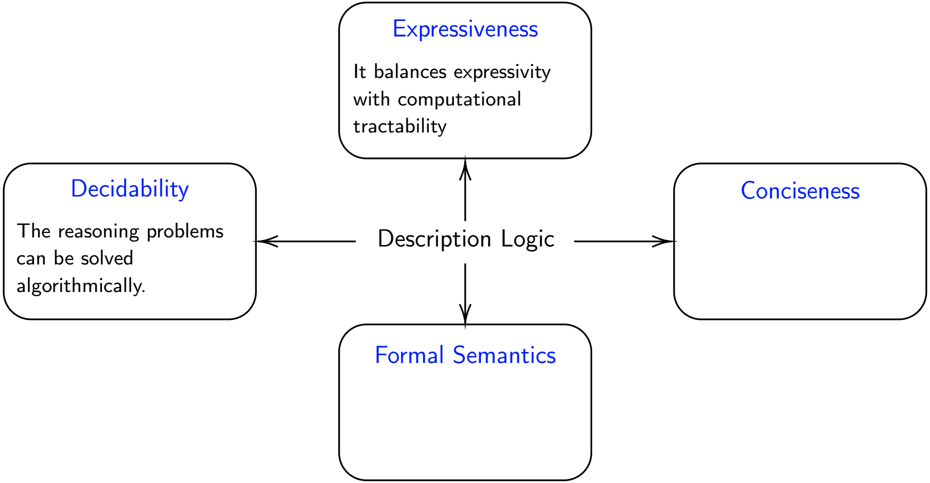
Takeaway: Key Features of $\mathcal{DL}$



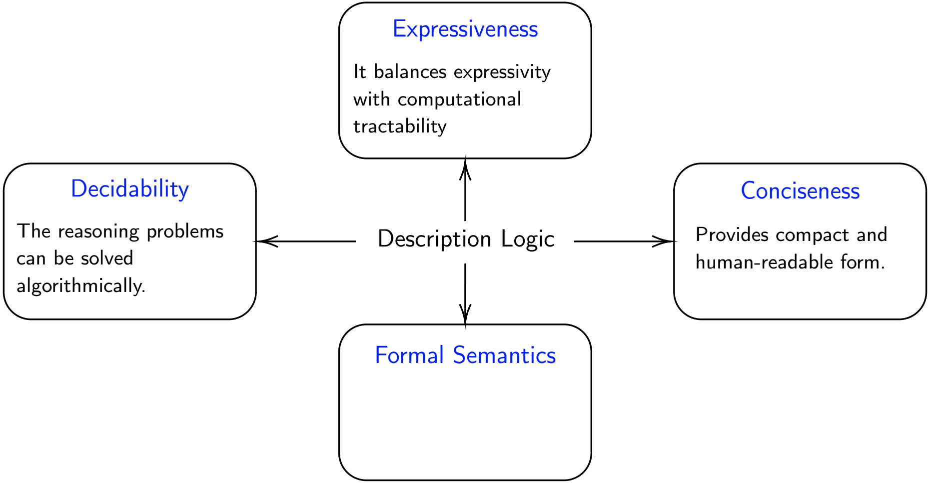
Takeaway: Key Features of $\mathcal{DL}$



Takeaway: Key Features of $\mathcal{DL}$



Takeaway: Key Features of $\mathcal{DL}$



Takeaway: Key Features of $\mathcal{DL}$

Description Logic

Expressiveness

It balances expressivity with computational tractability

Decidability

The reasoning problems can be solved algorithmically.

Conciseness

Provides compact and human-readable form.

Formal Semantics

Uses formal logic-based semantics to avoid ambiguity.

Constructors & Axioms in $\mathcal{DL}$ ALCQ (I)

- An interpretation: $\mathcal{I} = (\Delta^{\mathcal{I}}, .^{\mathcal{I}})$, with $\Delta^{\mathcal{I}}$ a non-empty domain set and $.^{\mathcal{I}}$ a mapping.

Thing	$\top$	$\equiv$	$\top^{\mathcal{I}} = \Delta^{\mathcal{I}}$
Nothing	$\bot$	$\equiv$	$\bot^{\mathcal{I}} = \emptyset$
Concept (class)	A	$\equiv$	$A^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}}$
Concept assertion	$a : C$	$\equiv$	$a^{\mathcal{I}} \in C^{\mathcal{I}}$
Negation	$\neg C$	$\equiv$	$\Delta^{\mathcal{I}} \setminus C^{\mathcal{I}}$
Conjunction	$C \sqcap D$	$\equiv$	$C^{\mathcal{I}} \cap D^{\mathcal{I}}$
Disjunction	$C \sqcup D$	$\equiv$	$C^{\mathcal{I}} \cup D^{\mathcal{I}}$
Subsumption	$C \sqsubseteq D$	$\equiv$	$C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$

Constructors & Axioms in $\mathcal{DL}$ ALCQ (II)

- An interpretation: $\mathcal{I} = (\Delta^{\mathcal{I}}, .^{\mathcal{I}})$, with $\Delta^{\mathcal{I}}$ a non-empty domain set and $.^{\mathcal{I}}$ a mapping.

Role (relationship)	R	$\equiv$	$R^{\mathcal{I}} \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$
Role assertion	$(a, b) : R$	$\equiv$	$(a^{\mathcal{I}}, b^{\mathcal{I}}) \in R^{\mathcal{I}}$
Universal restriction	$\forall R.C$	$\equiv$	$\{x \mid \forall y : (x, y) \in R^{\mathcal{I}} \Rightarrow y \in C^{\mathcal{I}}\}$
At-least qualified cardinality restriction	$\geq nR.C$	$\equiv$	$\{x \mid \# R^{\mathcal{I}}(x, C) \geq n\}$
At-most qualified cardinality restriction	$\leq mR.C$	$\equiv$	$\{x \mid \# R^{\mathcal{I}}(x, C) \leq m\}$
Role hierarchy	$R \sqsubseteq S$	$\equiv$	$R^{\mathcal{I}} \subseteq S^{\mathcal{I}}$
Transitive role	$R \in N_{RT}$	$\equiv$	$R^{\mathcal{I}} = (R^{\mathcal{I}})^{+}$

Problem Description

The Satisfiability Problem in $\mathcal{DL}$ Ontologies

The SAT Problem

Given an ontology $\mathcal{O}$ written in a Description Logic $\mathcal{L}$, and a concept C, is there a model $\mathcal{I}$ of $\mathcal{O}$ where $C^{\mathcal{I}} \neq \emptyset$?

- Does there exist an interpretation that satisfies all axioms in $\mathcal{O}$ and where C is non-empty?
- Ontology axioms constrain possible $\mathcal{I}$s, potentially making a concept unsatisfiable.

The Satisfiability Problem in $\mathcal{DL}$ Ontologies

HEC MONTRÉAL

The SAT Problem

Given an ontology $\mathcal{O}$ written in a Description Logic $\mathcal{L}$, and a concept C, is there a model $\mathcal{I}$ of $\mathcal{O}$ where $C^{\mathcal{I}} \neq \emptyset$?

- Does there exist an interpretation that satisfies all axioms in $\mathcal{O}$ and where C is non-empty?
- Ontology axioms constrain possible $\mathcal{I}$s, potentially making a concept unsatisfiable.
- **Example**:
  - Full-time Student $\sqsubseteq$ Student
  - Professor $\sqsubseteq \neg$ Student (Professors are not Students)
  - Is Professor $\sqcap$ Full-time Student satisfiable? No $\rightarrow$ Professor $\sqcap$ Full-time Student $\sqsubseteq \bot$

The Challenge of Qualified Cardinality Restrictions (QCRs)

HEC MONTRÉAL

- QCRs are expressive, but computationally challenging for reasoning algorithms.
- Reasoning with QCRs:
  - Tableaux algorithms: Introduce or merge individuals to satisfy cardinality constraints.
    - Example: For Person $\sqsubseteq \geq 2\texttt{hasChild}.$Student:
      - Start with individual ‘a: Person’, ‘a: $(\geq 2\texttt{hasChild}.$Student$)$’
      - Introduce ‘$\texttt{b1}, \texttt{b2}$: Student’ such that $\texttt{hasChild}(a, \texttt{b1})$, $\texttt{hasChild}(a, \texttt{b2})$
    - Challenges: Non-determinism, exponential complexity.

The Challenge of Qualified Cardinality Restrictions (QCRs)

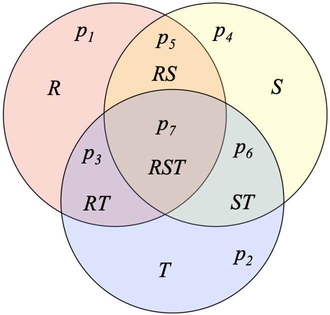
- QCRs are expressive, but computationally challenging for reasoning algorithms.
- Reasoning with QCRs:
  - Tableaux algorithms: Introduce or merge individuals to satisfy cardinality constraints.
    - Example: For Person $\sqsubseteq \geq 2 \texttt{hasChild}.$Student:
      - Start with individual 'a: Person', 'a: ($\geq 2\texttt{hasChild}.$Student)'
      - Introduce '$\texttt{b1}, \texttt{b2}$: Student' such that $\texttt{hasChild}(\mathrm{a}, \texttt{b1})$, $\texttt{hasChild}(\mathrm{a}, \texttt{b2})$
    - Challenges: Non-determinism, exponential complexity.
  - Reduction to other problems $\rightarrow$ Feasibility problem: Given a set of constraints $\mathcal{T}$, does there exist a solution x that satisfies all constraints in $\mathcal{T}$?

The Challenge of Qualified Cardinality Restrictions (QCRs)

- QCRs are expressive, but computationally challenging for reasoning algorithms.
- Reasoning with QCRs:
  - Tableaux algorithms: Introduce or merge individuals to satisfy cardinality constraints.
    - Example: For Person $\sqsubseteq \geq 2\texttt{hasChild}.$Student:
      - Start with individual 'a: Person', 'a: $(\geq 2\texttt{hasChild}.$Student)'
      - Introduce '$\texttt{b1}, \texttt{b2}$: Student' such that $\texttt{hasChild}(\texttt{a}, \texttt{b1})$, $\texttt{hasChild}(\texttt{a}, \texttt{b2})$
    - Challenges: Non-determinism, exponential complexity.
  - Reduction to other problems $\rightarrow$ Feasibility problem: Given a set of constraints $\mathcal{T}$, does there exist a solution x that satisfies all constraints in $\mathcal{T}$?

QCRs as Linear Inequalities: The Idea

- Example: $S = \{\geq 3R, \leq 2T, \geq 1S, \leq 1S\}$
- **Atomic Decomposition** of S:



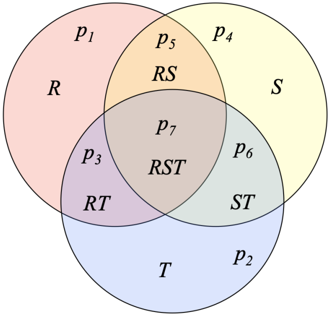
$$p_1 = \{R\}, p_2 = \{T\}, p_4 = \{S\},$$
$$p_3 = \{R, T\}, p_5 = \{R, S\}, p_6 = \{S, T\},$$
$$p_7 = \{R, S, T\}$$

QCRs as Linear Inequalities: The Idea

- Example: $S = \{\geq 3R, \leq 2T, \geq 1S, \leq 1S\}$
- **Atomic Decomposition** of S:

1. Define int variable v_{i_S, i_T, i_R} for each partition:

$$p_1 \to v_{001},\ p_2 \to v_{010},\ p_3 \to v_{011},\ p_4 \to v_{100},$$
$$p_5 \to v_{101},\ p_6 \to v_{110},\ p_7 \to v_{111}$$



$$p_1 = \{R\}, p_2 = \{T\}, p_4 = \{S\},$$
$$p_3 = \{R, T\}, p_5 = \{R, S\}, p_6 = \{S, T\},$$
$$p_7 = \{R, S, T\}$$

QCRs as Linear Inequalities: The Idea

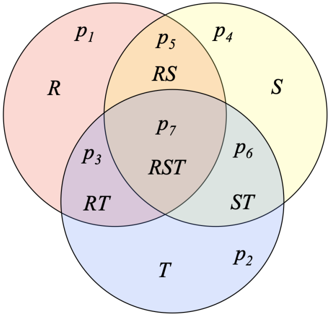
- Example: $S = \{\geq 3R, \leq 2T, \geq 1S, \leq 1S\}$
- **Atomic Decomposition** of S:

1. Define int variable v_{i_S, i_T, i_R} for each partition:

$$p_1 \rightarrow v_{001},\ p_2 \rightarrow v_{010},\ p_3 \rightarrow v_{011},\ p_4 \rightarrow v_{100},$$
$$p_5 \rightarrow v_{101},\ p_6 \rightarrow v_{110},\ p_7 \rightarrow v_{111}$$

2. Write S as:

$$v_{001} + v_{011} + v_{101} + v_{111} \geq 3$$
$$v_{010} + v_{011} + v_{110} + v_{111} \leq 2$$
$$v_{100} + v_{101} + v_{110} + v_{111} \leq 1$$
$$v_{100} + v_{101} + v_{110} + v_{111} \geq 1$$



$p_1 = \{R\}, p_2 = \{T\}, p_4 = \{S\},$
$p_3 = \{R, T\}, p_5 = \{R, S\}, p_6 = \{S, T\},$
$p_7 = \{R, S, T\}$

QCRs as Linear Inequalities: Compact Model

- Back to our own world!
- Let $\mathcal{R}$ be the set of all roles, and $\bar{\delta}_R$ and $\underline{\delta}_R$ be the right-hand side of at-most and at-least restrictions on a role R.

$$\min \quad \sum_{R \in \mathcal{R}} \sum_{i_R \in \{0,1\}} v_{i_1,\dots,i_{|\mathcal{R}|}}$$

$$\text{s.t.} \quad \underline{\delta}_R \leq \sum_{j \in \mathcal{R}} \sum_{\substack{j \neq R : i_j \in \{0,1\} \\ j = R : i_j = 1}} v_{i_1,\dots,i_{|\mathcal{R}|}} \leq \bar{\delta}_R \qquad R \in \mathcal{R}$$

$$v_{i_1,\dots,i_{|\mathcal{R}|}} \in \mathbb{Z}^+ \qquad R \in \mathcal{R}, i_R \in \{0,1\}$$

QCRs as Linear Inequalities: Compact Model

- Back to our own world!
- Let $\mathcal{R}$ be the set of all roles, and $\bar{\delta}_R$ and $\underline{\delta}_R$ be the right-hand side of at-most and at-least restrictions on a role R.

$$\begin{aligned}
\min \quad & \sum_{R \in \mathcal{R}} \sum_{i_R \in \{0,1\}} v_{i_1,\ldots,i_{|\mathcal{R}|}} \\
\text{s.t.} \quad & \underline{\delta}_R \leq \sum_{j \in \mathcal{R}} \sum_{\substack{j \neq R: i_j \in \{0,1\} \\ j = R: i_j = 1}} v_{i_1,\ldots,i_{|\mathcal{R}|}} \leq \bar{\delta}_R \quad && R \in \mathcal{R} \\
& v_{i_1,\ldots,i_{|\mathcal{R}|}} \in \mathbb{Z}^+ && R \in \mathcal{R}, i_R \in \{0,1\}
\end{aligned}$$

- How many partitions do we have?

QCRs as Linear Inequalities: Compact Model

- Back to our own world!
- Let $\mathcal{R}$ be the set of all roles, and $\bar{\delta}_R$ and $\underline{\delta}_R$ be the right-hand side of at-most and at-least restrictions on a role R.

$$\begin{aligned}
\min \quad & \sum_{R\in\mathcal{R}} \sum_{i_R\in\{0,1\}} v_{i_1,\dots,i_{|\mathcal{R}|}} \\
\text{s.t.} \quad & \underline{\delta}_R \le \sum_{j\in\mathcal{R}} \sum_{\substack{j\neq R: i_j\in\{0,1\} \\ j=R: i_j=1}} v_{i_1,\dots,i_{|\mathcal{R}|}} \le \bar{\delta}_R && R\in\mathcal{R} \\
& v_{i_1,\dots,i_{|\mathcal{R}|}} \in \mathbb{Z}^+ && R\in\mathcal{R}, i_R\in\{0,1\}
\end{aligned}$$

- How many partitions do we have?
- There's also other axioms and concepts we haven't considered yet...

Solution Methodology

Extended Formulation for QCRs (I)

- Define a mapping $\alpha(.)$ that assigns a newly defined sub-role $R' \sqsubseteq R$ to each QCR:

$$\alpha(\bowtie nR.C) = R'.$$

Extended Formulation for QCRs (I)

- Define a mapping $\alpha(.)$ that assigns a newly defined sub-role $R' \sqsubseteq R$ to each QCR:

$$\alpha(\bowtie nR.C) = R'.$$

- Define $S_Q = \{\alpha(\bowtie nR.C) \mid \bowtie nR.C \in S\} \cup \{C \mid \bowtie nR.C \in S\} \cup \{\top, \bot\}$.

Extended Formulation for QCRs (I)

- Define a mapping $\alpha(.)$ that assigns a newly defined sub-role $R' \sqsubseteq R$ to each QCR:
$$\alpha(\bowtie nR.C) = R'.$$
- Define $S_Q = \{\alpha(\bowtie nR.C) \mid \bowtie nR.C \in S\} \cup \{C \mid \bowtie nR.C \in S\} \cup \{\top, \bot\}$.
- Define $\mathcal{P}_{S_Q}$ as the power set of S_Q excluding the empty set, and any subset without a role.

Extended Formulation for QCRs (I)

HEC MONTRÉAL

- Define a mapping $\alpha(.)$ that assigns a newly defined sub-role $R' \sqsubseteq R$ to each QCR:

$$\alpha(\bowtie nR.C) = R'.$$

- Define $S_Q = \{\alpha(\bowtie nR.C) \mid \bowtie nR.C \in S\} \cup \{C \mid \bowtie nR.C \in S\} \cup \{\top, \bot\}$.
- Define $\mathcal{P}_{S_Q}$ as the power set of S_Q excluding the empty set, and any subset without a role.
- **A partition configuration**: Represents a partition p in $\mathcal{P}_{S_Q}$. It is a set of binary parameters $a_p^{R'}, R' \in S_Q$:

$$a_p^{R'} = \begin{cases} 1 & \text{if role } R' \in p \\ 0 & \text{otherwise.} \end{cases}$$

Extended Formulation for QCRs (I)

- Define a mapping $\alpha(.)$ that assigns a newly defined sub-role $R' \sqsubseteq R$ to each QCR:

$$\alpha(\bowtie nR.C) = R'.$$

- Define $S_Q = \{\alpha(\bowtie nR.C) \mid \bowtie nR.C \in S\} \cup \{C \mid \bowtie nR.C \in S\} \cup \{\top, \bot\}$.
- Define $\mathcal{P}_{S_Q}$ as the power set of S_Q excluding the empty set, and any subset without a role.
- **A partition configuration**: Represents a partition p in $\mathcal{P}_{S_Q}$. It is a set of binary parameters $a_p^{R'}, R' \in S_Q$:

$$a_p^{R'} = \begin{cases} 1 & \text{if role } R' \in p \\ 0 & \text{otherwise.} \end{cases}$$

- cost_p: Cost of partition p, defined as the number of concepts in p
  - $\rightarrow$ We want only explicitly entailed concepts

Extended Formulation for QCRs (II)

HEC MONTRÉAL

- $x_p \in \mathbb{Z}^+$: The number of individuals belonging to partition p in the optimal solution.

$$
\begin{aligned}
\rightarrow \quad \mathsf{EF}(\mathcal{P}_{S_Q}) = \min \quad & \sum_{p \in \mathcal{P}_{S_Q}} \mathsf{cost}_p x_p \\
\text{s.t.} \quad & \sum_{p \in \mathcal{P}_{S_Q}} a_p^{R'} x_p \geq \underline{\delta}_{R'}, \quad R' \in \{\alpha(\geq nR.C) \mid \geq nR.C \in S\} \\
& \sum_{p \in \mathcal{P}_{S_Q}} a_p^{R'} x_p \leq \bar{\delta}_{R'}, \quad R' \in \{\alpha(\leq nR.C) \mid \leq nR.C \in S\} \\
& x_p \in \mathbb{Z}^+, \quad p \in \mathcal{P}_{S_Q}.
\end{aligned}
$$

- A branch-and-price framework can implicitly enumerate the exponentially many partitions.
- We will take care of all other axioms inside the implicit enumeration.

Branch-and-Price for QCRs

- Let $\mathcal{P}' \subseteq \mathcal{P}_{S_Q}$. Then $\mathsf{EF}^{\mathsf{LP}}(\mathcal{P}')$ is the LP relaxation of EF over $\mathcal{P}'$.
- **Partition Generation**:
  - Let $\boldsymbol{\pi}$ and $\boldsymbol{\omega}$ be the dual vectors associated with $\geq$ and $\leq$ constraints in $\mathsf{EF}^{\mathsf{LP}}(\mathcal{P}')$, respectively.
  - Let $a^{R'} \in \{0, 1\}$ be a decision variable equal to 1 iff role R' is in the generated partition.
  - Let $b_C \in \{0, 1\}$ be a decision variable equal to 1 iff concept C is in the generated partition.

Branch-and-Price for QCRs

HEC MONTRÉAL

- Let $\mathcal{P}' \subseteq \mathcal{P}_{S_Q}$. Then $\text{EF}^{\text{LP}}(\mathcal{P}')$ is the LP relaxation of EF over $\mathcal{P}'$.
- **Partition Generation**:
  - Let $\boldsymbol{\pi}$ and $\boldsymbol{\omega}$ be the dual vectors associated with $\geq$ and $\leq$ constraints in $\text{EF}^{\text{LP}}(\mathcal{P}')$, respectively.
  - Let $a^{R'} \in \{0,1\}$ be a decision variable equal to 1 iff role R' is in the generated partition.
  - Let $b_C \in \{0,1\}$ be a decision variable equal to 1 iff concept C is in the generated partition.

$$\begin{aligned}
\to \ \text{PP} = \min \quad & \text{Reduced-cost}(\hat{\boldsymbol{\pi}}, \hat{\boldsymbol{\omega}}) \\
\text{s.t.} \quad & a^{R'} \bowtie b_C, && R' \in \{\alpha(\bowtie nR.C) \mid \bowtie nR.C \in S\}, C = R'.\text{Qualifier} \\
& b_C \leq b_\top, && C \in \{C \mid \bowtie nR.C \in S\} \\
& b_\perp = 0 \\
& \text{All other axioms} \\
& b_C, a^{R'} \in \{0,1\}, && R' \in \{\alpha(\bowtie nR.C) \mid \bowtie nR.C \in S\}, C \in \{C \mid \bowtie nR.C \in S\}
\end{aligned}$$

Mapping of Axioms (I)

- Basic axioms:
  - For every subsumption $A \sqsubseteq B$, add the following to PP:

$$b_A \leq b_B.$$

  - For every binary subsumption $A \sqcap B \sqsubseteq C$, add the following to PP:

$$b_A + b_B - 1 \leq b_C.$$

  - For every disjointness $A_1 \sqcap \cdots \sqcap A_n \sqsubseteq \perp, n \geq 2$, add the following to PP:

$$\sum_{i=1}^{n} b_{A_i} - n + 1 \leq b_{\perp}.$$

  - For modelling the negation between C and $\neg C$, add the following to PP:

$$b_C + b_{\neg C} = 1.$$

Mapping of Axioms (II)

- We can show that all the other axioms in $\mathcal{DL}$ ALCQ can be converted to basic axioms by introducing new concepts and basic axioms.
- Example:
  - $A \sqsubseteq A_1 \sqcap \cdots \sqcap A_n \;\equiv\; A \sqsubseteq A_i, \quad i = 1, \ldots, n$
  - $A \sqsubseteq B \sqcup C \;\equiv\; \neg B \sqcap \neg C \sqsubseteq \neg A$
  - $\geq nR \cdot C \sqsubseteq A \;\equiv\; \neg A \sqsubseteq \leq (n-1)R \cdot C$
  - $\leq nR \cdot C \sqsubseteq A \;\equiv\; \neg A \sqsubseteq \geq (n+1)R \cdot C$
  - $A \sqsubseteq \exists R \cdot B \;\equiv\; A \sqsubseteq \geq 1R \cdot B$
  - $\geq 1R \cdot B \sqsubseteq A \;\equiv\; \neg A \sqsubseteq \leq 0R \cdot B$
  - $\cdots$

Integrality

- Branching rule can be defined on binary variables $a^{R'}$.
- However, in all of our instances so far (real and synthetic ontologies), optimal solution returned by the column generation method have been integral!

Conjecture

The polyhedron of $\mathsf{EF}^{\mathsf{LP}}(\mathcal{P}')$ is integral.

- We haven't been able to prove this using the usual sufficient conditions. So, to be continued...

Results & Conclusion

Preliminary Experiments

- We compared our ILP-based reasoner with major OWL reasoners: FaCT++, HermiT, Konclude, and Racer.
- Benchmark ontologies:

Ontology Name	#Axioms	#Concepts	#Roles	#QCRs
canadian-parliament-factions-1	48	21	6	19
canadian-parliament-factions-2	56	24	7	25
canadian-parliament-factions-3	64	27	8	30
canadian-parliament-factions-4	72	30	9	35
canadian-parliament-factions-5	81	34	10	40
C-SAT-exp-ELQ	26	10	4	13
C-UnSAT-exp-ELQ	26	10	4	13
genomic-cds rules-ELQ-fragment-1	716	358	1	357
genomic-cds rules-ELQ-fragment-2	718	359	1	357

Preliminary Observations

- The only reasoners that can classify all variants of the simplest of the first benchmark ontology within the given time limit of 1000s are our ILP-based reasoner and Racer.
- Second benchmark:

	C-SAT-exp-ELQ					C-UnSAT-exp-ELQ				
n	ILP	Fac	Her	Kon	Rac	ILP	Fac	Her	Kon	Rac
40	0.6	TO	TO	TO	0.01	0.63	TO	TO	TO	0.01
20	0.62	TO	TO	TO	0.01	0.80	TO	TO	TO	0.01
10	0.63	TO	TO	TO	0.01	0.99	TO	TO	TO	0.01
5	0.72	6.3	4.4	0.91	0.01	0.74	TO	TO	784	0.01
3	0.62	0.17	0.18	0.33	0.01	0.75	0.25	1.15	1.18	0.01

- Ontologies `genomic-cds rules` contain many concepts using QCRs of the form $= 2\ \texttt{has}.A_i$, with no interaction between (A_i): all reasoners except Racer performed well.

Next...

- This is an ongoing work! So far, we've only focused on mappings and proof of concept.
- The normalization of non-basic axioms is taking longer than expected, however according to the $\mathcal{DL}$ literature should be possible in polynomial time. To be investigated.
- Conjecture, if proved, can simplify the implementation and presentation to non-OR communities.

Recording at GERAD Youtube Channel

HEC MONTRÉAL

Thanks for listening!