Tractable Extensions of the Description Logic $\mathcal{EL}$ with Numerical Datatypes

Despoina Magka, Yevgeny Kazakov and Ian Horrocks

Oxford University Computing Laboratory

April 27, 2010

Outline

$\mathcal{EL}$ Family of Description Logics

- Description logics: logical foundation for W3C ontology languages such as OWL and OWL 2

$\mathcal{EL}$ Family of Description Logics

- Description logics: logical foundation for W3C ontology languages such as OWL and OWL 2
- $\mathcal{EL}$ family of DLs [Baader et al., IJCAI 2003, 2005]:

Example

$\text{YoungParent} \equiv \text{Human} \sqcap \exists \text{hasChild.Human} \sqcap \exists \text{hasAge.}(<, 20)$

$\mathcal{EL}$ Family of Description Logics

- Description logics: logical foundation for W3C ontology languages such as OWL and OWL 2
- $\mathcal{EL}$ family of DLs [Baader et al., IJCAI 2003, 2005]:
  - Conjuction

Example

$\text{YoungParent} \equiv \text{Human} \sqcap \exists \text{hasChild}.\text{Human} \sqcap \exists \text{hasAge}.(<, 20)$

$\mathcal{EL}$ Family of Description Logics

- Description logics: logical foundation for W3C ontology languages such as OWL and OWL 2
- $\mathcal{EL}$ family of DLs [Baader et al., IJCAI 2003, 2005]:
  - Conjuction
  - Existential restriction

Example

YoungParent $\equiv$ Human $\sqcap$ $\exists$hasChild.Human $\sqcap$ $\exists$hasAge.$(<, 20)$

$\mathcal{EL}$ Family of Description Logics

- Description logics: logical foundation for W3C ontology languages such as OWL and OWL 2
- $\mathcal{EL}$ family of DLs [Baader et al., IJCAI 2003, 2005]:
  - Conjuction
  - Existential restriction
  - Domain/range restrictions, complex role inclusions, nominals

Example

$\text{YoungParent} \equiv \text{Human} \sqcap \exists \text{hasChild}.\text{Human} \sqcap \exists \text{hasAge}.(<, 20)$

$\mathcal{EL}$ Family of Description Logics

- Description logics: logical foundation for W3C ontology languages such as OWL and OWL 2
- $\mathcal{EL}$ family of DLs [Baader et al., IJCAI 2003, 2005]:
  - Conjuction
  - Existential restriction
  - Domain/range restrictions, complex role inclusions, nominals
  - Datatypes

Example

$\text{YoungParent} \equiv \text{Human} \sqcap \exists\text{hasChild.Human} \sqcap \exists\text{hasAge}.(<, 20)$

$\mathcal{EL}$ Family of Description Logics

- Description logics: logical foundation for W3C ontology languages such as OWL and OWL 2
- $\mathcal{EL}$ family of DLs [Baader et al., IJCAI 2003, 2005]:
  - Conjuction
  - Existential restriction
  - Domain/range restrictions, complex role inclusions, nominals
  - Datatypes

Example

$\text{YoungParent} \equiv \text{Human} \sqcap \exists \text{hasChild}.\text{Human} \sqcap \exists \text{hasAge}.(<, 20)$

- Sufficient expressivity for ontologies such as SNOMED CT and the Gene Ontology

$\mathcal{EL}$ Family of Description Logics

- Description logics: logical foundation for W3C ontology languages such as OWL and OWL 2
- $\mathcal{EL}$ family of DLs [Baader et al., IJCAI 2003, 2005]:
  - Conjuction
  - Existential restriction
  - Domain/range restrictions, complex role inclusions, nominals
  - Datatypes

Example

YoungParent $\equiv$ Human $\sqcap$ $\exists$hasChild.Human $\sqcap$ $\exists$hasAge.$(<, 20)$

- Sufficient expressivity for ontologies such as SNOMED CT and the Gene Ontology
- Polynomial-time reasoning algorithms for $\mathcal{EL}$

Convex Datatypes

- Dangerous combination of datatypes [Baader et al., 2005]

CONVEX DATATYPES

- Dangerous combination of datatypes [Baader et al., 2005]
- Restrict datatype use to ensure polynomiality

Convex Datatypes

- Dangerous combination of datatypes [Baader et al., 2005]
- Restrict datatype use to ensure polynomiality
- It should not be possible to express disjunction:

Convex Datatypes

- Dangerous combination of datatypes [Baader et al., 2005]
- Restrict datatype use to ensure polynomiality
- It should not be possible to express disjunction:
  $<, >$

Convex Datatypes

- Dangerous combination of datatypes [Baader et al., 2005]
- Restrict datatype use to ensure polynomiality
- It should not be possible to express disjunction:

~~$<, >$~~

Example

A ⊑ B ⊔ C can be expressed by:

$$\begin{array}{lcl} A & \sqsubseteq & \exists F.(<, 5) \\ \exists F.(<, 3) & \sqsubseteq & B \\ \exists F.(>, 2) & \sqsubseteq & C \end{array}$$

Convex Datatypes

- Dangerous combination of datatypes [Baader et al., 2005]
- Restrict datatype use to ensure polynomiality
- It should not be possible to express disjunction:

  ~~$<, >$~~ $<, =$

Example

A $\sqsubseteq$ B $\sqcup$ C can be expressed by:

$$\begin{aligned} A &\sqsubseteq \exists F.(<, 5) \\ \exists F.(=, 4) &\sqsubseteq B \\ \exists F.(<, 4) &\sqsubseteq C \end{aligned}$$

Convex Datatypes

- Dangerous combination of datatypes [Baader et al., 2005]
- Restrict datatype use to ensure polynomiality
- It should not be possible to express disjunction:

~~$<, >$~~ ~~$<, =$~~

Example

$\mathsf{A} \sqsubseteq \mathsf{B} \sqcup \mathsf{C}$ can be expressed by:

$$
\begin{aligned}
\mathsf{A} &\sqsubseteq \exists \mathsf{F}.(<, 5) \\
\exists \mathsf{F}.(=, 4) &\sqsubseteq \mathsf{B} \\
\exists \mathsf{F}.(<, 4) &\sqsubseteq \mathsf{C}
\end{aligned}
$$

Convex Datatypes

- Dangerous combination of datatypes [Baader et al., 2005]
- Restrict datatype use to ensure polynomiality
- It should not be possible to express disjunction:

~~$<, >$~~ ~~$<, =$~~ $>, =$

Example

A ⊑ B ⊔ C can be expressed by:

$$\begin{aligned} A &\sqsubseteq \exists F.(>, 5) \\ \exists F.(=, 6) &\sqsubseteq B \\ \exists F.(>, 6) &\sqsubseteq C \end{aligned}$$

Convex Datatypes

- Dangerous combination of datatypes [Baader et al., 2005]
- Restrict datatype use to ensure polynomiality
- It should not be possible to express disjunction:

~~$<, >$~~ ~~$<, =$~~ ~~$>, =$~~

Example

A $\sqsubseteq$ B $\sqcup$ C can be expressed by:

$$\begin{aligned} A &\sqsubseteq \exists F.(>, 5) \\ \exists F.(=, 6) &\sqsubseteq B \\ \exists F.(>, 6) &\sqsubseteq C \end{aligned}$$

Convex Datatypes

- Dangerous combination of datatypes [Baader et al., 2005]
- Restrict datatype use to ensure polynomiality
- It should not be possible to express disjunction:

~~$<, >$~~ ~~$<, =$~~ ~~$>, =$~~ $=$

Example

A $\sqsubseteq$ B $\sqcup$ C cannot be expressed.

Convex Datatypes

- Dangerous combination of datatypes [Baader et al., 2005]
- Restrict datatype use to ensure polynomiality
- It should not be possible to express disjunction:

  ~~$<, >$~~ ~~$<, =$~~ ~~$>, =$~~ $=$

Example

$A \sqsubseteq B \sqcup C$ cannot be expressed.

Convex Datatypes

- Dangerous combination of datatypes [Baader et al., 2005]
- Restrict datatype use to ensure polynomiality
- It should not be possible to express disjunction:

  $<, >$ $\quad$ $<, =$ $\quad$ $>, =$ $\quad$ $=$
- EL Profile of OWL 2 admits only equality

Convex Datatypes

- Dangerous combination of datatypes [Baader et al., 2005]
- Restrict datatype use to ensure polynomiality
- It should not be possible to express disjunction:
  ~~$<,>$~~ ~~$<,=$~~ ~~$>,=$~~ $=$
- EL Profile of OWL 2 admits only equality

Definition

Convexity property [Baader et al., 2005]: If a restriction implies a disjunction of restrictions, then it also implies one of its disjuncts.

Convex Datatypes

- Dangerous combination of datatypes [Baader et al., 2005]
- Restrict datatype use to ensure polynomiality
- It should not be possible to express disjunction:

  ~~$<, >$~~ ~~$<, =$~~ ~~$>, =$~~ $=$
- EL Profile of OWL 2 admits only equality

Definition

Convexity property [Baader et al., 2005]: If a restriction implies a disjunction of restrictions, then it also implies one of its disjuncts.

Example

Convex case:

If $(x < n) \rightarrow (x < m_1) \vee (x < m_2)$,

then $x < \max(m_1, m_2)$

Example

Not convex case:

$$(x < 5) \rightarrow (x < 2) \vee (x \geq 2)$$
$$(x < 5) \nrightarrow (x < 2)$$
$$(x < 5) \nrightarrow (x \geq 2)$$

Results Overview

- Allow for more extensive datatype use without loosing tractability

Results Overview

- Allow for more extensive datatype use without loosing tractability

- Key idea: distinguish positive and negative occurrences of datatypes

Results Overview

- Allow for more extensive datatype use without loosing tractability
- Key idea: distinguish positive and negative occurrences of datatypes
- Main result: full classification of tractable cases for $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$ and $\mathbb{R}$.

Motivating example

Example

$$\text{Panadol} \sqsubseteq \exists \text{contains}.(\text{Paracetamol} \sqcap \exists \text{mgPerTablet}.(=, 500))$$

$$\text{Patient} \sqcap \exists \text{hasAge}.(<, 6) \sqcap \exists \text{hasPrescription}. \exists \text{contains}.(\text{Paracetamol} \sqcap \exists \text{mgPerTablet}.(>, 250)) \sqsubseteq \bot$$

Motivating example

Example

$$\text{Panadol} \sqsubseteq \exists \text{contains}.(\text{Paracetamol} \sqcap \exists \text{mgPerTablet}.(=, 500))$$

$$\text{Patient} \sqcap \exists \text{hasAge}.(<, 6) \sqcap \exists \text{hasPrescription}.\exists \text{contains}.(\text{Paracetamol} \sqcap \exists \text{mgPerTablet}.(>, 250)) \sqsubseteq \bot$$

- Can Panadol be prescribed to a 3-year-old patient?

Motivating example

Example

$$\text{Panadol} \sqsubseteq \exists\text{contains}.(\text{Paracetamol} \sqcap \exists\text{mgPerTablet}.(=, 500))$$

$$\text{Patient} \sqcap \exists\text{hasAge}.(<, 6) \sqcap \exists\text{hasPrescription}.\exists\text{contains}.(\text{Paracetamol} \sqcap \exists\text{mgPerTablet}.(>, 250)) \sqsubseteq \bot$$

- Can Panadol be prescribed to a 3-year-old patient?
  $\text{X} \sqsubseteq \text{Patient} \sqcap \exists\text{hasAge}.(=, 3) \sqcap \exists\text{hasPrescription}.\text{Panadol}$

Motivating Example

Example

$\text{Panadol} \sqsubseteq \exists\text{contains}.(\text{Paracetamol} \sqcap \exists\text{mgPerTablet}.(=, 500))$

$\text{Patient} \sqcap \exists\text{hasAge}.(<, 6) \sqcap \exists\text{hasPrescription}.$
$\exists\text{contains}.(\text{Paracetamol} \sqcap \exists\text{mgPerTablet}.(>, 250)) \sqsubseteq \bot$

- Can Panadol be prescribed to a 3-year-old patient?
  $X \sqsubseteq \text{Patient} \sqcap \exists\text{hasAge}.(=, 3) \sqcap \exists\text{hasPrescription}.\text{Panadol}$
  Is X satisfiable?

Motivating example

Example

$\text{Panadol} \sqsubseteq \exists \text{contains}.(\text{Paracetamol} \sqcap \exists \text{mgPerTablet}.(=, 500))$

$\text{Patient} \sqcap \exists \text{hasAge}.(<, 6) \sqcap \exists \text{hasPrescription}.$
$\quad \exists \text{contains}.(\text{Paracetamol} \sqcap \exists \text{mgPerTablet}.(>, 250)) \sqsubseteq \bot$

- Can Panadol be prescribed to a 3-year-old patient?
  $X \sqsubseteq \text{Patient} \sqcap \exists \text{hasAge}.(=, 3) \sqcap \exists \text{hasPrescription}.\text{Panadol}$
  Is X satisfiable?
- Equality is used to state a fact such as the content of a drug and the age of a patient

Motivating example

Example

$\text{Panadol} \sqsubseteq \exists \text{contains}.(\text{Paracetamol} \sqcap \exists \text{mgPerTablet}.(=, 500))$

$\text{Patient} \sqcap \exists \text{hasAge}.(<, 6) \sqcap \exists \text{hasPrescription}.$
$\exists \text{contains}.(\text{Paracetamol} \sqcap \exists \text{mgPerTablet}.(>, 250)) \sqsubseteq \bot$

- Can Panadol be prescribed to a 3-year-old patient?
  $X \sqsubseteq \text{Patient} \sqcap \exists \text{hasAge}.(=, 3) \sqcap \exists \text{hasPrescription}.\text{Panadol}$
  Is X satisfiable?
- Equality is used to state a fact such as the content of a drug and the age of a patient
- Inequalities are used to trigger a rule

Motivating Example

Example

$\text{Panadol} \sqsubseteq \exists\text{contains}.(\text{Paracetamol} \sqcap \exists\text{mgPerTablet}.(=, 500))$

$\text{Patient} \sqcap \exists\text{hasAge}.(<, 6) \sqcap \exists\text{hasPrescription}.$
$\exists\text{contains}.(\text{Paracetamol} \sqcap \exists\text{mgPerTablet}.(>, 250)) \sqsubseteq \bot$

- Can Panadol be prescribed to a 3-year-old patient?
  $X \sqsubseteq \text{Patient} \sqcap \exists\text{hasAge}.(=, 3) \sqcap \exists\text{hasPrescription}.\text{Panadol}$
  Is X satisfiable?
- Equality is used to state a fact such as the content of a drug and the age of a patient
- Inequalities are used to trigger a rule
- Positive use of datatypes typically involves equality whereas negative use both equality and inequalities

$\mathcal{EL}^{\bot}$ with Numerical Datatypes

- Concept constructors

	Syntax	Semantics
Concept name	C	$C(x)$
Top	$\top$	$\top$
Bottom	$\bot$	$\bot$
Conjunction	C ⊓ D	$C(x) \wedge D(x)$
Existential restriction	∃R.C	$\exists y : R(x, y) \wedge C(y)$
Datatype restriction	$\exists F.(\substack{<\\=\\>}, n)$	$\exists v \in \mathcal{D} : F(x, v) \wedge v \substack{<\\=\\>} n$

$\mathcal{EL}^{\bot}$ with Numerical Datatypes

- Concept constructors

	Syntax	Semantics
Concept name	C	$C(x)$
Top	$\top$	$\top$
Bottom	$\bot$	$\bot$
Conjunction	C $\sqcap$ D	$C(x) \wedge D(x)$
Existential restriction	$\exists$R.C	$\exists y : R(x,y) \wedge C(y)$
Datatype restriction	$\exists$F.$(\overset{\leq}{\underset{\geq}{=}}, n)$	$\exists v \in \mathcal{D} : F(x,v) \wedge v \overset{\leq}{\underset{\geq}{=}} n$

$\mathcal{D}$ is the numerical domain: we consider $\mathcal{D} = \mathbb{N}, \mathbb{Z}, \mathbb{R}, \mathbb{Q}$

$\mathcal{EL}^{\perp}$ with Numerical Datatypes

- Concept constructors

	Syntax	Semantics
Concept name	C	$C(x)$
Top	$\top$	$\top$
Bottom	$\perp$	$\perp$
Conjunction	C $\sqcap$ D	$C(x) \wedge D(x)$
Existential restriction	$\exists$R.C	$\exists y : R(x,y) \wedge C(y)$
Datatype restriction	$\exists$F.$(\overset{\leq}{\underset{\geq}{=}}, n)$	$\exists v \in \mathcal{D} : F(x,v) \wedge v \overset{\leq}{\underset{\geq}{=}} n$

$\mathcal{D}$ is the numerical domain: we consider $\mathcal{D} = \mathbb{N}, \mathbb{Z}, \mathbb{R}, \mathbb{Q}$

- Axiom

Concept inclusion	C $\sqsubseteq$ D	$C(x) \rightarrow D(x)$

Outline

NORMALIZATION RULES

1. Normalization of the axioms

Normalization Rules

1. Normalization of the axioms

Normal forms	
NF1	$A \sqsubseteq B$
NF2	$A_1 \sqcap A_2 \sqsubseteq B$
NF3	$A \sqsubseteq \exists R.B$
NF4	$\exists R.B \sqsubseteq A$
NF5	$A \sqsubseteq \exists F.(\substack{\leq \\ = \\ \geq}, n)$
NF6	$\exists F.(\substack{\leq \\ = \\ \geq}, n) \sqsubseteq A$

Normalization Rules

1. Normalization of the axioms

Normal forms	
NF1	$A \sqsubseteq B$
NF2	$A_1 \sqcap A_2 \sqsubseteq B$
NF3	$A \sqsubseteq \exists R.B$
NF4	$\exists R.B \sqsubseteq A$
NF5	$A \sqsubseteq \exists F.(\stackrel{\leq}{\stackrel{=}{\geq}}, n)$
NF6	$\exists F.(\stackrel{\leq}{\stackrel{=}{\geq}}, n) \sqsubseteq A$

Example

$\exists R.A \sqsubseteq B \sqcap C$

Normalization Rules

1. Normalization of the axioms

Normal forms	
NF1	$A \sqsubseteq B$
NF2	$A_1 \sqcap A_2 \sqsubseteq B$
NF3	$A \sqsubseteq \exists R.B$
NF4	$\exists R.B \sqsubseteq A$
NF5	$A \sqsubseteq \exists F.(\overset{\leq}{\underset{\geq}{=}}, n)$
NF6	$\exists F.(\overset{\leq}{\underset{\geq}{=}}, n) \sqsubseteq A$

Example

$\exists R.A \sqsubseteq B \sqcap C$

$\rightarrow \quad \exists R.A \sqsubseteq \underline{D} \quad \underline{D} \sqsubseteq B \sqcap C$

Normalization Rules

1. **Normalization** of the axioms

Normal forms	
NF1	$A \sqsubseteq B$
NF2	$A_1 \sqcap A_2 \sqsubseteq B$
NF3	$A \sqsubseteq \exists R.B$
NF4	$\exists R.B \sqsubseteq A$
NF5	$A \sqsubseteq \exists F.(\lesseqqgtr, n)$
NF6	$\exists F.(\lesseqqgtr, n) \sqsubseteq A$

Example

$\exists R.A \sqsubseteq B \sqcap C$

$\rightarrow \exists R.A \sqsubseteq D \quad D \sqsubseteq B \sqcap C$

Normalization Rules

1 Normalization of the axioms

Normal forms	
NF1	$A \sqsubseteq B$
NF2	$A_1 \sqcap A_2 \sqsubseteq B$
NF3	$A \sqsubseteq \exists R.B$
NF4	$\exists R.B \sqsubseteq A$
NF5	$A \sqsubseteq \exists F.(\substack{\leq\\=\\\geq}, n)$
NF6	$\exists F.(\substack{\leq\\=\\\geq}, n) \sqsubseteq A$

Example

$\exists R.A \sqsubseteq B \sqcap C$

$\rightarrow \quad \exists R.A \sqsubseteq D \quad D \sqsubseteq B \sqcap C$

$\rightarrow \quad \exists R.A \sqsubseteq D \quad \underline{D \sqsubseteq B} \quad \underline{D \sqsubseteq C}$

Normalization Rules

1. Normalization of the axioms

Normal forms	
NF1	$A \sqsubseteq B$
NF2	$A_1 \sqcap A_2 \sqsubseteq B$
NF3	$A \sqsubseteq \exists R.B$
NF4	$\exists R.B \sqsubseteq A$
NF5	$A \sqsubseteq \exists F.(\lesseqgtr, n)$
NF6	$\exists F.(\lesseqgtr, n) \sqsubseteq A$

Example

$\exists R.A \sqsubseteq B \sqcap C$

$\rightarrow \exists R.A \sqsubseteq D \quad D \sqsubseteq B \sqcap C$

$\rightarrow \exists R.A \sqsubseteq D \quad D \sqsubseteq B \quad D \sqsubseteq C$

2. Saturation of the axioms under a set of rules

$\mathcal{EL}^{\perp}(\mathcal{D})$ Reasoning Rules

Common rules with $\mathcal{EL}^{++}$ [Baader et al., 2005]

$\mathcal{EL}^{\bot}(\mathcal{D})$ Reasoning Rules

Common rules with $\mathcal{EL}^{++}$ [Baader et al., 2005]

IR1 $\dfrac{}{A \sqsubseteq A}$ **IR2** $\dfrac{}{A \sqsubseteq \top}$ **CR1** $\dfrac{A \sqsubseteq B}{A \sqsubseteq C}$ $B \sqsubseteq C \in \mathcal{O}$

$\mathcal{EL}^\perp(\mathcal{D})$ Reasoning Rules

Common rules with $\mathcal{EL}^{++}$ [Baader et al., 2005]

IR1 $\dfrac{}{A \sqsubseteq A}$ **IR2** $\dfrac{}{A \sqsubseteq \top}$ **CR1** $\dfrac{A \sqsubseteq B}{A \sqsubseteq C}$ $B \sqsubseteq C \in \mathcal{O}$

CR2 $\dfrac{A \sqsubseteq B \quad A \sqsubseteq C}{A \sqsubseteq D}$ $B \sqcap C \sqsubseteq D \in \mathcal{O}$

$\mathcal{EL}^\perp(\mathcal{D})$ Reasoning Rules

Common rules with $\mathcal{EL}^{++}$ [Baader et al., 2005]

IR1 $\dfrac{}{A \sqsubseteq A}$ **IR2** $\dfrac{}{A \sqsubseteq \top}$ **CR1** $\dfrac{A \sqsubseteq B}{A \sqsubseteq C}$ $B \sqsubseteq C \in \mathcal{O}$

CR2 $\dfrac{A \sqsubseteq B \quad A \sqsubseteq C}{A \sqsubseteq D}$ $B \sqcap C \sqsubseteq D \in \mathcal{O}$

CR3 $\dfrac{A \sqsubseteq B}{A \sqsubseteq \exists R.C}$ $B \sqsubseteq \exists R.C \in \mathcal{O}$

$\mathcal{EL}^{\bot}(\mathcal{D})$ Reasoning Rules

Common rules with $\mathcal{EL}^{++}$ [Baader et al., 2005]

IR1 $\dfrac{}{A \sqsubseteq A}$ **IR2** $\dfrac{}{A \sqsubseteq \top}$ **CR1** $\dfrac{A \sqsubseteq B}{A \sqsubseteq C}$ $B \sqsubseteq C \in \mathcal{O}$

CR2 $\dfrac{A \sqsubseteq B \quad A \sqsubseteq C}{A \sqsubseteq D}$ $B \sqcap C \sqsubseteq D \in \mathcal{O}$

CR3 $\dfrac{A \sqsubseteq B}{A \sqsubseteq \exists R.C}$ $B \sqsubseteq \exists R.C \in \mathcal{O}$

CR4 $\dfrac{A \sqsubseteq \exists R.B \quad B \sqsubseteq C}{A \sqsubseteq D}$ $\exists R.C \sqsubseteq D \in \mathcal{O}$

$\mathcal{EL}^{\bot}(\mathcal{D})$ Reasoning Rules

Common rules with $\mathcal{EL}^{++}$ [Baader et al., 2005]

IR1 $$\frac{}{A \sqsubseteq A}$$

IR2 $$\frac{}{A \sqsubseteq \top}$$

CR1 $$\frac{A \sqsubseteq B}{A \sqsubseteq C} \quad B \sqsubseteq C \in \mathcal{O}$$

CR2 $$\frac{A \sqsubseteq B \quad A \sqsubseteq C}{A \sqsubseteq D} \quad B \sqcap C \sqsubseteq D \in \mathcal{O}$$

CR3 $$\frac{A \sqsubseteq B}{A \sqsubseteq \exists R.C} \quad B \sqsubseteq \exists R.C \in \mathcal{O}$$

CR4 $$\frac{A \sqsubseteq \exists R.B \quad B \sqsubseteq C}{A \sqsubseteq D} \quad \exists R.C \sqsubseteq D \in \mathcal{O}$$

CR5 $$\frac{A \sqsubseteq \exists R.B \quad B \sqsubseteq \bot}{A \sqsubseteq \bot}$$

$\mathcal{EL}^{\perp}(\mathcal{D})$ Reasoning Rules for datatypes

ID1 $\dfrac{}{\mathsf{A} \sqsubseteq \bot}$ $\mathsf{A} \sqsubseteq \exists \mathsf{F}.(<, 0) \in \mathcal{O}$

$\mathcal{EL}^{\perp}(\mathcal{D})$ Reasoning Rules for datatypes

ID1 $\dfrac{}{A \sqsubseteq \bot}$ $A \sqsubseteq \exists F.(<, 0) \in \mathcal{O}$

CD1 $\dfrac{A \sqsubseteq B}{A \sqsubseteq \exists F.(\overset{\leq}{\underset{\geq}{=}}, n)}$ $B \sqsubseteq \exists F.(\overset{\leq}{\underset{\geq}{=}}, n) \in \mathcal{O}$

$\mathcal{EL}^{\perp}(\mathcal{D})$ Reasoning Rules for datatypes

ID1 $$\frac{}{A \sqsubseteq \bot} \quad A \sqsubseteq \exists F.(<, 0) \in \mathcal{O}$$

CD1 $$\frac{A \sqsubseteq B}{A \sqsubseteq \exists F.(\substack{\leq \\ = \\ \geq}, n)} \quad B \sqsubseteq \exists F.(\substack{\leq \\ = \\ \geq}, n) \in \mathcal{O}$$

CD2$_{(<,<)}$ $$\frac{A \sqsubseteq \exists F.(<, m)}{A \sqsubseteq B} \quad \exists F.(<, n) \sqsubseteq B \in \mathcal{O}, m \leq n$$

$\mathcal{EL}^{\perp}(\mathcal{D})$ Reasoning Rules for datatypes

ID1 $\quad \dfrac{}{\mathsf{A} \sqsubseteq \bot} \quad \mathsf{A} \sqsubseteq \exists \mathsf{F}.(<, 0) \in \mathcal{O}$

CD1 $\quad \dfrac{\mathsf{A} \sqsubseteq \mathsf{B}}{\mathsf{A} \sqsubseteq \exists \mathsf{F}.(\stackrel{\leq}{\overline{\geq}}, n)} \quad \mathsf{B} \sqsubseteq \exists \mathsf{F}.(\stackrel{\leq}{\overline{\geq}}, n) \in \mathcal{O}$

CD2 $_{(<,<)} \quad \dfrac{\mathsf{A} \sqsubseteq \exists \mathsf{F}.(<, m)}{\mathsf{A} \sqsubseteq \mathsf{B}} \quad \exists \mathsf{F}.(<, n) \sqsubseteq \mathsf{B} \in \mathcal{O}\ , m \leq n$

CD2 $_{(=,<)} \quad \dfrac{\mathsf{A} \sqsubseteq \exists \mathsf{F}.(=, m)}{\mathsf{A} \sqsubseteq \mathsf{B}} \quad \exists \mathsf{F}.(<, n) \sqsubseteq \mathsf{B} \in \mathcal{O}\ , m < n$

$\mathcal{EL}^\perp(\mathcal{D})$ Reasoning Rules for datatypes

ID1 $\quad \dfrac{}{A \sqsubseteq \bot} \quad A \sqsubseteq \exists F.(<, 0) \in \mathcal{O}$

CD1 $\quad \dfrac{A \sqsubseteq B}{A \sqsubseteq \exists F.(\overset{\leq}{\underset{\geq}{=}}, n)} \quad B \sqsubseteq \exists F.(\overset{\leq}{\underset{\geq}{=}}, n) \in \mathcal{O}$

CD2$_{(<,<)}$ $\quad \dfrac{A \sqsubseteq \exists F.(<, m)}{A \sqsubseteq B} \quad \exists F.(<, n) \sqsubseteq B \in \mathcal{O}\ , m \leq n$

CD2$_{(=,<)}$ $\quad \dfrac{A \sqsubseteq \exists F.(=, m)}{A \sqsubseteq B} \quad \exists F.(<, n) \sqsubseteq B \in \mathcal{O}\ , m < n$

CD2$_{(=,=)}$ $\quad \dfrac{A \sqsubseteq \exists F.(=, m)}{A \sqsubseteq B} \quad \exists F.(=, n) \sqsubseteq B \in \mathcal{O}\ , m = n \quad \ldots$

Safe Datatypes

- The algorithm is:
  - sound: all rules derive logical consequences of the premises

Safe Datatypes

- The algorithm is:
  - sound: all rules derive logical consequences of the premises
  - polynomial: only polynomially different axioms are derived

Safe Datatypes

- The algorithm is:
  - sound: all rules derive logical consequences of the premises
  - polynomial: only polynomially different axioms are derived
  - not complete in general

Safe Datatypes

- The algorithm is:
  - sound: all rules derive logical consequences of the premises
  - polynomial: only polynomially different axioms are derived
  - not complete in general
  - complete: provided the datatypes are convex

Safe Datatypes

- The algorithm is:
  - sound: all rules derive logical consequences of the premises
  - polynomial: only polynomially different axioms are derived
  - not complete in general
  - complete: provided the datatypes are convex

Definition

Safety property: If a positive relation implies a disjunction of negative relations, then it also implies one of its disjuncts.

Safe Datatypes

- The algorithm is:
  - sound: all rules derive logical consequences of the premises
  - polynomial: only polynomially different axioms are derived
  - not complete in general
  - complete: provided the datatypes are convex

Definition

Safety property: If a positive relation implies a disjunction of negative relations, then it also implies one of its disjuncts.

Example

$$\text{Panadol} \sqsubseteq \exists\text{contains}.(\text{Paracetamol} \sqcap \exists\text{mgPerTablet}.(=, 500))$$

$$\text{Patient} \sqcap \exists\text{hasAge}.(<, 6) \sqcap \exists\text{hasPrescription}.\ \exists\text{contains}.(\text{Paracetamol} \sqcap \exists\text{mgPerTablet}.(>, 250)) \sqsubseteq \perp$$

Safe Cases for $\mathbb{N}$

Positive relations	Negative relations
$=$	$<, \leq, >, \geq, =$
$<, \leq, >, \geq, =$	$<, \leq$
$<, \leq, >, \geq, =$	$>, \geq$
$>, \geq, =$	$<, \leq, =$

Safe Cases for $\mathbb{N}$

Positive relations	Negative relations
$=$	$<, \leq, >, \geq, =$
$<, \leq, >, \geq, =$	$<, \leq$
$<, \leq, >, \geq, =$	$>, \geq$
$>, \geq, =$	$<, \leq, =$

- All cases are safe:

SAFE CASES FOR $\mathbb{N}$

Positive relations	Negative relations
$=$	$<, \leq, >, \geq, =$
$<, \leq, >, \geq, =$	$<, \leq$
$<, \leq, >, \geq, =$	$>, \geq$
$>, \geq, =$	$<, \leq, =$

- All cases are safe:

If $(x = n) \rightarrow \bigvee_{i=1}^{k} (x \lesseqgtr m_i)$, then $\exists i$ such that $(x \lesseqgtr m_i)$.

Safe Cases for $\mathbb{N}$

Positive relations	Negative relations
$=$	$<, \leq, >, \geq, =$
$<, \leq, >, \geq, =$	$<, \leq$
$<, \leq, >, \geq, =$	$>, \geq$
$>, \geq, =$	$<, \leq, =$

- All cases are safe:

If $(x \lesseqgtr n) \rightarrow (x < m_1) \vee (x < m_2)$, then $x < \max(m_1, m_2)$.

SAFE CASES FOR $\mathbb{N}$

Positive relations	Negative relations
$=$	$<, \leq, >, \geq, =$
$<, \leq, >, \geq, =$	$<, \leq$
$<, \leq, >, \geq, =$	$>, \geq$
$>, \geq, =$	$<, \leq, =$

- All cases are safe:

If $(x \; \substack{\leq \\ = \\ \geq} \; n) \rightarrow (x > m_1) \vee (x > m_2)$, then $x > \min(m_1, m_2)$.

SAFE CASES FOR $\mathbb{N}$

Positive relations	Negative relations
$=$	$<, \leq, >, \geq, =$
$<, \leq, >, \geq, =$	$<, \leq$
$<, \leq, >, \geq, =$	$>, \geq$
$>, \geq, =$	$<, \leq, =$

- All cases are safe:

$$(x > n) \nrightarrow (x < m_1) \vee (x = m_2)$$

SAFE CASES FOR $\mathbb{N}$

Positive relations	Negative relations
$=, <$	$<, \leq, >, \geq, =$
$<, \leq, >, \geq, =$	$<, \leq$
$<, \leq, >, \geq, =$	$>, \geq$
$>, \geq, =$	$<, \leq, =$

- All cases are safe:

$$(x > n) \nrightarrow (x < m_1) \vee (x = m_2)$$

- All cases are maximal:

Safe Cases for $\mathbb{N}$

Positive relations	Negative relations
$=, <$	$<, \leq, >, \geq, =$
$<, \leq, >, \geq, =$	$<, \leq$
$<, \leq, >, \geq, =$	$>, \geq$
$>, \geq, =$	$<, \leq, =$

- All cases are safe:

$$(x > n) \not\rightarrow (x < m_1) \vee (x = m_2)$$

- All cases are maximal:

$$\begin{array}{ccc} (x < 2) & \rightarrow & (x = 1) \vee (x = 0) \\ (x < 2) & \not\rightarrow & (x = 1) \\ (x < 2) & \not\rightarrow & (x = 0) \end{array}$$

Safe Cases for $\mathbb{N}$

Positive relations	Negative relations
$=$	$<, \leq, >, \geq, =$
$<, \leq, >, \geq, =$	$<, \leq, <$
$<, \leq, >, \geq, =$	$>, \geq$
$>, \geq, =$	$<, \leq, =$

- All cases are safe:

$$(x > n) \nrightarrow (x < m_1) \vee (x = m_2)$$

- All cases are maximal:

$$\begin{array}{ccc} (x < 3) & \rightarrow & (x = 2) \vee (x < 2) \\ (x < 3) & \nrightarrow & (x = 2) \\ (x < 3) & \nrightarrow & (x < 2) \end{array}$$

Safe Cases for $\mathbb{N}$

Positive relations	Negative relations
$=$	$<, \leq, >, \geq, =$
$<, \leq, >, \geq, =$	$<, \leq$
$<, \leq, >, \geq, =$	$>, \geq$
$>, \geq, =$	$<, \leq, =, >$

- All cases are safe:

$$(x > n) \not\rightarrow (x < m_1) \vee (x = m_2)$$

- All cases are maximal:

$$\begin{array}{lcl} (x > 2) & \rightarrow & (x = 3) \vee (x > 3) \\ (x > 2) & \not\rightarrow & (x = 3) \\ (x > 2) & \not\rightarrow & (x > 3) \end{array}$$

Safe Cases for $\mathbb{Z}$

Positive relations	Negative relations
$=$	$<, \leq, >, \geq, =$
$<, \leq, >, \geq, =$	$=$
$<, \leq, >, \geq, =$	$<, \leq$
$<, \leq, >, \geq, =$	$>, \geq$
$>, \geq, =$	$<, \leq, =$
$<, \leq, =$	$>, \geq, =$

Safe Cases for $\mathbb{Z}$

Positive relations	Negative relations
$=$	$<, \leq, >, \geq, =$
$<, \leq, >, \geq, =$	$=$
$<, \leq, >, \geq, =$	$<, \leq$
$<, \leq, >, \geq, =$	$>, \geq$
$>, \geq, =$	$<, \leq, =$
$<, \leq, =$	$>, \geq, =$

- Additional datatype restrictions: integers do not have a minimal element such as 0.

SAFE CASES FOR $\mathbb{Z}$

Positive relations	Negative relations
$=$	$<, \leq, >, \geq, =$
$<, \leq, >, \geq, =$	$=$
$<, \leq, >, \geq, =$	$<, \leq$
$<, \leq, >, \geq, =$	$>, \geq$
$>, \geq, =$	$<, \leq, =$
$<, \leq, =$	$>, \geq, =$

- Additional datatype restrictions: integers do not have a minimal element such as 0.
- All cases are safe:

Safe Cases for $\mathbb{Z}$

Positive relations	Negative relations
$=$	$<, \leq, >, \geq, =$
$<, \leq, >, \geq, =$	$=$
$<, \leq, >, \geq, =$	$<, \leq$
$<, \leq, >, \geq, =$	$>, \geq$
$>, \geq, =$	$<, \leq, =$
$<, \leq, =$	$>, \geq, =$

- Additional datatype restrictions: integers do not have a minimal element such as 0.
- All cases are safe:

$$(x < 2) \nrightarrow (x = 1) \vee (x = 0) \vee \ldots$$

Safe Cases for $\mathbb{Q}$ and $\mathbb{R}$

Positive relations	Negative relations
$=$	$<, \leq, >, \geq, =$
$<, \leq, >, \geq, =$	$\leq, =$
$<, \leq, >, \geq, =$	$\geq, =$
$<, \leq, >, \geq, =$	$<, \leq$
$<, \leq, >, \geq, =$	$>, \geq$
$<, >, \geq, =$	$<, \leq, =$
$<, \leq, >, =$	$>, \geq, =$

Safe Cases for $\mathbb{Q}$ and $\mathbb{R}$

Positive relations	Negative relations
$=$	$<, \leq, >, \geq, =$
$<, \leq, >, \geq, =$	$\leq, =$
$<, \leq, >, \geq, =$	$\geq, =$
$<, \leq, >, \geq, =$	$<, \leq$
$<, \leq, >, \geq, =$	$>, \geq$
$<, >, \geq, =$	$<, \leq, =$
$<, \leq, >, =$	$>, \geq, =$

- Density property: between every two different numbers there exists a third one.

Safe Cases for $\mathbb{Q}$ and $\mathbb{R}$

Positive relations	Negative relations
$=$	$<, \leq, >, \geq, =$
$<, \leq, >, \geq, =$	$\leq, =$
$<, \leq, >, \geq, =$	$\geq, =$
$<, \leq, >, \geq, =$	$<, \leq$
$<, \leq, >, \geq, =$	$>, \geq$
$<, >, \geq, =$	$<, \leq, =$
$<, \leq, >, =$	$>, \geq, =$

- Density property: between every two different numbers there exists a third one.
- All cases are safe:

Safe Cases for $\mathbb{Q}$ and $\mathbb{R}$

Positive relations	Negative relations
$=$	$<, \leq, >, \geq, =$
$<, \leq, >, \geq, =$	$\leq, =$
$<, \leq, >, \geq, =$	$\geq, =$
$<, \leq, >, \geq, =$	$<, \leq$
$<, \leq, >, \geq, =$	$>, \geq$
$<, >, \geq, =$	$<, \leq, =$
$<, \leq, >, =$	$>, \geq, =$

- Density property: between every two different numbers there exists a third one.
- All cases are safe:

$$(x < n) \rightarrow_{\mathbb{Z}} (x = n - 1) \vee (x < n - 1)$$

Safe Cases for $\mathbb{Q}$ and $\mathbb{R}$

Positive relations	Negative relations
$=$	$<, \leq, >, \geq, =$
$<, \leq, >, \geq, =$	$\leq, =$
$<, \leq, >, \geq, =$	$\geq, =$
$<, \leq, >, \geq, =$	$<, \leq$
$<, \leq, >, \geq, =$	$>, \geq$
$<, >, \geq, =$	$<, \leq, =$
$<, \leq, >, =$	$>, \geq, =$

- Density property: between every two different numbers there exists a third one.
- All cases are safe:

$$(x < n) \rightarrow_{\mathbb{Z}} (x = n - 1) \vee (x < n - 1)$$

$$(x < n) \not\rightarrow_{\mathbb{R}} (x = n - 1) \vee (x < n - 1)$$

Outline

Results Overview

- Polynomial, sound and complete reasoning procedure for extensions of $\mathcal{EL}^{\perp}$ with numerical datatypes

RESULTS OVERVIEW

- Polynomial, sound and complete reasoning procedure for extensions of $\mathcal{EL}^{\perp}$ with numerical datatypes
- Full classification of safe datatypes for the cases of $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$ and $\mathbb{R}$

Results Overview

- Polynomial, sound and complete reasoning procedure for extensions of $\mathcal{EL}^{\perp}$ with numerical datatypes
- Full classification of safe datatypes for the cases of $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$ and $\mathbb{R}$
- Common safe datatype for $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$ and $\mathbb{R}$:

Results Overview

- Polynomial, sound and complete reasoning procedure for extensions of $\mathcal{EL}^{\perp}$ with numerical datatypes
- Full classification of safe datatypes for the cases of $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$ and $\mathbb{R}$
- Common safe datatype for $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$ and $\mathbb{R}$:

Positive relations	Negative relations
$=$	$<, \leq, >, \geq, =$

Results Overview

- Polynomial, sound and complete reasoning procedure for extensions of $\mathcal{EL}^{\perp}$ with numerical datatypes
- Full classification of safe datatypes for the cases of $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$ and $\mathbb{R}$
- Common safe datatype for $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$ and $\mathbb{R}$:

Positive relations	Negative relations
$=$	$<, \leq, >, \geq, =$

Interesting from a modeling point of view:

- Positive use of datatypes describes precise facts $\rightsquigarrow$ equality

RESULTS OVERVIEW

- Polynomial, sound and complete reasoning procedure for extensions of $\mathcal{EL}^{\perp}$ with numerical datatypes
- Full classification of safe datatypes for the cases of $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$ and $\mathbb{R}$
- Common safe datatype for $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$ and $\mathbb{R}$:

Positive relations	Negative relations
$=$	$<, \leq, >, \geq, =$

Interesting from a modeling point of view:

- Positive use of datatypes describes precise facts $\rightsquigarrow$ equality
- Negative use of datatypes refers to a range of situations $\rightsquigarrow$ both equality and inequalities

RESULTS OVERVIEW

- Polynomial, sound and complete reasoning procedure for extensions of $\mathcal{EL}^{\perp}$ with numerical datatypes
- Full classification of safe datatypes for the cases of $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$ and $\mathbb{R}$
- Common safe datatype for $\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$ and $\mathbb{R}$:

Positive relations	Negative relations
$=$	$<, \leq, >, \geq, =$

Interesting from a modeling point of view:

- Positive use of datatypes describes precise facts $\rightsquigarrow$ equality
- Negative use of datatypes refers to a range of situations $\rightsquigarrow$ both equality and inequalities
- Potential extension of the EL Profile in OWL 2 that currently supports only equality

Future work

- Extend the reasoning algorithm:

FUTURE WORK

- Extend the reasoning algorithm:
    - complex role inclusions

Future work

- Extend the reasoning algorithm:
  - complex role inclusions
  - functional data properties

FUTURE WORK

- Extend the reasoning algorithm:
  - complex role inclusions
  - functional data properties
  - nominals

Future work

- Extend the reasoning algorithm:
  - complex role inclusions
  - functional data properties
  - nominals
  - domain and range restrictions

FUTURE WORK

- Extend the reasoning algorithm:
  - complex role inclusions
  - functional data properties
  - nominals
  - domain and range restrictions
  - Horn $\mathcal{SHIQ}$ [Kazakov, IJCAI, 2009]

FUTURE WORK

- Extend the reasoning algorithm:
  - complex role inclusions
  - functional data properties
  - nominals
  - domain and range restrictions
  - Horn $\mathcal{SHIQ}$ [Kazakov, IJCAI, 2009]
- More fine-grained analysis by also considering which data properties correspond to which relations

FUTURE WORK

- Extend the reasoning algorithm:
  - complex role inclusions
  - functional data properties
  - nominals
  - domain and range restrictions
  - Horn $\mathcal{SHIQ}$ [Kazakov, IJCAI, 2009]
- More fine-grained analysis by also considering which data properties correspond to which relations
- Thank you for your attention! Questions?

FUTURE WORK

- Extend the reasoning algorithm:
  - complex role inclusions
  - functional data properties
  - nominals
  - domain and range restrictions
  - Horn $\mathcal{SHIQ}$ [Kazakov, IJCAI, 2009]
- More fine-grained analysis by also considering which data properties correspond to which relations
- Thank you for your attention! Questions?