History

Phase 1

implementation of systems based on incomplete structural subsumption algorithms

Phase 2

- tableau-based algorithms and complexity results
- first tableau-based systems (Kris, Crack)
- first formal study of optimization methods

Phase 3

- tableau-based algorithms for very expressive DLs
- highly-optimized tableau-based systems (FaCT, Racer)
- relationship to modal logic and FOL

Phase 4

- Web Ontology Language (OWL) based on DL
- industrial-strength reasoners and ontology editors
- light-weight (tractable) DLs

Phase 5

- non-standard reasoning
- ontology management
- semantic extensions

Chapter 2

A Basic Description Logic

$\mathcal{ALC}$

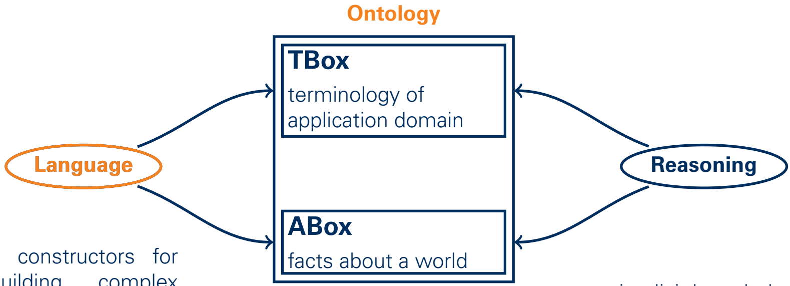
attributive language with complement

Naming Schema

- basic language $\mathcal{AL}$
- additional constructors are denoted by appending a letter
- $\mathcal{C}$ stands for complement

  $\mathcal{ALC}$ is obtained by adding $\neg$ to $\mathcal{AL}$

Structure of Description Logic Systems



- constructors for building complex concepts
- formal logic-based semantics

- implicit knowledge
- practical algorithms

Section 2.1

The Description Language

Definition 2.1 (Syntax of $\mathcal{ALC}$)

Let N_C and N_R two disjoint sets of concept names and role names, respectively.

$\mathcal{ALC}$ (complex) concepts are defined by induction:

- if $A \in N_C$, then A is an $\mathcal{ALC}$ concept
- if C, D are $\mathcal{ALC}$ concepts and $r \in N_R$, then the following are $\mathcal{ALC}$ concepts:
  - $C \sqcap D$ (conjunction)
  - $C \sqcup D$ (disjunction)
  - $\neg C$ (negation)
  - $\exists r.C$ (existential restriction)
  - $\forall r.C$ (value restriction)

Abbreviations

- $\top := A \sqcup \neg A$ (top)
- $\bot := A \sqcap \neg A$ (bottom)

Notation

- concept names are also called atomic concepts
- all other concepts are called complex
- instead of $\mathcal{ALC}$ concept, we often say concept

- A, B stand for concept names
- C, D for (complex) concepts
- r, s for role names

Examples of Concepts

$\text{Hero} \sqcap \text{Female}$

$\forall \text{fights}.\text{Mutant}$

$\text{Rich} \sqcup \neg \text{Human}$

$\exists \text{fights}.\neg \text{Human}$

$\text{Mutant} \sqcap \exists \text{fights}.(\neg \text{Human} \sqcup \forall \text{sidekick}.\text{Female})$

Definition 2.2 (Semantics of $\mathcal{ALC}$)

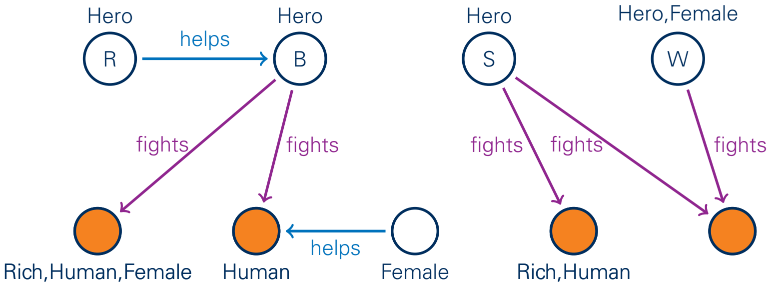
An interpretation $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^\mathcal{I})$ consists of:

- a non-empty domain $\Delta^\mathcal{I}$, and
- an extension mapping $\cdot^\mathcal{I}$ (also called interpretation function):
  - $A^\mathcal{I} \subseteq \Delta^\mathcal{I}$ for all $A \in N_C$ (concepts interpreted as sets)
  - $r^\mathcal{I} \subseteq \Delta^\mathcal{I} \times \Delta^\mathcal{I}$ for all $r \in N_R$ (roles interpreted as binary relations)

The extension mapping is extended to concept descriptions as follows:

$$
\begin{aligned}
(C \sqcap D)^\mathcal{I} &:= C^\mathcal{I} \cap D^\mathcal{I} \\
(C \sqcup D)^\mathcal{I} &:= C^\mathcal{I} \cup D^\mathcal{I} \\
(\neg C)^\mathcal{I} &:= \Delta^\mathcal{I} \setminus C^\mathcal{I} \\
(\exists r.C)^\mathcal{I} &:= \{d \in \Delta^\mathcal{I} \mid \text{ there is } e \in \Delta^\mathcal{I} \text{ with } (d,e) \in r^\mathcal{I} \text{ and } e \in C^\mathcal{I}\} \\
(\forall r.C)^\mathcal{I} &:= \{d \in \Delta^\mathcal{I} \mid \text{ for all } e \in \Delta^\mathcal{I} : (d,e) \in r^\mathcal{I} \text{ implies } e \in C^\mathcal{I}\}
\end{aligned}
$$

Interpretation Example



$(\text{ Hero} \sqcap \exists\text{fights.Human })^{\mathcal{I}} = \{\text{B,S}\}$

$(\text{ Hero} \sqcap \forall\text{fights.(Rich} \sqcup \neg\text{Human) })^{\mathcal{I}} = \{\text{R,S,W}\}$

$(\ \forall\text{helps.Human })^{\mathcal{I}} = \Delta^{\mathcal{I}} \setminus \{\text{R}\}$

DL vs FOL

$\mathcal{ALC}$ can be seen as a fragment of First-order Logic

- concept names are unary predicates
- role names are binary predicates

Interpretations can obviously be seen as first-order interpretations for this signature

concepts then correspond to FOL formulae with one free variable

Let $\phi(x)$ be such a formula with free variable x, and $\mathcal{I}$ an interpretation. The extension of ϕ w.r.t. $\mathcal{I}$ is given by

$$\phi^{\mathcal{I}} := \{d \in \Delta^{\mathcal{I}} \mid \mathcal{I} \models \phi(d)\}$$

Goal

translate $\mathcal{ALC}$ concepts C into FOL formulae $\tau_x(C)$ such that their extensions coincide

Translation to FOL

Syntactic translation $C \rightsquigarrow \tau_x(C)$

- $\tau_x(A) := A(x)$ for $A \in N_C$
- $\tau_x(C \sqcap D) := \tau_x(C) \wedge \tau_x(D)$
- $\tau_x(C \sqcup D) := \tau_x(C) \vee \tau_x(D)$
- $\tau_x(\neg C) := \neg\tau_x(C)$
- $\tau_x(\exists r.C) := \exists y.(r(x,y) \wedge \tau_y C)$ — y new variable different from x
- $\tau_x(\forall r.C) := \forall y.(r(x,y) \rightarrow \tau_y C)$

Example

$$\begin{aligned}\tau_x(\forall r.(A \sqcap \exists s.B)) &= \forall y.(r(x,y) \rightarrow \tau_y(A \sqcap \exists s.B)) \\ &= \forall y.(r(x,y) \rightarrow (A(y) \wedge \exists z.(s(y,z) \wedge B(z))))\end{aligned}$$

Lemma 2.3

C and $\tau_x(C)$ have the same extension; that is,

Proof by induction on structure of C

$$C^{\mathcal{I}} = \{d \in \Delta^{\mathcal{I}} \mid \mathcal{I} \models \tau_x(C)(d)\}.$$

Decidable Fragments of FOL

$\mathcal{ALC}$ can be seen as a fragment of FOL:
each concept C yields a formula $\tau_x(C)$ with one free variable

Decidability

These formulae belong to known decidable subclasses of FOL:

- two variable fragment
- guarded fragment

$$
\begin{aligned}
\tau_x(\forall r.(A \sqcap \exists s.B)) &= \forall y.(r(x,y) \rightarrow \tau_y(A \sqcap \exists s.B)) \\
&= \forall y.(r(x,y) \rightarrow (A(y) \wedge \exists x.(s(y,x) \wedge B(x))))
\end{aligned}
$$

DL vs Modal Logic

$\mathcal{ALC}$ is a syntactic variant of multimodal K

- concept names are propositional variables
- role names are transition relations

multimodal K: several pairs of boxes and diamonds

Concept descriptions C yield modal formulas $\Theta(C)$

- $\Theta(A) := A$ for $A \in N_C$
- $\Theta(C \sqcap D) := \Theta(C) \wedge \Theta(D)$
- $\Theta(C \sqcup D) := \Theta(C) \vee \Theta(D)$
- $\Theta(\neg C) := \neg\Theta(C)$
- $\Theta(\exists r.C) := [r]\Theta(C)$
- $\Theta(\forall r.C) := \langle r\rangle\, \Theta(C)$

C and $\Theta(C)$ have the same semantics:

$\mathcal{I}$ is a Kripke structure

$C^{\mathcal{I}}$ is the set of worlds that make $\Theta(C)$ true in $\mathcal{I}$.

More Expressivity

$\mathcal{ALC}$ is only one example of many description logics that have been studied

many other constructors exist and can be used for KR

Qualified Number Restrictions ($\mathcal{Q}$)

- $(\geq n\, r.C)^{\mathcal{I}} := \{d \in \Delta^{\mathcal{I}} \mid |\{e \mid (d,e) \in r^{\mathcal{I}}, e \in C^{\mathcal{I}}\}| \geq n\}$
  "at least n r-successors that belong to C"
- $(\leq n\, r.C)^{\mathcal{I}} := \{d \in \Delta^{\mathcal{I}} \mid |\{e \mid (d,e) \in r^{\mathcal{I}}, e \in C^{\mathcal{I}}\}| \leq n\}$
  "at most n r-successors that belong to C"

A hero that fights at least two villains, of which at most one is a sidekick

$$\text{Hero} \sqcap (\geq 2\ \text{fights.Villain}) \sqcap (\leq 1\ \text{fights.}(\exists \text{helps.Villain}))$$

More Expressivity

$\mathcal{ALC}$ is only one example of many description logics that have been studied

many other constructors exist and can be used for KR

Number Restrictions ($\mathcal{N}$)

- $(\geq n\ r)^\mathcal{I} := (\geq n\ r.\top)^\mathcal{I} = \{d \in \Delta^\mathcal{I} \mid |\{e \mid (d, e) \in r^\mathcal{I}, e \in C^\mathcal{I}\}| \geq n\}$
  "at least n r-successors"
- $(\leq n\ r)^\mathcal{I} := (\leq n\ r.\top)^\mathcal{I} = \{d \in \Delta^\mathcal{I} \mid |\{e \mid (d, e) \in r^\mathcal{I}, e \in C^\mathcal{I}\}| \leq n\}$
  "at most n r-successors"

More Expressivity

$\mathcal{ALC}$ is only one example of many description logics that have been studied

many other constructors exist and can be used for KR

Inverse Roles ($\mathcal{I}$)

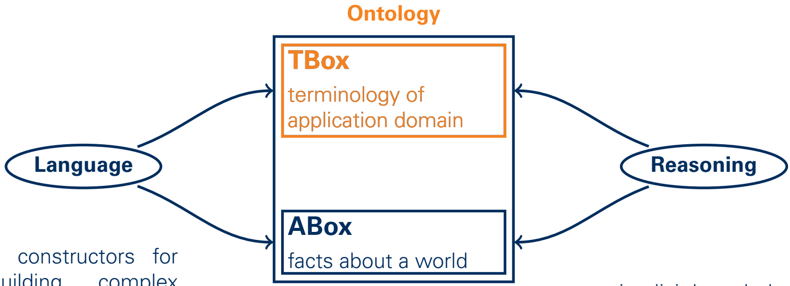
- for a role name r, r^{-1} denotes the inverse:

$$(r^{-1})^{\mathcal{I}} := \{(e, d) \mid (d, e) \in r^{\mathcal{I}}\}$$

A hero that only fights villains with female sidekicks

$$\text{Hero} \sqcap \forall \text{fights}.(\text{Villain} \sqcap \exists \text{helps}^{-1}.\text{Female})$$

Structure of Description Logic Systems



- constructors for building complex concepts
- formal logic-based semantics

- implicit knowledge
- practical algorithms

Section 2.2

Terminological Knowledge

Definition 2.4 (GCIs and TBoxes)

- A general concept inclusion (GCI) is of the form $C \sqsubseteq D$, where C, D are concepts
- A TBox is a finite set of GCIs
- The interpretation $\mathcal{I}$ satisfies the GCI $C \sqsubseteq D$ iff $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$
- $\mathcal{I}$ is a model of the TBox $\mathcal{T}$ iff it satisfies all GCIs in $\mathcal{T}$

Note: this definition is not specific of $\mathcal{ALC}$; applies to any description language

Example

$$\text{Hero} \sqcap \text{Villain} \sqsubseteq \bot$$

$$\text{Hero} \sqcap \forall \text{hasPower}.\bot \sqsubseteq \text{Rich} \sqcup \exists \text{hasSidekick}^{-1}.\text{Rich}$$

Two TBoxes are equivalent if they have the same models

Definition 2.5

A concept definition is of the form $A \equiv C$ where

- A is a concept name
- C is a concept description

The interpretation $\mathcal{I}$ satisfies the concept definition $A \equiv C$ if $A^{\mathcal{I}} = C^{\mathcal{I}}$

$A \equiv C$ abbreviates the two GCIs $A \sqsubseteq C$, $C \sqsubseteq A$

Definition 2.5 (continued)

An acyclic TBox is a finite set of concept definitions that

- does not contain multiple definitions

$$\begin{array}{c} A \equiv C \\ A \equiv D \end{array}$$

- does not contain cyclic definitions (directly or indirectly)

$$\begin{array}{c} A \equiv \exists r.B \\ B \equiv C \\ C \equiv \forall s.A \end{array}$$

The interpretation $\mathcal{I}$ is a model of the acyclic TBox $\mathcal{T}$ is it satisfies all concept definitions in $\mathcal{T}$

A concept name A occurring in $\mathcal{T}$ is a

- defined concept iff there is C such that $A \equiv C \in \mathcal{T}$;
- primitive concept otherwise

Example

$$
\begin{aligned}
\text{Heroine} &\equiv \text{Hero} \sqcap \text{Female} \\
\text{Sidekick} &\equiv \exists \text{helps}.\top \\
\text{Criminal} &\equiv \exists \text{fights}.\text{Hero} \\
\text{MutantCriminal} &\equiv \text{Criminal} \sqcap \forall \text{fights}.\text{Mutant} \\
\text{Superhero} &\equiv \text{Hero} \sqcap (\text{Rich} \sqcup \neg \text{Human} \sqcup \exists \text{hasPower}.\text{SuperPower}) \\
\text{Overlord} &\equiv (\geq 3\ \text{helps}^{-1}.\text{Criminal}) \sqcap \forall \text{fights}.\text{Superhero}
\end{aligned}
$$

Proposition 2.6

For every acyclic TBox $\mathcal{T}$, we can effectively construct an equivalent acyclic TBox $\widehat{\mathcal{T}}$ such that the right-hand sides of concept definitions in $\widehat{\mathcal{T}}$ contain only primitive concepts.

Proof

blackboard

We call $\widehat{\mathcal{T}}$ the expanded version of $\mathcal{T}$

Primitive Interpretations

Given an acyclic TBox $\mathcal{T}$, a primitive interpretation $\mathcal{J}$ for $\mathcal{T}$ consists of a non-empty set $\Delta^{\mathcal{J}}$ together with an extension mapping $\cdot^{\mathcal{J}}$ that maps

- primitive concepts P to sets $P^{\mathcal{J}} \subseteq \Delta^{\mathcal{J}}$
- role names r to binary relations $r^{\mathcal{J}} \subseteq \Delta^{\mathcal{J}} \times \Delta^{\mathcal{J}}$

The interpretation $\mathcal{I}$ is an extension of the primitive interpretation $\mathcal{J}$ iff

- $\Delta^{\mathcal{J}} = \Delta^{\mathcal{I}}$,
- $P^{\mathcal{J}} = P^{\mathcal{I}}$ for all primitive concepts P
- $r^{\mathcal{J}} = r^{\mathcal{I}}$ for all role names r

Corollary 2.7

Let $\mathcal{T}$ be an acyclic TBox. Any primitive interpretation $\mathcal{J}$ has a unique extension to a model of $\mathcal{T}$

Proof

blackboard

Translation to FOL

Any $\mathcal{ALC}$-TBox can be translated into FOL:

$$\tau(\mathcal{T}) := \bigwedge_{C \sqsubseteq D \in \mathcal{T}} \forall x.(\tau_x(C) \rightarrow \tau_x(D))$$

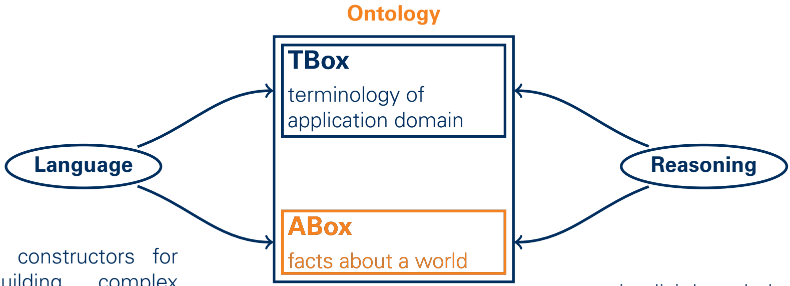
Lemma 2.8

Let $\mathcal{T}$ a TBox and $\tau(\mathcal{T})$ its translation into FOL.
Then $\mathcal{T}$ and $\tau(\mathcal{T})$ have the same models

Proof

blackboard

Structure of Description Logic Systems



- constructors for building complex concepts
- formal logic-based semantics

- implicit knowledge
- practical algorithms

Section 2.3

Assertional Knowledge

Definition 2.9 (Assertions and ABoxes)

An assertion is of the form

$C(a)$ (concept assertion) or $r(a, b)$ (role assertion)

where C is a concept, r a role, and a, b are individual names from a set N_I (disjoint with N_C, N_R)

An ABox is a finite set of assertions

Interpretations $\mathcal{I}$ are extended to assign elements $a^{\mathcal{I}} \in \Delta^{\mathcal{I}}$ to individual names $a \in N_I$

$\mathcal{I}$ is a model of the ABox $\mathcal{A}$ if it satisfies all its assertions:

- $a^{\mathcal{I}} \in C^{\mathcal{I}}$ for all $C(a) \in \mathcal{A}$
- $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in r^{\mathcal{I}}$ for all $r(a, b) \in \mathcal{A}$

Examples of Assertions

Rich(batman)

¬Human(superman)

fights(superman,bizarro)

helps(robin,batman)

Translation to FOL

Any $\mathcal{ALC}$-ABox can be translated into FOL:

$$\tau(\mathcal{A}) := \bigwedge_{C(a)\in\mathcal{A}} \tau_x(C)(a) \wedge \bigwedge_{r(a,b)\in\mathcal{A}} r(a,b)$$

(individual names are viewed as constants)

Lemma 2.10

Let $\mathcal{A}$ an ABox and $\tau(\mathcal{A})$ its translation into FOL.
Then $\mathcal{A}$ and $\tau(\mathcal{A})$ have the same models

Proof

easy (Exercise!)

Definition 2.11

An ontology $\mathcal{O} = (\mathcal{T}, \mathcal{A})$ consists of a TBox $\mathcal{T}$ and an ABox $\mathcal{A}$

The interpretation $\mathcal{I}$ is a model of the ontology $\mathcal{O} = (\mathcal{T}, \mathcal{A})$ iff
it is a model of $\mathcal{T}$ and a model of $\mathcal{A}$

FOL translation: $\tau(\mathcal{O}) := \tau(\mathcal{T}) \wedge \tau(\mathcal{A})$

Lemma 2.12

Let $\mathcal{O}$ be an ontology and $\tau(\mathcal{O})$ its FOL translation. $\mathcal{O}$ and $\tau(\mathcal{O})$ have the same models

Proof

Immediate from Lemmas 2.8 and 2.10

Nominals

We can increase the expressive power of the description language by using individual names as concept constructors

They are interpreted as singleton sets containing only the extension of the individual name

Nominals ($\mathcal{O}$)

Syntax: $\{a\}$ for $a \in N_I$

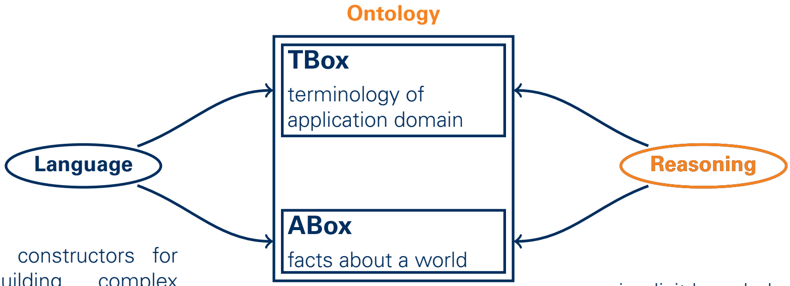
Semantics: $\{a\}^{\mathcal{I}} := \{a^{\mathcal{I}}\}$

Using nominals, ABox assertions can be expressed through GCIs:

$C(a)$ is expressed by $\{a\} \sqsubseteq C$

$r(a, b)$ is expressed by $\{a\} \sqsubseteq \exists r.\{b\}$

Structure of Description Logic Systems



- constructors for building complex concepts
- formal logic-based semantics

- implicit knowledge
- practical algorithms

Section 2.4

Reasoning Problems and Services

Reasoning Problems

Goal: to make implicitly represented knowledge explicit

Definition 2.13 (Terminological Reasoning)

Let $\mathcal{T}$ be a TBox. Terminological reasoning refers to deciding the following problems

Satisfiability:

C is satisfiable w.r.t. $\mathcal{T}$ iff $C^{\mathcal{I}} \neq \emptyset$ for some model $\mathcal{I}$ of $\mathcal{T}$

Subsumption:

C is subsumed by D w.r.t. $\mathcal{T}$ ($C \sqsubseteq_{\mathcal{T}} D$) iff $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ for all models $\mathcal{I}$ of $\mathcal{T}$

Equivalence:

C is equivalent to D w.r.t. $\mathcal{T}$ ($C \equiv_{\mathcal{T}} D$) iff $C^{\mathcal{I}} = D^{\mathcal{I}}$ for all models $\mathcal{I}$ of $\mathcal{T}$

If $\mathcal{T} = \emptyset$ we simply remove the "w.r.t $\mathcal{T}$" from the name

Examples

- $A \sqcap \neg A$ and $\forall r.A \sqcap \exists r.\neg A$ are not satisfiable (unsatisfiable)
- $A \sqcap \neg A$ and $\forall r.A \sqcap \exists r.\neg A$ are equivalent
- $A \sqcap B$ is subsumed by A and by B
- $\exists r.(A \sqcap B)$ is subsumed by $\exists r.A$ and by $\exists r.B$
- $\forall r.(A \sqcap B)$ is equivalent to $\forall r.A \sqcap \forall r.B$
- $\exists r.A \sqcap \forall r.B$ is subsumed by $\exists r.(A \sqcap B)$

Properties of Subsumption

Lemma 2.14

1. The subsumption relation $\sqsubseteq_\mathcal{T}$ is a pre-order on concepts:
   - $C \sqsubseteq_\mathcal{T} C$ (reflexivity)
   - if $C \sqsubseteq_\mathcal{T} D$ and $D \sqsubseteq_\mathcal{T} E$, then $C \sqsubseteq_\mathcal{T} E$ (transitivity)

2. Existential restrictions and value restrictions are monotonic w.r.t. subsumption
   - if $C \sqsubseteq_\mathcal{T} D$, then

$$\exists r.C \sqsubseteq_\mathcal{T} \exists r.D \quad \text{and} \quad \forall r.C \sqsubseteq_\mathcal{T} \forall r.D$$

Proof

blackboard

Note

$\sqsubseteq_\mathcal{T}$ is not a partial order since it is not antisymmetric:

$C \sqsubseteq_\mathcal{T} D$ and $D \sqsubseteq_\mathcal{T} C$ does not imply that $C = D$

(no syntactic equivalence, just semantical)

Definition 2.15 (Assertional Reasoning)

Let $\mathcal{O} = (\mathcal{T}, \mathcal{A})$ be an ontology. The following are assertional reasoning problems

Consistency:

$\mathcal{O}$ is consistent iff there exists a model of $\mathcal{O}$

Instance:

a is an instance of C w.r.t. $\mathcal{O}$ iff $a^{\mathcal{I}} \in C^{\mathcal{I}}$ for all models $\mathcal{I}$ of $\mathcal{O}$

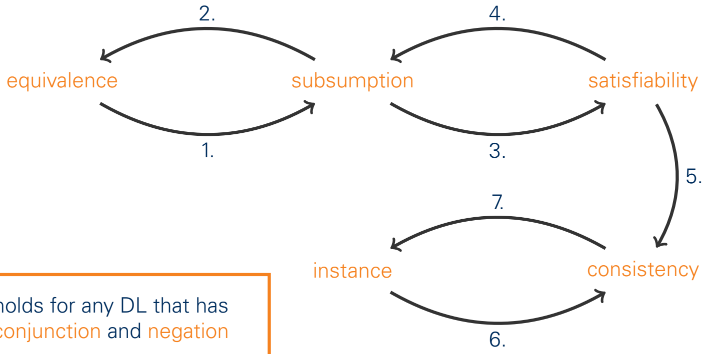
Lemma 2.16

Let $\mathcal{O} = (\mathcal{T}, \mathcal{A})$ be an ontology.
If a is an instance of C w.r.t. $\mathcal{O}$ and $C \sqsubseteq_{\mathcal{T}} D$, then a is an instance of D w.r.t. $\mathcal{O}$

Exercise!

Reductions Between Reasoning Problems

The following polynomial time reductions between the reasoning problems hold



holds for any DL that has
conjunction and negation

Theorem 2.17

Let $\mathcal{O} = (\mathcal{T}, \mathcal{A})$ be an ontology, C, D concepts and $a \in N_I$.

1. $C \equiv_\mathcal{T} D$ iff $C \sqsubseteq_\mathcal{T} D$ and $D \sqsubseteq_\mathcal{T} C$
2. $C \sqsubseteq_\mathcal{T} D$ iff $C \equiv_\mathcal{T} C \sqcap D$
3. $C \sqsubseteq_\mathcal{T} D$ iff $C \sqcap \neg D$ is unsatisfiable w.r.t. $\mathcal{T}$
4. C is satisfiable w.r.t. $\mathcal{T}$ iff $C \not\sqsubseteq_\mathcal{T} \bot$
5. C is satisfiable w.r.t. $\mathcal{T}$ iff $(\mathcal{T}, \{C(a)\})$ is consistent
6. a is an instance of C w.r.t. $\mathcal{O}$ iff $(\mathcal{T}, \mathcal{A} \cup \{\neg C(a)\})$ is inconsistent
7. $\mathcal{O}$ is consistent iff a is not an instance of $\bot$ w.r.t. $\mathcal{O}$

Proof

blackboard

Expansions

Let $\mathcal{O} = (\mathcal{T}, \mathcal{A})$ be an ontology where $\mathcal{T}$ is acyclic, and C a concept

The expanded versions $\widehat{C}$ and $\widehat{\mathcal{A}}$ of C and $\mathcal{A}$ w.r.t. $\mathcal{T}$ are obtained by

replacing all defined concepts occurring in C and $\mathcal{A}$ by their defintions from $\widehat{\mathcal{T}}$

$\mathcal{T}$:

$$\begin{array}{rcl} \text{Woman} & \equiv & \text{Person} \sqcap \text{Female} \\ \text{Criminal} & \equiv & \exists\text{fights.Hero} \\ \text{Minion} & \equiv & \text{Person} \sqcap \exists\text{helps.Criminal} \end{array}$$

C: Woman $\sqcap$ Minion

$$\widehat{C} = \text{Person} \sqcap \text{Female} \sqcap \text{Person} \sqcap \exists\text{helps}.\exists\text{fights.Hero}$$

Expansions

Let $\mathcal{O} = (\mathcal{T}, \mathcal{A})$ be an ontology where $\mathcal{T}$ is acyclic, and C a concept

The expanded versions $\widehat{C}$ and $\widehat{\mathcal{A}}$ of C and $\mathcal{A}$ w.r.t. $\mathcal{T}$ are obtained by

replacing all defined concepts occurring in C and $\mathcal{A}$ by their defintions from $\widehat{\mathcal{T}}$

Proposition 2.18

1. C is satisfiable w.r.t. $\mathcal{T}$ iff $\widehat{C}$ is satisfiable
2. $\mathcal{O} = (\mathcal{T}, \mathcal{A})$ is consistent iff $(\emptyset, \widehat{\mathcal{A}})$ is consistent

Proof

blackboard

similar reductions exist for other reasoning problems

Not a Polynomial Reduction

The expansion of concepts and ABoxes is in general not polynomial

The acyclic TBox $\mathcal{T}$

$$
\begin{aligned}
A_0 &\equiv \forall r.A_1 \sqcap \forall s.A_1 \\
A_1 &\equiv \forall r.A_2 \sqcap \forall s.A_2 \\
&\vdots \\
A_{n-1} &\equiv \forall r.A_n \sqcap \forall s.A_n
\end{aligned}
$$

has n axioms, all of the same size; i.e., the size of $\mathcal{T}$ is linear in n

The expanded version $\widehat{A_0}$ of A_0 contains the name A_n 2^n times!

induction on n

Relationship with FOL

We can translate $\mathcal{ALC}$ reasoning into FOL reasoning

Lemma 2.19

Let $\mathcal{O} = (\mathcal{T}, \mathcal{A})$ be an ontology, C, D be concepts, and $a \in N_I$

1. $C \sqsubseteq_{\mathcal{T}} D$ iff $\tau(\mathcal{T}) \models \forall x.(\tau_x(C)(x) \rightarrow \tau_x(D)(x))$
2. $\mathcal{O}$ is consistent iff $\tau(\mathcal{O})$ is consistent
3. a is an instance of C w.r.t. $\mathcal{O}$ iff $\tau(\mathcal{O}) \models \tau_x(C)(a)$

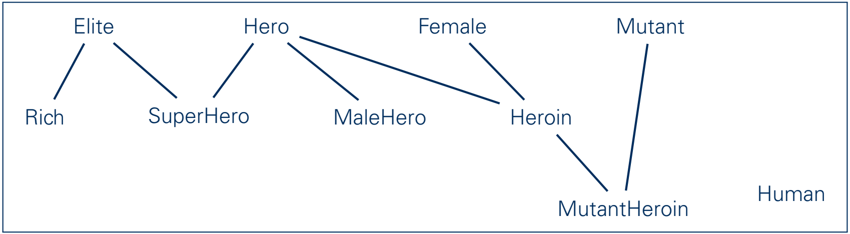
Proof

blackboard

Classification

Computing the subsumption relations between all concept names in $\mathcal{T}$

$$
\begin{aligned}
\text{Heroine} &\equiv \text{Hero} \sqcap \text{Female} \\
\text{MaleHero} &\equiv \text{Hero} \sqcap \neg\text{Female} \\
\text{MutantHeroine} &\equiv \text{Heroine} \sqcap \text{Mutant} \\
\text{Elite} &\equiv \text{Rich} \sqcup \neg\text{Human} \\
\text{Superhero} &\equiv \text{Hero} \sqcap \text{Elite}
\end{aligned}
$$



Realization

Computing the most specific concept names to which an individual belongs

$$
\begin{aligned}
\text{Heroine} &\equiv \text{Hero} \sqcap \text{Female} \\
\text{MaleHero} &\equiv \text{Hero} \sqcap \neg\text{Female} \\
\text{MutantHeroine} &\equiv \text{Heroine} \sqcap \text{Mutant} \\
\text{Elite} &\equiv \text{Rich} \sqcup \neg\text{Human} \\
\text{Superhero} &\equiv \text{Hero} \sqcap \text{Elite}
\end{aligned}
$$

Hero(Superman)

Superman is an instance of

Hero, MaleHero, Elite, Superhero