Semantic Networks and Description Logics
Description Logics – Reasoning Services and Reductions

Knowledge Representation and Reasoning

December 12, 2005
Description Logics – Reasoning Services and Reductions

Motivation

Basic Reasoning Services

Eliminating the TBox

General TBox Reasoning Services

General ABox Reasoning Services

Summary and Outlook
Example TBox & ABox

\[
\begin{align*}
\text{Male} & \equiv \neg \text{Female} \\
\text{Human} & \sqsubseteq \text{Living entity} \\
\text{Woman} & \equiv \text{Human} \sqcap \neg \text{Female} \\
\text{Man} & \equiv \text{Human} \sqcap \text{Male} \\
\text{Mother} & \equiv \text{Woman} \sqcap \exists \text{has-child}.\text{Human} \\
\text{Father} & \equiv \text{Man} \sqcap \exists \text{has-child}.\text{Human} \\
\text{Parent} & \equiv \text{Father} \sqcup \text{Mother} \\
\text{Grandmother} & \equiv \text{Woman} \sqcap \exists \text{has-child}.\text{Parent} \\
\text{Mother-without-daughter} & \equiv \text{Mother} \sqcap \forall \neg \text{has-child}.\text{Male} \\
\text{Mother-with-many-children} & \equiv \text{Mother} \sqcap (\geq 3 \text{has-child})
\end{align*}
\]
Motivation: Reasoning Services

- What do we want to know?
  - We want to check whether the *knowledge base* is reasonable:
    - Is each defined concept in a TBox satisfiable?
    - Is a given TBox satisfiable?
    - Is a given ABox satisfiable?
- What can we *conclude* from the represented knowledge?
  - Is concept $X$ *subsumed* by concept $Y$?
  - Is an object a *instance* of a concept $X$?
- These problems can be *reduced* to logical satisfiability or implication – using the logical semantics.
- We take a different route: We will try to simplify these problems and then we specify *direct inference methods*.
Satisfiability of Concept Descriptions in a TBox

**Motivation:** Given a TBox $\mathcal{T}$ and a concept description $C$, does $C$ make sense, i.e., is $C$ **satisfiable**?

**Test:**
- Does there exist a *model* $I$ of $\mathcal{T}$ such that $C^I \neq \emptyset$?
- Is the formula $\exists x : C(x)$ together with the formulas resulting from the translation of $\mathcal{T}$ satisfiable?

**Example:** *Mother-without-daughter $\sqcap \forall$has-child.Female* is unsatisfiable.
Satisfiability of Concept Descriptions (without a TBox)

- **Motivation**: Given a concept description $C$ in “isolation”, i.e., in an *empty TBox*, does $C$ make sense, i.e., is $C$ *satisfiable*?

- **Test**:
  - Does there exist an *interpretation* $I$ such that $C^I \neq \emptyset$?
  - Is the formula $\exists x : C(x)$ satisfiable?

- **Example**: Woman $\sqcap (\leq 0 \text{ has-child}) \sqcap (\geq 1 \text{ has-child})$ is unsatisfiable.
Eliminating the TBox

Reduction: Getting Rid of the TBox

- We can **reduce** satisfiability in a TBox to simple satisfiability.

- **Idea:**
  - Since TBoxes are *cycle-free*, one can understand a concept definition as a kind of “macro”
  - For a given TBox $\mathcal{T}$ and a given concept description $C$, all defined concept symbols appearing in $C$ can be **expanded** until $C$ contains only undefined concept symbols
  - An **expanded** concept description is then satisfiable iff $C$ is satisfiable in $\mathcal{T}$
  - **Problem:** What do we do with partial definitions (using $\sqsubseteq$)?
Normalized Terminologies

- A terminology is called **normalized** when it does not contain definitions using $\sqsubseteq$.
- In order to **normalize** a terminology, replace

  $$ A \sqsubseteq C $$

  by

  $$ A \equiv A^* \sqcap C, $$

  where $A^*$ is a **fresh** concept symbol (not appearing elsewhere in $\mathcal{T}$).
- If $\mathcal{T}$ is a terminology, the normalized terminology is denoted by $\tilde{\mathcal{T}}$. 
Normalizing is Reasonable

Theorem (Normalization Invariance)

If I is a model of the terminology T, then there exists a model I′ of \( \tilde{T} \) (and vice versa) such that for all concept symbols A appearing in T we have:

\[ A^I = A^{I′}. \]

Proof.

“⇒”: Let I be a model of T. This model should be extended to I′ so that the freshly introduced concept symbols also get interpretations. Assume \((A \sqsubseteq C) \in \tilde{T}\), i.e., we have \((A \sqsupseteq A^* \sqcap C) \in \tilde{T}\). Then set \(A^* I′ = A^I\). I′ obviously satisfies \( \tilde{T} \) and has the same interpretation for all symbols in T.

⇐ Given a model I′ of \( \tilde{T} \), its restriction to symbols of I′ is the interpretation we looked for. \( \square \)
TBox Unfolding

- We say that a *normalized TBox* is **unfolded by one step** when all defined concept symbols on the right sides are replaced by their defining terms.

- **Example:** Mother \( \equiv \) Woman \( \sqcap \ldots \) is unfolded to Mother \( \equiv \) (Human \( \sqcap \) Female) \( \sqcap \ldots \)

- We write \( U(T) \) to denote a one-step unfolding and \( U^n(T) \) to denote an \( n \)-step unfolding.

- We say \( T \) is **unfolded** if \( U(T) = T \).

- We say that \( U^n(T) \) is the **unfolding** of \( T \) if \( U^n(T) = U^{n+1}(T) \). If such an unfolding exists, it is denoted by \( \hat{T} \).
Properties of Unfoldings (1): Existence

Theorem (Existence of unfolded terminology)

For each normalized terminology $\bar{T}$, there exists its unfolding $\hat{T}$.

Proof idea.
The main reason is that terminologies have to be cycle-free. The proof can be done by induction of the definition depth of concepts.
Properties of Unfoldings (2): Equivalence

Theorem (Model equivalence for unfolded terminologies)

$I$ is a model of a normalized terminology $\mathcal{T}$ iff it is a model of $\tilde{\mathcal{T}}$.

Proof Sketch.

$\Rightarrow$: Let $I$ be a model of $\mathcal{T}$. Then it is also a model of $\mathcal{U}(\mathcal{T})$, since on the right side of the definitions only terms with identical interpretations are substituted. However, then it must also be a model of $\tilde{\mathcal{T}}$.

$\Leftarrow$: Let $I$ be a model for $\mathcal{U}(\mathcal{T})$. Clearly, this is also a model of $\mathcal{T}$ (with the same argument as above). This means that any model $\tilde{\mathcal{T}}$ is also a model of $\mathcal{T}$. □
Generating Models

- All concept and role names \textit{not appearing on the left hand side} in a terminology $\mathcal{T}$ are called \textbf{primitive components}.
- Interpretations restricted to primitive components are called \textbf{initial interpretations}.

\textbf{Theorem (Model extension)}

\begin{quote}
For each initial interpretation $\mathcal{J}$ of a normalized TBox, there exists a unique interpretation $\mathcal{I}$ extending $\mathcal{J}$ and satisfying $\mathcal{T}$.
\end{quote}

\textbf{Proof idea.}

Use $\sim \mathcal{T}$ and compute an interpretation for all defined symbols.

\textbf{Corollary (Model existence for TBoxes)}

Each TBox has at least one model.
Unfolding of Concept Descriptions

Similar to the unfolding of TBoxes, we can define **unfolding of concept descriptions**.

We write $\hat{C}$ for the **unfolded version** of $C$.

**Theorem (Satisfiability of unfolded concepts)**

*An concept description $C$ is satisfiable in a terminology $\mathcal{T}$ iff $\hat{C}$ satisfiable in an empty terminology.*

**Proof.**

$\Rightarrow$: trivial.

$\Leftarrow$: Use the interpretation for all the symbols in $\hat{C}$ to generate an initial interpretation of $\mathcal{T}$. Then extend it to a full model $I$ of $\mathcal{T}$. This satisfies $\mathcal{T}$ as well as $\hat{C}$. Since $\hat{C}^I = C^I$, it satisfies also $C$. $\blacksquare$
Subsumption in a TBox

- **Motivation:** Given a terminology $\mathcal{T}$ and two concept descriptions $C$ and $D$, is $C$ subsumed by (or a sub-concept of) $D$ in $\mathcal{T}$ ($C \sqsubseteq_{\mathcal{T}} D$)?

- **Test:**
  - Is $C$ interpreted as a subset of $D$ for all models $I$ of $\mathcal{T}$ ($C^I \subseteq D^I$)?
  - Is the formula $\forall x : (C(x) \rightarrow D(x))$ a logical consequence of the translation of $\mathcal{T}$ to predicate logic?

- **Example:** Grandmother $\sqsubseteq_{\mathcal{T}}$ Mother
Subsumption
(Without a TBox)

► **Motivation**: Given two concept descriptions $C$ and $D$, is $C$ **subsumed by** $D$ regardless of a TBox (or in an **empty TBox**), written $C \sqsubseteq D$?

► **Test**:
  ► Is $C$ interpreted as a subset of $D$ for **all interpretations** $I$ ($C^I \subseteq D^I$)?
  ► Is the formula $\forall x : (C(x) \rightarrow D(x))$ **logically valid**?

► **Example**: Human $\sqcap$ Female $\sqsubseteq$ Human
Reductions

- Subsumption in a TBox can be reduced to subsumption in the empty TBox
- **Normalize** and **unfold** TBox and concept descriptions.
- Subsumption in the empty TBox can be reduced to unsatisfiability
- $C \subseteq D$ iff $C \cap \neg D$ is unsatisfiable
- Unsatisfiability can be reduced to subsumption
- $C$ is unsatisfiable iff $C \subseteq (C \cap \neg C)$
Classifi cation

- **Motivation:** Compute all subsumption relationships (and represent them using only a minimal number of relationships) in order to
  - check the modeling – does the terminology make sense?
  - use the precomputed relations later when subsumption queries have to be answered
  - reduce to subsumption
  - it is a *generalized sorting* problem!
ABox Satisfiability

- **Motivation**: An ABox should *model* the real world, i.e., it should have a model.
- **Test**: Check for a model
- **Example**:

  \[
  X : (\forall r. \neg C) \\
  Y : C \\
  (X, Y) : r
  \]

  is not satisfiable.
ABox Satisfiability in a TBox

- **Motivation**: Is a given ABox $\mathcal{A}$ compatible with the terminology introduced in $\mathcal{T}$?
- **Test**: Is $\mathcal{T} \cup \mathcal{A}$ satisfiable?
- **Example**: If we extend our example with
  
  MARGRET: Woman
  
  (DIANA, MARGRET): has-child,

  then the ABox becomes unsatisfiable in the given TBox.

- **Reduction**:
  - to satisfiability of an ABox
  - *Normalize* terminology, then *unfold* all concept and role descriptions in the ABox
Instance Relations

- **Motivation:** Which additional ABox formulas of the form $a : C$ follow logically from a given ABox and TBox?

- **Test:**
  - Is $a^I \in C^I$ true in all models of $I$ of $\mathcal{T} \cup \mathcal{A}$?
  - Does the formula $C(a)$ logically follow from the translation of $\mathcal{A}$ and $\mathcal{T}$ to predicate logic?

- **Reductions:**
  - Instance relations wrt. an ABox and a TBox can be reduced to instance relations wrt. ABox.
  - Use *normalization* and *unfolding*
  - Instance relations in an ABox can be reduced to ABox unsatisfiability:

\[ a : C \text{ holds in } \mathcal{A} \iff \mathcal{A} \cup \{ a : \neg C \} \text{ is unsatisfiable} \]
Examples

- ELIZABETH: Mother-with-many-children?
  - yes
- WILLIAM: \( \neg \) Female?
  - yes
- ELIZABETH: Mother-without-daughter?
  - no (no CWA!)
- ELIZABETH: Grandmother?
  - no (only male, but not necessarily human!)
Realization

- **Idea:** For a given object $a$, determine the most specialized concept symbols such that $a$ is an instance of these concepts.

- **Motivation:**
  - Similar to *classification*
  - Is the minimal representation of the instance relations (in the set of concept symbols)
  - Will give us faster answers for instance queries!

- **Reduction:** Can be reduced to (a sequence of) instance relation tests.
Retrieval

- **Motivation**: Sometimes, we want to get the set of instances of a concept (as in database queries)
- **Example**: Asking for all instances of the concept Male, we will get the answer CHARLES, ANDREW, EDWARD, WILLIAM.
- **Reduction**: Compute the set of instances by testing the instance relation for each object
- **Implementation**: Realization can be used to speed this up
Reasoning Services – Summary

- Satisfiability of concept descriptions
  - in a given TBox or in an empty TBox
- Subsumption between concept descriptions
  - in a given TBox or in an empty TBox
- Classification
- Satisfiability of an ABox
  - in a given TBox or in an empty TBox
- Instance relations in an ABox
  - in a given TBox or in an empty TBox
- Realization
- Retrieval
Outlook

- How to determine *subsumption* between two concept description (in the empty TBox)?
- How to determine *instance relations/ABox satisfiability*?
- How to implement the mentioned reductions *efficiently*?
- Does normalization and unfolding introduce another source of *computational complexity*?