Proc. 23rd Int. Workshop on Description Logics (DL2010), CEUR-WS 573, Waterloo, Canada, 2010.

Checking Full Satisfiability of Conceptual Models*

Alessandro Artale, Diego Calvanese, and Angélica Ibáñez-García

KRDB Research Centre
Free University of Bozen-Bolzano, Italy
{artale,calvanese}@inf.unibz.it yibanez@gmail.com

Abstract. UML class diagrams (UCDs) are the de-facto standard formalism for the analysis and design of information systems. By adopting formal language techniques to capture constraints expressed by UCDs one can exploit automated reasoning tools to detect relevant properties, such as schema and class satisfiability and subsumption between classes. Among the reasoning tasks of interest, the basic one is detecting *full satisfiability* of a diagram, i.e., whether there exists an instantiation of the diagram where *all* classes and associations of the diagram are non-empty and all the constraints of the diagram are respected. In this paper we establish tight complexity results for full satisfiability for various fragments of UML class diagrams. This investigation shows that the full satisfiability problem is ExpTime-complete in the full scenario, NP-complete if we drop isa between relationships, and NLogSpace-complete if we further drop covering over classes.

1 Introduction

UML (Unified Modeling Language)[1] is the de-facto standard formalism for the analysis and design of information systems. One of the most important components of UML are *class diagrams* (UCDs), which model the domain of interest in terms of objects organized in classes and associations between them (representing relations between class instances). The semantics of UCDs is by now well established, and several works propose to represent it using various kinds of formal languages, e.g., [5,8,7,9,10,4,1,2]. Thus, one can in principle reason on UCDs. The reasoning tasks that one is interested in are, e.g., subsumption between two classes, and satisfiability of a specific class or association in the diagram. Here, we consider *full satisfiability* of a diagram [12], i.e., the fact that there is at least one model of the diagram where each class and association is non-empty. This property is of importance since the presence of some unsatisfiable class or association actually means either that the diagram contains unnecessary information that should be removed, or that there is some modelling error that lead to the loss of satisfiability. In fact, it can be considered as the most fundamental property that should be satisfied by UCDs.

* This work has been partially supported by the EU project Ontorule (ICT-231875).
[1] http://www.omg.org/spec/UML/

The only work that addressed explicitly the complexity of full satisfiability of UCDs is [12], which includes a classification of UCDs based on *inconsistency triggers*. Each inconsistency trigger is a pattern for recognizing possible inconsistencies of the diagram, based on the interaction between different modelling constraints. [12] introduces various algorithms for checking full satisfiability of UCDs with diverse expressive power, together with an analysis of their computational complexity. Full satisfiability of UCDs is computed in ExpTime in the most general case; in NP if association generalization and multiple and overwriting inheritance of attributes is dropped; and in P if the diagrams are further restricted by forbidding covering constraints. According to the results reported in [12], the complexity of checking full satisfiability of UCDs can be reduced if the value types of the attributes associated to sub-classes are sub-types of the value types for the respective attributes associated to the super-classes. The algorithms handling these *restricted* UCDs are claimed to compute full satisfiability respectively in PSpace (instead of ExpTime) and P (instead of NP).

However, our results show that even when attributes are not considered at all in the UCDs, the complexity of the problem does not change. Indeed this paper shows that the full satisfiability problem is ExpTime-complete in the full scenario, NP-complete if we drop isa between relationships, and NLogSpace-complete if we further drop covering over classes. Thus, the complexity of full satisfiability coincides in all cases with that of class satisfiability [1]. Our results build on the formalization of UCDs in terms of DLs given in [4,1]. In fact, our upper bounds are an almost direct consequence of the corresponding upper bounds of the corresponding DL formalization. On the other hand, the obtained lower bounds are more involved, and in some cases require a careful analysis of the corresponding proof for class satisfiability. The results presented here hold also for the Entity-Relationship model and other conceptual models.

The rest of the paper is organized as follows. In Section 2, we briefly introduce the DL $\mathcal{ALC}$, on which we base our results, and show that full satisfiability in $\mathcal{ALC}$ is ExpTime-complete. In Sections 3 and 4, we provide our results on full satisfiability of various variants of UCDs.

2 Full Satisfiability in the Description Logic $\mathcal{ALC}$

We start by studying *full satisfiability* for the DL $\mathcal{ALC}$, one of the basic variants of DLs [3]. We first define the notion of *full satisfiability* of a TBox and then we show that it has the same complexity as classical satisfiability for $\mathcal{ALC}$.

Definition 1 (TBox Full Satisfiability). An $\mathcal{ALC}$ TBox $\mathcal{T}$ is said to be *fully satisfiable* if there exists a model $\mathcal{I}$ of $\mathcal{T}$ such that $A^{\mathcal{I}} \neq \emptyset$, for every atomic concept A in $\mathcal{T}$. We say that $\mathcal{I}$ is a *full model* of $\mathcal{T}$.

Lemma 2. *Concept satisfiability w.r.t. $\mathcal{ALC}$ TBoxes can be linearly reduced to full satisfiability of $\mathcal{ALC}$ TBoxes.*

Proof. Let $\mathcal{T}$ be an $\mathcal{ALC}$ TBox and C an $\mathcal{ALC}$ concept. As pointed out in [6], C is satisfiable w.r.t. $\mathcal{T}$ if and only if $C \sqcap A_{\mathcal{T}}$ is satisfiable w.r.t. the TBox $\mathcal{T}_1$

consisting of the single assertion $A_\mathcal{T} \sqsubseteq \bigsqcap_{C_1 \sqsubseteq C_2 \in \mathcal{T}} (\neg C_1 \sqcup C_2) \sqcap \bigsqcap_{1 \leq i \leq n} \forall P_i.\, A_\mathcal{T}$, where $A_\mathcal{T}$ is a fresh atomic concept and $P_1, \ldots, P_n$ are all the atomic roles in $\mathcal{T}$ and C. In order to reduce the problem to full satisfiability, we extend $\mathcal{T}_1$ to $\mathcal{T}_2 = \mathcal{T}_1 \cup \{A_C \sqsubseteq C \sqcap A_\mathcal{T}\}$, with A_C a fresh atomic concept, and prove that

$$C \sqcap A_\mathcal{T} \text{ is satisfiable w.r.t. } \mathcal{T}_1 \text{ iff } \mathcal{T}_2 \text{ is fully satisfiable.}$$

($\Rightarrow$) Let $\mathcal{I}$ be a model of $\mathcal{T}_1$ such that $(C \sqcap A_\mathcal{T})^\mathcal{I} \neq \emptyset$. We construct an interpretation of $\mathcal{T}_2$, $\mathcal{J} = (\Delta^\mathcal{I} \cup \{d^{top}\}, \cdot^\mathcal{J})$, with $d^{top} \notin \Delta^\mathcal{I}$, such that:

$$\begin{array}{l}
A_\mathcal{T}^\mathcal{J} = A_\mathcal{T}^\mathcal{I}, \qquad A_C^\mathcal{J} = (C \sqcap A_\mathcal{T})^\mathcal{I}, \\
A^\mathcal{J} = A^\mathcal{I} \cup \{d^{top}\} \quad \text{for each atomic concept } A \text{ in } \mathcal{T} \text{ and } C, \\
P^\mathcal{J} = P^\mathcal{I} \quad \text{for each atomic role } P \text{ in } \mathcal{T} \text{ and } C.
\end{array}$$

Obviously, the extension of every atomic concept is non-empty in $\mathcal{J}$. Next, we show that $\mathcal{J}$ is a model of $\mathcal{T}_2$, by relying on the fact (easily proved by structural induction) that $D^\mathcal{I} \subseteq D^\mathcal{J}$, for each subconcept D of concepts in $\mathcal{T}_1$. Then, it is easy to show that $\mathcal{J}$ satisfies the two assertion in $\mathcal{T}_2$:

$$\begin{aligned}
A_\mathcal{T}^\mathcal{J} = A_\mathcal{T}^\mathcal{I} &\subseteq (\bigsqcap_{C_1 \sqsubseteq C_2 \in \mathcal{T}} (\neg C_1 \sqcup C_2) \sqcap \bigsqcap_{1 \leq i \leq n} \forall P_i.A_\mathcal{T})^\mathcal{I} \\
&\subseteq (\bigsqcap_{C_1 \sqsubseteq C_2 \in \mathcal{T}} (\neg C_1 \sqcup C_2) \sqcap \bigsqcap_{1 \leq i \leq n} \forall P_i.\, A_\mathcal{T})^\mathcal{J} \\
A_C^\mathcal{J} = (C \sqcap A_\mathcal{T})^\mathcal{I} &\subseteq (C \sqcap A_\mathcal{T})^\mathcal{J}
\end{aligned}$$

($\Leftarrow$) Conversely, every *full model* $\mathcal{J}$ of $\mathcal{T}_2$ is also a model of $\mathcal{T}_1$ with $(C \sqcap A_\mathcal{T})^\mathcal{J} \neq \emptyset$, as $A_C^\mathcal{J} \subseteq (C \sqcap A_\mathcal{T})^\mathcal{J}$. □

Theorem 3. *Full satisfiability of $\mathcal{ALC}$ TBoxes is* ExpTime*-complete.*

Proof. The ExpTime membership is straightforward, as deciding full satisfiability of an $\mathcal{ALC}$ TBox $\mathcal{T}$ can be reduced to deciding satisfiability of the TBox $\mathcal{T} \cup \bigcup_{1 \leq i \leq n} \{\top \sqsubseteq \exists P'.A_i\}$, where $A_1, \ldots, A_n$ are all the atomic concepts in $\mathcal{T}$, and P' is a fresh atomic role. The ExpTime-hardness follows from Lemma 2. □

We now modify the reduction of Lemma 2 so that it applies also to *primitive* $\mathcal{ALC}^-$ TBoxes, i.e., TBoxes that contain only assertions of the form: $A \sqsubseteq B$, $A \sqsubseteq \neg B$, $A \sqsubseteq B \sqcup B'$, $A \sqsubseteq \forall P.\, B$, $A \sqsubseteq \exists P.\, B$, where A, B, B' are atomic concepts, and P is an atomic role.

Theorem 4. *Full satisfiability of primitive $\mathcal{ALC}^-$ TBoxes is* ExpTime*-complete.*

Proof. The ExpTime membership follows from Theorem 3. For proving the ExpTime-hardness, we use a result in [4] showing that concept satisfiability in $\mathcal{ALC}$ can be reduced to atomic concept satisfiability w.r.t. primitive $\mathcal{ALC}^-$ TBoxes. Let $\mathcal{T}^- = \{A_j \sqsubseteq D_j \mid 1 \leq j \leq m\}$ be a primitive $\mathcal{ALC}^-$ TBox, and A_0

an atomic concept. By Lemma 2, we have that A_0 is satisfiable w.r.t. $\mathcal{T}^-$ if and only if the TBox $\mathcal{T}_2'$ containing the assertions

$$A_{\mathcal{T}^-} \sqsubseteq \bigsqcap_{A_j \sqsubseteq D_j \in \mathcal{T}^-} (\neg A_j \sqcup D_j) \sqcap \bigsqcap_{1 \leq i \leq n} \forall P_i . A_{\mathcal{T}^-}, \qquad A_0' \sqsubseteq A_0 \sqcap A_{\mathcal{T}^-},$$

is fully satisfiable, with $A_{\mathcal{T}^-}, A_0'$ fresh atomic concepts. $\mathcal{T}_2'$ is not a primitive $\mathcal{ALC}^-$ TBox, but it is equivalent to the TBox containing the assertions:

$$\begin{array}{lll} A_0' \sqsubseteq A_{\mathcal{T}^-} & A_{\mathcal{T}^-} \sqsubseteq \neg A_1 \sqcup D_1 & A_{\mathcal{T}^-} \sqsubseteq \forall P_1 . A_{\mathcal{T}^-} \\ & \vdots & \vdots \\ A_0' \sqsubseteq A_0 & A_{\mathcal{T}^-} \sqsubseteq \neg A_m \sqcup D_m & A_{\mathcal{T}^-} \sqsubseteq \forall P_n . A_{\mathcal{T}^-}, \end{array}$$

Finally, to get a primitive $\mathcal{ALC}^-$ TBox, $\mathcal{T}_2^-$, we replace each assertion of the form $A_{\mathcal{T}^-} \sqsubseteq \neg A_j \sqcup D_j$ by $A_{\mathcal{T}^-} \sqsubseteq B_j^1 \sqcup B_j^2$, $B_j^1 \sqsubseteq \neg A_j$, and $B_j^2 \sqsubseteq D_j$, with B_j^1 and B_j^2 fresh atomic concepts, for $j \in \{1, \ldots, m\}$.

We show now that $\mathcal{T}_2'$ is fully satisfiable iff $\mathcal{T}_2^-$ is fully satisfiable:

($\Rightarrow$) Let $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ be a full model of $\mathcal{T}_2'$. We extend $\mathcal{I}$ to an interpretation $\mathcal{J}$ of $\mathcal{T}_2^-$. Let $\Delta^{\mathcal{J}} = \Delta^{\mathcal{I}} \cup \{d^+, d^-\}$, with $\{d^+, d^-\} \cap \Delta^{\mathcal{I}} = \emptyset$, and define $\cdot^{\mathcal{J}}$ as follows:

$$\begin{aligned} A_{\mathcal{T}^-}^{\mathcal{J}} &= A_{\mathcal{T}^-}^{\mathcal{I}}, \quad {A_0'}^{\mathcal{J}} = {A_0'}^{\mathcal{I}}, \\ A^{\mathcal{J}} &= A^{\mathcal{I}} \cup \{d^+\}, \quad \text{for every other atomic concept } A \text{ in } \mathcal{T}_2', \\ {B_j^1}^{\mathcal{J}} &= (\neg A_j)^{\mathcal{J}} \text{ and } {B_j^2}^{\mathcal{J}} = D_j^{\mathcal{J}}, \quad \text{for each } A_{\mathcal{T}^-} \sqsubseteq B_j^1 \sqcup B_j^2 \in \mathcal{T}_2^-, \\ P^{\mathcal{J}} &= P^{\mathcal{I}} \cup \{(d^+, d^+)\}, \quad \text{for each atomic role } P \text{ in } \mathcal{T}_2^-. \end{aligned}$$

It is easy to see that $\mathcal{J}$ is a full model of $\mathcal{T}_2^-$.

($\Leftarrow$) Trivial, since every model of $\mathcal{T}_2^-$ is a model of $\mathcal{T}_2'$. $\square$

3 Full Satisfiability of UML Class Diagrams

Three notions of UCD satisfiability have been proposed in the literature [13,4,12,11]. First, *diagram satisfiability* refers to the existence of a *model*, i.e., an interpretation that satisfies all constraints expressed by the diagram and where at least one class has a nonempty extension. Second, *class satisfiability* refers to the existence of a model of the diagram where the given class has a nonempty extension. Third, we can check whether there is a model of an UML diagram that satisfies *all* classes and *all* relationships in a diagram. This last notion of satisfiability, referred here as *full satisfiability* and introduced in [12] is thus stronger than diagram satisfiability, since a model of a diagram that satisfies all classes is, by definition, also a model of that diagram.

We adopt the formalization of UCDs in terms of DLs as given in [4,1]. For lack of space we give here only a brief overview of such formalization. Classes are formalized by atomic concepts; and relations by roles. Generalization between

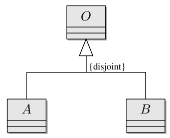

Fig. 1. Encoding of $A \sqsubseteq \neg B$

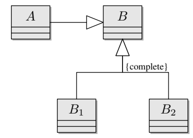

Fig. 2. Encoding of $A \sqsubseteq B_1 \sqcup B_2$

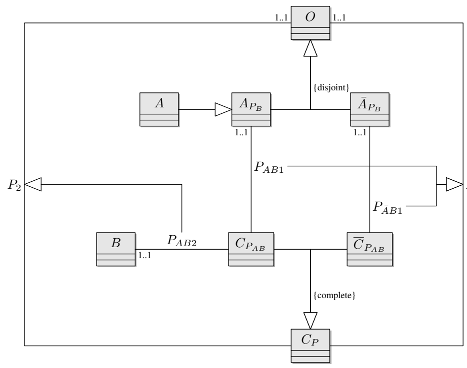

Fig. 3. Encoding of $A \sqsubseteq \forall P.B$

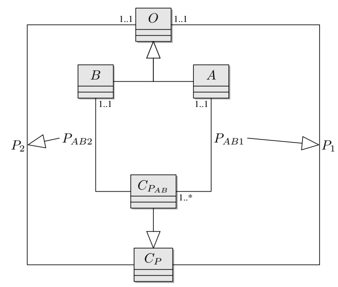

Fig. 4. Encoding of $A \sqsubseteq \exists P.B$

classes (e.g., $C_1 \text{ISA} C_2$) are formalized by concept inclusions ($C_1 \sqsubseteq C_2$); disjointness constrains between two classes C_1 and C_2 by means of axioms of the form $C_1 \sqsubseteq \neg C_2$; and covering constraints by axioms of the form $C \sqsubseteq C_1 \sqcup C_2$. Finally, multiplicity constraints are formalized using qualified number restrictions.

Definition 5 (UML Full Satisfiability). A UCD, $\mathcal{D}$, is *fully satisfiable* if there is an interpretation, $\mathcal{I}$, that satisfies all the constraints expressed in $\mathcal{D}$ and such that $C^{\mathcal{I}} \neq \emptyset$ for every class C in $\mathcal{D}$, and $R^{\mathcal{I}} \neq \emptyset$ for every association R in $\mathcal{D}$. We say that $\mathcal{I}$ is a *full model* of $\mathcal{D}$.

We now address the complexity of full satisfiability for UCDs. For the lower bounds, we use the results presented in Section 2 and reduce full satisfiability of primitive $\mathcal{ALC}^-$ TBoxes to full satisfiability of UCDs. This reduction is based on the ones used in [4,1] for determining the lower complexity bound of schema satisfiability in the extended Entity-Relationship model.

Given a primitive $\mathcal{ALC}^-$ TBox $\mathcal{T}$, construct an UCD $\Sigma(\mathcal{T})$ as follows: for each atomic concept A in $\mathcal{T}$, introduce a class A in $\Sigma(\mathcal{T})$. Additionally, introduce a class O that generalizes (possibly indirectly) all the classes in $\Sigma(\mathcal{T})$ that encode an atomic concept in $\mathcal{T}$. For each atomic role P, introduce a class C_P, which reifies the binary relation P. Further, introduce two functional associations P_1, and P_2 that represent, respectively, the first and second component of P. The assertions in $\mathcal{T}$ are encoded as follows:

– The correspondence of UCDs and DLs gives a straightforward encoding for assertions of the form $A \sqsubseteq B$, $A \sqsubseteq \neg B$, and $A \sqsubseteq B_1 \sqcup B_2$ (see Fig. 1 and Fig. 2).

- For each assertion of the form $A \sqsubseteq \forall P.\, B$, add the auxiliary classes $C_{P_{AB}}$ and $\overline{C}_{P_{AB}}$, and the associations P_{AB1}, $P_{\bar{A}B1}$, and P_{AB2}, and construct the diagram shown in Fig. 3.
- For each assertion of the form $A \sqsubseteq \exists P.\, B$, add the auxiliary class $C_{P_{AB}}$ and the associations P_{AB1} and P_{AB2}, and construct the diagram shown in Fig. 4.

Lemma 6. *A primitive $\mathcal{ALC}^-$ TBox $\mathcal{T}$ is fully satisfiable iff the UCD $\Sigma(\mathcal{T})$, constructed as above, is fully satisfiable.*

Proof. ($\Leftarrow$) Let $\mathcal{J} = (\Delta^{\mathcal{J}}, \cdot^{\mathcal{J}})$ be a full model of $\Sigma(\mathcal{T})$. We construct a full model $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ of $\mathcal{T}$ by taking $\Delta^{\mathcal{I}} = \Delta^{\mathcal{J}}$. Further, for every concept name A and for every atomic role P in $\mathcal{T}$, we define respectively $A^{\mathcal{I}} = A^{\mathcal{J}}$ and[2] $P^{\mathcal{I}} = (P_1^-)^{\mathcal{J}} \circ P_2^{\mathcal{J}}$. Let us show that $\mathcal{I}$ satisfies every assertion in $\mathcal{T}$.

($A \sqsubseteq B$, $A \sqsubseteq \neg B$, and $A \sqsubseteq B_1 \sqcup B_2$): The statement easily follows from the construction of $\mathcal{I}$.

($A \sqsubseteq \forall P.B$): Let $o \in A^{\mathcal{I}} = A^{\mathcal{J}}$ and $o' \in \Delta^{\mathcal{I}} = \Delta^{\mathcal{J}}$, such that $(o, o') \in P^{\mathcal{I}}$. Since $P^{\mathcal{I}} = (P_1^-)^{\mathcal{J}} \circ P_2^{\mathcal{J}}$, there is $o'' \in \Delta^{\mathcal{J}}$ such that $(o, o'') \in (P_1^-)^{\mathcal{J}}$, and $(o'', o') \in P_2^{\mathcal{J}}$. Then, $o'' \in C_P^{\mathcal{J}} = C_{P_{AB}}^{\mathcal{J}} \cup \overline{C}_{P_{AB}}^{\mathcal{J}}$. We claim that $o'' \in C_{P_{AB}}^{\mathcal{J}}$. Suppose otherwise, then there is a unique $o_1 \in \Delta^{\mathcal{J}}$, such that $(o'', o_1) \in P_{\bar{A}B1}^{\mathcal{J}}$ and $o_1 \in \bar{A}_{P_B}^{\mathcal{J}}$. It follows from $P_{\bar{A}B1}^{\mathcal{J}} \subseteq P_1^{\mathcal{J}}$ and by the multiplicity constraint over C_P, that $o_1 = o$. This rises a contradiction, because $o \in A^{\mathcal{J}} \subseteq A_{P_B}^{\mathcal{J}}$ and, $A_{P_B}^{\mathcal{J}}$ and $\bar{A}_{P_B}^{\mathcal{J}}$ are disjoint. Then $o'' \in C_{P_{AB}}^{\mathcal{J}}$. Further, there is a unique $o_2 \in \Delta^{\mathcal{J}}$ with $(o'', o_2) \in P_{AB2}^{\mathcal{J}}$ and $o_2 \in B^{\mathcal{J}}$. From $P_{AB2}^{\mathcal{J}} \subseteq P_2^{\mathcal{J}}$ and the multiplicity constraint on C_P, it follows that $o_2 = o'$. Thus, we have that $o' \in B^{\mathcal{J}} = B^{\mathcal{I}}$, and therefore, $o \in (\forall P.B)^{\mathcal{I}}$.

($A \sqsubseteq \exists P.B$): Let $o \in A^{\mathcal{I}} = A^{\mathcal{J}}$. Then, there is $o' \in \Delta^{\mathcal{J}}$ such that $(o', o) \in P_{AB1}^{\mathcal{J}}$ and $o' \in C_{P_{AB}}^{\mathcal{J}}$. Then, there is $o'' \in \Delta^{\mathcal{J}}$ with $(o', o'') \in P_{AB2}^{\mathcal{J}}$ and $o'' \in B^{\mathcal{J}} = B^{\mathcal{I}}$. Then, since $P_{AB2}^{\mathcal{J}} \subseteq P_2^{\mathcal{J}}$, $P_{AB1}^{\mathcal{J}} \subseteq P_1^{\mathcal{J}}$ and $P^{\mathcal{I}} = (P_1^-)^{\mathcal{J}} \circ P_2^{\mathcal{J}}$, we can conclude that $(o, o'') \in P^{\mathcal{I}}$.

($\Rightarrow$) Let $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ be a full model of $\mathcal{T}$, and let $role(\mathcal{T})$ be the set of role names in $\mathcal{T}$. Extend $\mathcal{I}$ to a legal instantiation $\mathcal{J} = (\Delta^{\mathcal{J}}, \cdot^{\mathcal{J}})$ of $\Sigma(\mathcal{T})$, by assigning suitable extensions to the auxiliary classes and associations in $\Sigma(\mathcal{T})$. Let $\Delta^{\mathcal{J}} = \Delta^{\mathcal{I}} \cup \Gamma \cup \Lambda$, where: $\Lambda = \biguplus_{A \sqsubseteq \forall P.B \in \mathcal{T}} \{a_{AP_B}, a_{\bar{A}P_B}\}$, such that $\Delta^{\mathcal{I}} \cap \Lambda = \emptyset$, and $\Gamma = \biguplus_{P \in role(\mathcal{T})} \Delta_P$, with:

$$\Delta_P = P^{\mathcal{I}} \cup \bigcup_{A \sqsubseteq \forall P.B \in \mathcal{T}} \{(a_{AP_B}, b), (a_{\bar{A}P_B}, \bar{o})\}$$

with b an arbitrary instance of B, and $\bar{o}$ an arbitrary element of $\Delta^{\mathcal{I}}$. We set $O^{\mathcal{J}} = \Delta^{\mathcal{I}} \cup \Lambda$, $A^{\mathcal{J}} = A^{\mathcal{I}}$ for each class A corresponding to an atomic concept in $\mathcal{T}$, and $C_P^{\mathcal{J}} = \Delta_P$ for each $P \in role(\mathcal{T})$. Additionally, the extensions of the associations P_1 and P_2 are defined as follows:

$$P_1^{\mathcal{J}} = \{((o, o'), o) \mid (o, o') \in C_P^{\mathcal{J}}\}, \qquad P_2^{\mathcal{J}} = \{((o, o'), o') \mid (o, o') \in C_P^{\mathcal{J}}\}.$$

We now show that $\mathcal{J}$ is a full model of $\Sigma(\mathcal{T})$.

[2] We use $r_1 \circ r_2$ to denote the composition of two binary relations r_1 and r_2.

1. For the portions of $\Sigma(\mathcal{T})$ due to TBox assertions of the form $A \sqsubseteq B$, $A \sqsubseteq \neg B$, and $A \sqsubseteq B_1 \sqcup B_2$, the statement follows from the construction of $\mathcal{J}$.
2. For each TBox assertion in $\mathcal{T}$ of the form $A \sqsubseteq \forall P.\, B$, let us define

$$
\begin{array}{ll}
A^{\mathcal{J}}_{P_B} = A^{\mathcal{I}} \cup \{a_{A_{P_B}}\}, & \bar{A}^{\mathcal{J}}_{P_B} = O^{\mathcal{J}} \setminus A^{\mathcal{J}}_{P_B}, \\
C^{\mathcal{J}}_{P_{AB}} = \{(o,o') \in C^{\mathcal{J}}_P \mid o \in A^{\mathcal{J}}_{P_B}\}, & \overline{C}^{\mathcal{J}}_{P_{AB}} = \{(o,o') \in C^{\mathcal{J}}_P \mid o \in \bar{A}^{\mathcal{J}}_{P_B}\}, \\
P^{\mathcal{J}}_{AB1} = \{((o,o'),o) \in P^{\mathcal{J}}_1 \mid o \in A^{\mathcal{J}}_{P_B}\}, & P^{\mathcal{J}}_{\bar{A}B1} = \{((o,o'),o) \in P^{\mathcal{J}}_1 \mid o \in \bar{A}^{\mathcal{J}}_{P_B}\}, \\
P^{\mathcal{J}}_{AB2} = \{((o,o'),o') \in P^{\mathcal{J}}_2 \mid o \in A^{\mathcal{J}}_{P_B}\}\,. &
\end{array}
$$

   It is not difficult to see that $\mathcal{J}$ satisfies the fragment of $\Sigma(\mathcal{T})$ as shown in Fig. 3. Further, it is clear that the extension of the classes that encode atomic concepts in $\mathcal{T}$ are non-empty. For the classes A_{P_B}, $\bar{A}_{P_B}$, $C_{P_{AB}}$, and $\overline{C}_{P_{AB}}$ we have that

$$
a_{A_{P_B}} \in A^{\mathcal{J}}_{P_B}, \quad a_{\bar{A}_{P_B}} \in \bar{A}^{\mathcal{J}}_{P_B}, \quad (a_{A_{P_B}}, b) \in C^{\mathcal{J}}_{P_{AB}}, \quad (a_{\bar{A}_{P_B}}, \bar{o}) \in \overline{C}^{\mathcal{J}}_{P_{AB}}\,.
$$

   For the associations P_1, P_2, P_{AB1}, P_{AB2} and $P_{\bar{A}B1}$ we have that

$$
\begin{array}{ll}
((a_{A_{P_B}}, b), a_{A_{P_B}}) \in P^{\mathcal{J}}_{AB1} \subseteq P^{\mathcal{J}}_1, & ((a_{\bar{A}_{P_B}}, \bar{o}), a_{\bar{A}_{P_B}}) \in P^{\mathcal{J}}_{\bar{A}B1}, \\
((a_{A_{P_B}}, b), b) \in P^{\mathcal{J}}_{AB2} \subseteq P^{\mathcal{J}}_2\,. &
\end{array}
$$

3. For each TBox assertion in $\mathcal{T}$ of the form $A \sqsubseteq \exists P.B$, let us define the extensions for the *auxiliary* classes and associations as follows:

$$
\begin{array}{l}
C^{\mathcal{J}}_{P_{AB}} = \{(o,o') \in C^{\mathcal{J}}_P \mid o \in A^{\mathcal{I}} \text{ and } o' \in B^{\mathcal{I}}\}, \\
P^{\mathcal{J}}_{AB1} = \{((o,o'),o) \in P^{\mathcal{J}}_1 \mid (o,o') \in C^{\mathcal{J}}_{P_{AB}}\}, \\
P^{\mathcal{J}}_{AB2} = \{((o,o'),o') \in P^{\mathcal{J}}_2 \mid (o,o') \in C^{\mathcal{J}}_{P_{AB}}\}\,.
\end{array}
$$

   We have that $C^{\mathcal{J}}_{P_{AB}} \neq \emptyset$ as there exists a pair $(a,b) \in \Delta_P$ with $a \in A^{\mathcal{I}}$, and $b \in B^{\mathcal{I}}$. Since $C^{\mathcal{J}}_{P_{AB}} \neq \emptyset$, we have that $P^{\mathcal{J}}_{AB1} \neq \emptyset$ and $P^{\mathcal{J}}_{AB2} \neq \emptyset$. □

Theorem 7. *Full satisfiability of UCDs is* ExpTime*-complete.*

Proof. We establish the upper bound by a reduction to class satisfiability in UCDs, which is known to be ExpTime-complete [4]. Given a UCD $\mathcal{D}$, with classes $C_1, \ldots, C_n$, we construct the UCD $\mathcal{D}'$ by adding to $\mathcal{D}$ a new class $C_\top$ and new associations R_i, for $i \in \{1, \ldots, n\}$. Furthermore, to check that every association is populated we use reification, i.e., we replace each association P in the diagram $\mathcal{D}$ between the classes C_i and C_j (such that neither C_i nor C_j is constrained to participate at least once to P) with a class C_P and two functional associations P_1 and P_2 to represent each component of P. Finally, we add the constraints shown in Fig. 5. Intuitively, we have that if there is a model $\mathcal{I}$ of the extended diagram $\mathcal{D}'$ in which $C^{\mathcal{I}}_\top \neq \emptyset$, then the multiplicity constraint $1..*$ on the association R_P forces the existence of at least one instance o of C_P. By the functionality of P_1 and P_2 there are at least to elements o_i and o_j, such that $o_i \in C^{\mathcal{I}}_i$, $o_j \in C^{\mathcal{I}}_j$, $(o, o_i) \in P^{\mathcal{I}}_1$ and $(o, o_j) \in P^{\mathcal{I}}_2$. Then, one instance of P can be the pair (o_i, o_j). Conversely, if there is a full model $\mathcal{J}$ of $\mathcal{D}$, it is easy to extend it to a model $\mathcal{I}$ of $\mathcal{D}'$ that satisfies $C_\top$.

The ExpTime-hardness follows from Lemma 6 and Theorem 4. □

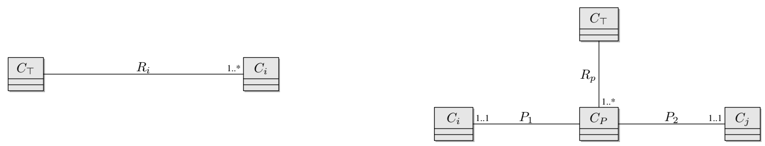

Fig. 5. Reducing UML full satisfiability to class satisfiability

4 Full Satisfiability of Restricted UML Class Diagrams

In this section, we investigate the complexity of the full satisfiability problem for two sub-languages: UML$_{bool}$, which disallows ISA between associations and UML$_{ref}$, where also completeness between classes is forbidden. By building on the techniques used for the satisfiability proofs in [1], we show that also in this case checking for full satisfiability does not change the complexity of the problem.

We first show that deciding full satisfiability for UML$_{bool}$ diagrams is NP-complete. For the lower bound, we provide a polynomial reduction of the 3SAT problem (which is known to be NP-complete) to full satisfiability of UML$_{bool}$ CDs.

Let an instance of 3SAT be given by a set $\phi = \{c_1, \ldots, c_m\}$ of 3-clauses over a finite set Π of propositional variables. Each clause is such that $c_i = \ell_i^1 \vee \ell_i^2 \vee \ell_i^3$, for $i \in \{1, \ldots, m\}$, where each ℓ_j^k is a literal, i.e., a variable or its negation. We construct an UML$_{bool}$ diagram $\mathcal{D}_\phi$ as follows: $\mathcal{D}_\phi$ contains the classes $C_\phi, C_\top$, one class C_i for each clause $c_i \in \phi$, and two classes C_p and $C_{\neg p}$ for each variable $p \in \Pi$. To describe the constraints imposed by $\mathcal{D}_\phi$, we provide the corresponding DL inclusion assertions, since they are more compact to write than an UCD. For every $i \in \{1, \ldots, m\}$, $j \in \{1, 2, 3\}$, and $p \in \Pi$, we have the assertions

$$\begin{array}{rlrlrl} C_\phi \sqsubseteq C_\top, & & C_i \sqsubseteq C_\top, & & C_{l_i^j} \sqsubseteq C_i, & \\ C_p \sqsubseteq C_\top, & & C_\phi \sqsubseteq C_i, & & C_i \sqsubseteq C_{\ell_i^1} \sqcup C_{\ell_i^2} \sqcup C_{\ell_i^3}, & \\ C_{\neg p} \sqsubseteq C_\top, & & C_\top \sqsubseteq C_p \sqcup C_{\neg p}, & & C_{\neg p} \sqsubseteq \neg C_p. & \end{array}$$

Clearly, the size of $\mathcal{D}_\phi$ is polynomial in the size of ϕ.

Lemma 8. *A set ϕ of 3-clauses is satisfiable if and only if the UML$_{bool}$ class diagram $\mathcal{D}_\phi$, constructed as above, is fully satisfiable.*

Proof. ($\Rightarrow$) Let $\mathcal{J} \models \phi$. Define an interpretation $\mathcal{I} = (\{0, 1\}, \cdot^\mathcal{I})$, with

$$C_\top^\mathcal{I} = \{0, 1\}$$

$$C_\ell^\mathcal{I} = \begin{cases} \{1\}, & \text{if } \mathcal{J} \models \ell \\ \{0\}, & \text{otherwise} \end{cases} \qquad \begin{array}{l} C_i^\mathcal{I} = C_{\ell_i^1}^\mathcal{I} \cup C_{\ell_i^2}^\mathcal{I} \cup C_{\ell_i^3}^\mathcal{I}, \quad \text{for } c_i = \ell_i^1 \vee \ell_i^2 \vee \ell_i^3 \\ C_\phi^\mathcal{I} = C_1^\mathcal{I} \cap \cdots \cap C_m^\mathcal{I}. \end{array}$$

Clearly, $C^\mathcal{I} \neq \emptyset$ for every class C representing a clause or a literal, and for $C = C_\top$. Moreover, as at least one literal ℓ_i^j in each clause is such that $\mathcal{J} \models \ell_i^j$, then $1 \in C_i^\mathcal{I}$ for every $i \in \{1, \ldots, m\}$, and therefore $1 \in C_\phi^\mathcal{I}$. It is straightforward to check that $\mathcal{I}$ satisfies $\mathcal{T}$.

($\Leftarrow$) Let $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ be a full model of $\mathcal{D}_\phi$. We construct a model $\mathcal{J}$ of ϕ by taking an element $o \in C_\phi^{\mathcal{I}}$, and setting, for every variable $p \in \Pi$, $\mathcal{J} \models p$ if and only if $o \in C_p^{\mathcal{I}}$. Let us show that $\mathcal{J} \models \phi$. Indeed, for each $i \in \{1, \ldots, m\}$, since $o \in C_\phi^{\mathcal{I}}$ and by the generalization $C_\phi \sqsubseteq C_i$, we have that $o \in C_i^{\mathcal{I}}$, and by the completeness constraint $C_i \sqsubseteq C_{\ell_i^1} \sqcup C_{\ell_i^2} \sqcup C_{\ell_i^3}$, there is some $j_i \in \{1, 2, 3\}$ such that $o \in C_{\ell_i^{j_i}}$. If $\ell_i^{j_i}$ is a variable, then $\mathcal{J} \models \ell_i^{j_i}$ by construction, and thus $\mathcal{J} \models c_i$. Otherwise, if $\ell_i^{j_i} = \neg p$ for some variable p, then, by the disjointness constraint $C_{\neg p} \sqsubseteq \neg C_p$, we have that $o \notin C_p^{\mathcal{I}}$. Thus, $\mathcal{J} \models \neg p$, and therefore, $\mathcal{J} \models c_i$. □

Theorem 9. *Full satisfiability of UML$_{bool}$ is* NP-*complete*

Proof. The NP-hardness follows from Lemma 8. To prove the NP upper bound, we reduce full satisfiability to class satisfiability, which, for the case of UML$_{bool}$, is known to be in NP [1]. We use a similar encoding as the one used in the proof of Theorem 7 (see Fig. 5). □

We turn now to UML$_{ref}$ class diagrams and show that full satisfiability in this case is NLogSpace-complete. We provide a reduction of the reachability problem on (acyclic) directed graphs, which is known to be NLogSpace-complete (see e.g., [14]) to the complement of full satisfiability of UML$_{ref}$ CDs.

Let $G = (V, E, s, t)$ be an instance of reachability, where V is a set of vertices, $E \subseteq V \times V$ is a set of directed edges, s is the start vertex, and t the terminal vertex. We construct an UML$_{ref}$ diagram $\mathcal{D}_G$ from G as follows:

- $\mathcal{D}_G$ has two classes C_v^1 and C_v^2, for each vertex $v \in V \setminus \{s\}$, and one class C_s corresponding to the start vertex s.
- For each edge $(u, v) \in E$ with $u \neq s$ and $v \neq s$, $\mathcal{D}_G$ contains the following constraints (again expressed as DL inclusion assertions):

$$C_u^1 \sqsubseteq C_v^1, \qquad\qquad C_u^2 \sqsubseteq C_v^2.$$

- For each edge $(s, v) \in E$, $\mathcal{D}_G$ contains the following constraints:

$$C_s \sqsubseteq C_v^1, \qquad\qquad C_s \sqsubseteq C_v^2.$$

- For each edge $(u, s) \in E$, $\mathcal{D}_G$ contains the following constraints:

$$C_u^1 \sqsubseteq C_s, \qquad\qquad C_u^2 \sqsubseteq C_s.$$

- The classes C_t^1 and C_t^2 are constrained to be disjoint in $\mathcal{D}$, expressed by:

$$C_t^1 \sqsubseteq \neg C_t^2.$$

The following lemma establishes the correctness of the reduction.

Lemma 10. *t is reachable from s in G iff $\mathcal{D}_G$ is not fully satisfiable.*

Proof. ($\Rightarrow$) Let $\pi = v_1, \ldots, v_n$ be a path in G with $v_1 = s$ and $v_n = t$. We claim that the class C_s in the constructed diagram $\mathcal{D}_G$ is unsatisfiable. Suppose otherwise, that there is a model $\mathcal{I}$ of $\mathcal{D}_G$ with $o \in C_s^{\mathcal{I}}$, for some $o \in \Delta^{\mathcal{I}}$. From π, the construction yields a number of generalization constraints in $\mathcal{D}_G$ such that the following holds:

$$C_s^{\mathcal{I}} \subseteq \cdots \subseteq C_t^{1\,\mathcal{I}} \qquad\qquad C_s^{\mathcal{I}} \subseteq \cdots \subseteq C_t^{2\,\mathcal{I}}$$

From this we obtain that $o \in (C_t^1)^{\mathcal{I}}$ and $o \in (C_t^2)^{\mathcal{I}}$, which violates the disjointness between the classes C_t^1 and C_t^2, in contradiction to $\mathcal{I}$ being a model of $\mathcal{D}_G$. Hence, C_s is unsatisfiable, and therefore $\mathcal{D}_G$ is not fully satisfiable.

($\Leftarrow$) Assume that t is not reachable from s in G. We construct a full model $\mathcal{I}$ of $\mathcal{D}_G$. Let $\Delta^{\mathcal{I}} = \{d_s\} \cup \bigcup_{v \in V \setminus \{s\}} \{d_v^1, d_v^2\}$. Define inductively a sequence of interpretations as follows:

$\mathcal{I}^0 := \left(\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}^0}\right)$, such that:

$$C_s^{\mathcal{I}^0} := \{d_s\}, \quad C_v^{i\,\mathcal{I}^0} := \{d_v^i\}, \, \forall i \in \{1,2\}, v \in V \setminus \{s\}.$$

$\mathcal{I}^{n+1} := \left(\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}^{n+1}}\right)$, such that:

$$\begin{aligned} C_s^{\mathcal{I}^{n+1}} &:= C_s^{\mathcal{I}^n} \cup \bigcup_{(u,s) \in E} \left(C_u^{1\,\mathcal{I}^n} \cup C_u^{2\,\mathcal{I}^n}\right) \\ C_v^{i\,\mathcal{I}^{n+1}} &:= C_v^{i\,\mathcal{I}^n} \cup \bigcup_{(u,v) \in E,\, u \neq s} C_u^{i\,\mathcal{I}^n} \cup \bigcup_{(s,v) \in E} C_s^{\mathcal{I}^n} \end{aligned}$$

The definition induces a monotone operator over a complete lattice, and hence it has a fixed point. Let $\mathcal{I}$ be defined by such a fixed point. It is easy to check that $\mathcal{I}$ is such that for all $i \in \{1,2\}$, and $u, v \in V \setminus \{s\}$ the following holds:

1. For each class C_v^i, we have that $d_v^i \in C_v^{i\,\mathcal{I}}$.
2. $d_s \in C_s^{\mathcal{I}}$.
3. For all $d \in \Delta^{\mathcal{I}}$, $d \in C_u^{i\,\mathcal{I}}$ implies $d \in C_v^{i\,\mathcal{I}}$ iff v is reachable from u in G.
4. For all $d_u^i \in \Delta^{\mathcal{I}}$, $d_u^i \in C_v^{j\,\mathcal{I}}$ for $i \neq j$ iff s is reachable from u in G, and v is reachable from s in G.
5. $d_s \in C_v^{i\,\mathcal{I}}$ iff v is reachable from s in G.

From (1) and (2) we have that all classes in $\mathcal{D}_G$ are populated in $\mathcal{I}$. It remains to show that $\mathcal{I}$ satisfies $\mathcal{D}_G$. A generalization between the classes C_u^i and C_v^i corresponds to the edge $(u,v) \in E$. This means that v is reachable from u in G, and therefore, by (3) we have that $C_u^{i\,\mathcal{I}} \subseteq C_v^{i\,\mathcal{I}}$. A similar argument holds for generalizations involving the class C_s. Furthermore, the classes C_t^1 and C_t^2 are disjoint under $\mathcal{I}$. To show this, suppose that there is an element $d \in \Delta^{\mathcal{I}}$ such that $d \in C_t^{1\mathcal{I}} \cap C_t^{2\mathcal{I}}$. Then by (5), $d \neq d_s$, as t is not reachable from s. Moreover, $d \neq d_v^i$ for all $i \in \{1,2\}$ and $v \in V \setminus \{s\}$. Indeed, suppose w.l.o.g. that $i = 1$. Then, by (4), $d_v^1 \in C_t^{2\mathcal{I}}$ iff s is reachable from v, and t is reachable from s, which leads to a contradiction. Hence, $C_t^{1\mathcal{I}} \cap C_t^{2\mathcal{I}} = \emptyset$. $\square$

Language	Classes			Associations			Complexity
	ISA	disjoint	complete	ISA	multiplicity	refinement	
UML	✓	✓	✓	✓	✓	✓	EXPTIME
UML_{bool}	✓	✓	✓	✗	✓	✓	NP
UML_{ref}	✓	✓	✗	✗	✓	✓	NLOGSPACE

Table 1. Complexity results for full satisfiability in UML

Theorem 11. *Full-satisfiability of UML_{ref} class diagrams is* NLOGSPACE-*complete.*

Proof. The NLOGSPACE membership follows from the NLOGSPACE membership of class satisfiability [1], and a reduction similar to the one used in Theorem 9. Since NLOGSPACE = coNLOGSPACE (by the Immerman-Szelepcsényi theorem; see, e.g., [14]), and as the above reduction is logspace bounded, it follows that full consistency of UML_{ref} class diagrams is NLOGSPACE-hard. □

5 Conclusions

This paper investigates the problem of *full satisfiability* in the context of UML class diagrams, i.e., whether there is at least one model of the diagram where each class and association is non-empty. Our results (reported in Table 1) show that the complexity of full satisfiability matches the complexity of the classical class diagram satisfiability check. We show a similar result also for the problem of checking the full satisfiability of a TBox expressed in the description logic $\mathcal{ALC}$.

References

1. A. Artale, D. Calvanese, R. Kontchakov, V. Ryzhikov, and M. Zakharyaschev. Reasoning over extended ER models. In *Proc. of the 26th Int. Conf. on Conceptual Modeling (ER 2007)*, volume 4801 of *Lecture Notes in Computer Science*, pages 277–292. Springer, 2007.
2. A. Artale, D. Calvanese, R. Kontchakov, and M. Zakharyaschev. The *DL-Lite* family and relations. *J. of Artificial Intelligence Research*, 36:1–69, 2009.
3. F. Baader, D. Calvanese, D. McGuinness, D. Nardi, and P. F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation and Applications*. Cambridge University Press, 2003.
4. D. Berardi, D. Calvanese, and G. De Giacomo. Reasoning on UML class diagrams. *Artificial Intelligence*, 168(1–2):70–118, 2005.
5. A. Borgida. Description logics in data management. *IEEE Trans. on Knowledge and Data Engineering*, 7(5):671–682, 1995.
6. M. Buchheit, F. M. Donini, and A. Schaerf. Decidable reasoning in terminological knowledge representation systems. *J. of Artificial Intelligence Research*, 1:109–138, 1993.

7. D. Calvanese, M. Lenzerini, and D. Nardi. Unifying class-based representation formalisms. *J. of Artificial Intelligence Research*, 11:199–240, 1999.
8. T. Clark and A. S. Evans. Foundations of the Unified Modeling Language. In D. Duke and A. Evans, editors, *Proc. of the 2nd Northern Formal Methods Workshop*. Springer, 1997.
9. A. Evans, R. France, K. Lano, and B. Rumpe. Meta-modelling semantics of UML. In H. Kilov, editor, *Behavioural Specifications for Businesses and Systems*, chapter 2. Kluwer Academic Publishers, 1999.
10. D. Harel and B. Rumpe. Modeling languages: Syntax, semantics and all that stuff. Technical Report MCS00-16, The Weizmann Institute of Science, Rehovot, Israel, 2000.
11. M. Jarrar and S. Heymans. Towards pattern-based reasoning for friendly ontology debugging. *Int. J. on Artificial Intelligence Tools*, 17(4):607–634, 2008.
12. K. Kaneiwa and K. Satoh. Consistency checking algorithms for restricted UML class diagrams. In *Proc. of the 4th Int. Symp. on Foundations of Information and Knowledge Systems (FoIKS 2006)*, pages 219–239, 2006.
13. M. Lenzerini and P. Nobili. On the satisfiability of dependency constraints in entity-relationship schemata. *Information Systems*, 15(4):453–461, 1990.
14. C. H. Papadimitriou. *Computational Complexity*. Addison Wesley Publ. Co., 1994.