Reasoning over Extended ER Models

A. Artale[1], D. Calvanese[1], R. Kontchakov[2], V. Ryzhikov[1], M. Zakharyaschev[2]

[1] Faculty of Computer Science
Free University of Bozen-Bolzano
I-39100 Bolzano, Italy
lastname@inf.unibz.it

[2] School of Comp. Science and Inf. Sys.
Birkbeck College
London WC1E 7HX, UK
{roman,michael}@dcs.bbk.ac.uk

Abstract. We investigate the computational complexity of reasoning over various fragments of the Extended Entity-Relationship (EER) language, which includes a number of constructs: ISA between entities and relationships, disjointness and covering of entities and relationships, cardinality constraints for entities in relationships and their refinements as well as multiplicity constraints for attributes. We extend the known EXPTIME-completeness result for UML class diagrams [5] and show that reasoning over EER diagrams with ISA between relationships is EXPTIME-complete even without relationship covering. Surprisingly, reasoning becomes NP-complete when we drop ISA between relationships (while still allowing all types of constraints on entities). If we further omit disjointness and covering over entities, reasoning becomes polynomial. Our lower complexity bound results are proved by direct reductions, while the upper bounds follow from the correspondences with expressive variants of the description logic *DL-Lite*, which we establish in this paper. These correspondences also show the usefulness of *DL-Lite* as a language for reasoning over conceptual models and ontologies.

1   Introduction

Conceptual modelling formalisms, such as the Entity-Relationship model [3], are used in the phase of conceptual database design, where the aim is to capture at best the semantics of the modelled application. This is achieved by expressing the constraints that hold on the entities, attributes and relationships, which represent the domain of interest, through suitable constructors provided by the conceptual modelling language. Thus, on the one hand it would be desirable to make such a language as expressive as possible in order to represent as many aspects of the modelled reality as possible. On the other hand, when using an expressive language, the designer faces the problem of understanding complex interactions that may occur between different parts of the conceptual model under construction and the constraints therein. Such interactions may force, e.g., some class (or even all classes) in the model to become inconsistent in the sense that there cannot exist a database state satisfying all constraints in which the class (respectively, all classes) is populated by at least one object. Or a class may turn out to be a subclass of another one, even though this is not explicitly asserted in the model. To understand the consequences, both explicit and implicit, of the constraints in the conceptual model being constructed, it is essential to provide automated reasoning

support, especially in those application scenarios where models may become very large and/or have complex interactions between constraints.

In this paper, we address these issues and investigate the computational complexity of reasoning in conceptual modelling languages equipped with various forms of constraints. Our analysis is carried out in the context of the Extended Entity-Relationship (EER) language [14], where the domain of interest is represented through *entities* (representing sets of objects), possibly equipped with *attributes*, and *relationships* (representing relations among objects). Note, however, that all of our results can also be adapted to other conceptual modelling formalisms, such as UML class diagrams[1]. Specifically, the kind of constraints that will be taken into account in this paper are the ones typically used in conceptual modelling, namely:

– *ISA* relations between both entities and relationships;
– *disjointness* and *covering* (referred to as the *Boolean* constructors in what follows) between both entities and relationships;
– *cardinality* constraints for participation of entities in relationships;
– *refinement* of cardinalities for sub-entities participating in relationships; and
– *multiplicity* constraints for attributes.

The hierarchy of EER languages considered in the paper is shown in the table below together with the complexity results for reasoning in these languages (all our languages include cardinality, refinement and multiplicity constraints).

lang.	entities			relationships			complexity
	ISA	disjoint	covering	ISA	disjoint	covering	
	$C_1 \sqsubseteq C_2$	$C_1 \sqcap C_2 \sqsubseteq \bot$	$C = C_1 \sqcup C_2$	$R_1 \sqsubseteq R_2$	$R_1 \sqcap R_2 \sqsubseteq \bot$	$R = R_1 \sqcup R_2$	
ER_{full}	+	+	+	+	+	+	ExpTime [5]
ER_{isaR}	+	+	+	+	—	—	ExpTime
ER_{bool}	+	+	+	—	—	—	NP
ER_{ref}	+	+	—	—	—	—	NLogSpace

In our investigation we exploit the tight correspondences between conceptual modelling formalisms, such as the ER model, and variants of Description Logics (DLs) [11]. DLs [2] are a family of logics studied in knowledge representation that are specifically tailored towards the representation of structured class-based information; quite often these logics enjoy nice computational properties.

It was shown [5] that reasoning with respect to UML class diagrams is an ExpTime-complete problem, and it is easy to see that this result carries over to ER_{full} diagrams as well (cf., e.g., [11]). The upper complexity bound result is established by encoding UML class diagrams in an expressive variant of DL, $\mathcal{DLR}_{ifd}$, reasoning in which is known to be in ExpTime (cf., [7]). The proof of the lower bound is by reduction of reasoning over knowledge bases in the DL $\mathcal{ALC}$ [2], which is an ExpTime-complete problem. The reduction proposed in [5] makes use of both ISA and the Boolean constructors between relationships. Here we strengthen this result by showing that even if we drop the Booleans between relationships from ER_{full} (obtaining the language denoted by ER_{isaR}) reasoning still stays ExpTime-complete.

---

[1] See, e.g., http://www.uml.org/.

2

We then prove that reasoning in the language ER_{bool}, which essentially corresponds to ER_{isaR} without ISA between relationships, can be done in NP, and is also NP-complete. Thus, quite surprisingly, ISA between relationships turns out to be a major source of complexity for reasoning over schemas, making it jump from NP to EXP-TIME. To prove the NP upper complexity bound we again exploit the correspondence with DLs: specifically, we resort to *DL-Lite$_{bool}$*, the *Boolean extension* of the tractable DL *DL-Lite* [8, 9], reasoning in which is an NP-complete problem [1]: we show that ER_{bool} schemas can be captured by knowledge bases in *DL-Lite$_{bool}$* so that the reasoning services carry over. The lower complexity bound is shown by a polynomial reduction of the satisfiability problem in propositional calculus.

Finally, we further restrict the language of ER_{bool} by dropping the covering constructor and obtaining the language called ER_{ref}. We prove that the reasoning problem for ER_{ref} is NLOGSPACE-complete. The NLOGSPACE membership is shown by reduction to reasoning in *DL-Lite$_{krom}$*, the Krom fragment of *DL-Lite$_{bool}$*, which is known to be NLOGSPACE-complete [1]. Hardness for NLOGSPACE follows from a reduction of the graph reachability problem to reasoning in ER_{ref}.

The correspondence between conceptual modelling languages like ER_{bool} and ER_{ref} and the *DL-Lite* family of DLs, developed and exploited in this paper, shows that both *DL-Lite$_{bool}$* and *DL-Lite$_{krom}$* are useful languages for reasoning over conceptual models and ontologies, even though they are not equipped with all the constructors that are typical of rich ontology languages such as OWL and its variants [4].

Our analysis is similar in spirit to [13], where the consistency checking problem for the EER model equipped with forms of inclusion and disjointness constraints is studied and a polynomial-time algorithm for the problem is given (assuming constant arities of relationships). Such a polynomial-time result is incomparable with the one for ER_{ref}, since ER_{ref} lacks both ISA and disjointness for relationships (both present in [13]); on the other hand, it is equipped with cardinality and multiplicity constraints. We also mention [16], where reasoning over cardinality constraints in the basic ER model is investigated and a polynomial-time algorithm for strong schema consistency is given, and [10], where the study is extended to the case when ISA between entities is also allowed and an exponential algorithm for entity consistency is provided. Note, however, that in [16, 10] the reasoning problem is analysed under the assumption that databases are finite, whereas we do not require finiteness in this paper.

The paper is organised as follows. In Section 2, we introduce some members of the *DL-Lite* family. Section 3 is devoted to the formal definition of the conceptual modelling language ER_{full} and the relevant reasoning problems. In Sections 4-6, we present the main results of the paper by establishing the computational complexity of reasoning over various fragments of ER_{full}: ER_{isaR}, ER_{bool} and ER_{ref}, respectively. Section 7 concludes the paper.

2 The *DL-Lite* Languages

We consider the extension *DL-Lite$_{bool}$* [1] of the description logic *DL-Lite* [8, 9]. The language of *DL-Lite$_{bool}$* contains *concept names* $A_0, A_1, \dots$ and *role names* $P_0, P_1, \dots$.

Complex *roles R* and *concepts C* of *DL-Lite_bool* are defined as follows:

$$
\begin{aligned}
R &::= \quad P_i \quad | \quad P_i^-, \\
B &::= \quad \bot \quad | \quad A_i \quad | \quad \geq q\, R, \\
C &::= \quad B \quad | \quad \neg C \quad | \quad C_1 \sqcap C_2,
\end{aligned}
$$

where $q \geq 1$. Concepts of the form B are called *basic concepts*. A *DL-Lite_bool knowledge base*, $\mathcal{K}$, is a finite set of axioms of the form $C_1 \sqsubseteq C_2$.

A *DL-Lite_bool interpretation* is a structure of the form

$$
\mathcal{I} \;=\; \big(\Delta, A_0^{\mathcal{I}}, A_1^{\mathcal{I}}, \ldots, P_0^{\mathcal{I}}, P_1^{\mathcal{I}}, \ldots\big), \tag{1}
$$

where Δ is a nonempty set, $A_i^{\mathcal{I}} \subseteq \Delta$ and $P_i^{\mathcal{I}} \subseteq \Delta \times \Delta$, for all i. The role and concept constructors are interpreted in $\mathcal{I}$ as usual:

$$
\begin{aligned}
(P_i^-)^{\mathcal{I}} &= \{(y,x) \in \Delta \times \Delta \mid (x,y) \in P_i^{\mathcal{I}}\}, & \text{(inverse role)} \\
\bot^{\mathcal{I}} &= \emptyset, & \text{(the empty set)} \\
(\geq q\, R)^{\mathcal{I}} &= \{x \in \Delta \mid \sharp\{y \in \Delta \mid (x,y) \in R^{\mathcal{I}}\} \geq q\}, & \text{('at least } q \text{ } R\text{-successors')} \\
(\neg C)^{\mathcal{I}} &= \Delta \setminus C^{\mathcal{I}}, & \text{('not in } C\text{')} \\
(C_1 \sqcap C_2)^{\mathcal{I}} &= C_1^{\mathcal{I}} \cap C_2^{\mathcal{I}}, & \text{('both in } C_1 \text{ and } C_2\text{')}
\end{aligned}
$$

where $\sharp X$ denotes the cardinality of the set X. The standard abbreviations $\top := \neg\bot$, $\exists R := (\geq 1\, R)$ and $\leq q\, R := \neg(\geq q+1\, R)$ we need are self-explanatory and correspond to the intended semantics. We say that an interpretation $\mathcal{I}$ *satisfies* an axiom $C_1 \sqsubseteq C_2$ if $C_1^{\mathcal{I}} \subseteq C_2^{\mathcal{I}}$. A knowledge base $\mathcal{K}$ is *satisfiable* if there is an interpretation $\mathcal{I}$ that satisfies all the members of $\mathcal{K}$ (such an interpretation $\mathcal{I}$ is called a *model* of $\mathcal{K}$). A concept C is *satisfiable w.r.t. a knowledge base* $\mathcal{K}$ if there is a model $\mathcal{I}$ of $\mathcal{K}$ such that $C^{\mathcal{I}} \neq \emptyset$.

We also consider a sublanguage of *DL-Lite_bool*, the *Krom fragment DL-Lite_krom*, where only axioms of the following form are allowed:

$$
B_1 \sqsubseteq B_2 \qquad \text{or} \qquad B_1 \sqsubseteq \neg B_2 \qquad \text{or} \qquad \neg B_1 \sqsubseteq B_2,
$$

where B_1, B_2 are basic concepts (i.e., are of the form $\bot$, A_i or $\geq q\, R$).

The following result is proved in [1] and will be used later on:

Theorem 1. *The concept and KB satisfiability problem is* NP-*complete for DL-Lite_bool KBs and* NLogSpace-*complete for DL-Lite_krom KBs.*

3   The Conceptual Modelling Language

In this section, we define the notion of a *conceptual schema* by providing syntax and semantics for the fully-fledged conceptual modelling language ER_{full} (the formalisation adopted here is based on previous presentations in [2, 3, 11]). First citizens of a conceptual schema are *entities*, *relationships* and *attributes*. Arguments of relationships—specifying the part played by an entity when participating in a particular relationship—are denoted by means of so-called *role names*. Given a conceptual schema, we make the

following assumptions about names: relationship and entity names are unique; attribute names are local to entities (i.e., the same attribute can be used by different entities but its type must be the same); role names are local to relationships (this freedom will be limited when considering conceptual models without sub-relationships).

3.1 Syntax

In what follows we make use of the notion of *labelled tuples*. Let X be a finite set $\{x_1, \ldots, x_n\}$ of *labels* and Y a finite set. An *X-labelled tuple over Y* is simply a (total) function $T \colon X \to Y$. For $x \in X$, we write $T[x]$ to refer to the element $y \in Y$ labelled by x. Given $y_1, \ldots, y_n \in Y$, the expression $\langle x_1 \colon y_1, \ldots, x_n \colon y_n \rangle$ stands for the X-labelled tuple T over Y such that $T[x_i] = y_i$, for $1 \leq 1 \leq n$. We also write $(x_i, y_i) \in T$ if $T[x_i] = y_i$. The set of all X-labelled tuples over Y is denoted by $T_Y(X)$.

Definition 1 (*ER_{full} Syntax*). An *ER_{full} conceptual schema Σ* is a tuple

$$(\mathcal{L}, \text{REL}, \text{ATT}, \text{CARD}_R, \text{CARD}_A, \text{REF}, \text{ISA}, \text{DISJ}, \text{COV}),$$

where

- $\mathcal{L}$ is the disjoint union of alphabets $\mathcal{E}$ for *entity* symbols, $\mathcal{A}$ for *attribute* symbols, $\mathcal{R}$ for *relationship* symbols, $\mathcal{U}$ for *role* symbols, and $\mathcal{D}$ for *domain* symbols. We will call the tuple $(\mathcal{E}, \mathcal{A}, \mathcal{R}, \mathcal{U}, \mathcal{D})$ the *signature* of the schema Σ.

- $\text{REL} \colon \mathcal{R} \to \bigcup_{\nu \subseteq \mathcal{U}, \nu \neq \emptyset} T_{\mathcal{E}}(\nu)$ is a (total) function that assigns to every relation symbol a tuple over the entity symbols labelled with a nonempty set of role symbols: $\text{REL}(R) = \langle U_1 \colon E_1, \ldots, U_m \colon E_m \rangle$, where m is the *arity* of R. Note that the roles U_i are pairwise distinct while the entities E_i can be repeated.

- $\text{ATT} \colon \mathcal{E} \to \bigcup_{\alpha \subseteq \mathcal{A}} T_{\mathcal{D}}(\alpha)$ is a (total) function that assigns to every entity symbol a tuple over the domain symbols labelled with some (possibly empty) set of attribute symbols: $\text{ATT}(E) = \langle A_1 \colon D_1, \ldots, A_h \colon D_h \rangle$.

- $\text{CARD}_R \colon \mathcal{R} \times \mathcal{U} \times \mathcal{E} \to \mathbb{N} \times (\mathbb{N} \cup \{\infty\})$ is a *partial* function defining *cardinality constraints*. The value of $\text{CARD}_R(R, U, E)$ may be defined only if $(U, E) \in \text{REL}(R)$.

- $\text{CARD}_A \colon \mathcal{A} \times \mathcal{E} \to \mathbb{N} \times (\mathbb{N} \cup \{\infty\})$ is a *partial* function defining *multiplicity for attributes*. The value of $\text{CARD}_A(A, E)$ may be defined only if $(A, D) \in \text{ATT}(E)$, for some $D \in \mathcal{D}$.

- $\text{REF} \colon \mathcal{R} \times \mathcal{U} \times \mathcal{E} \to \mathbb{N} \times (\mathbb{N} \cup \{\infty\})$ is a *partial* function defining *refinement* of cardinality constraints for sub-entities (see ISA below). The value of $\text{REF}(R, U, E)$ may be defined only if E ISA E' and $(U, E') \in \text{REL}(R)$. Note that REF subsumes classical cardinality constraints (CARD_R).

- ISA is the union of two binary relations ISA_E and ISA_R, where $\text{ISA}_R \subseteq \mathcal{E} \times \mathcal{E}$ and $\text{ISA}_R \subseteq \mathcal{R} \times \mathcal{R}$. These two binary relations define the ISA hierarchy on entities and relationships, respectively.

- DISJ is the union of two binary relations DISJ_E and DISJ_R, where $\text{DISJ}_E \subseteq 2^{\mathcal{E}} \times \mathcal{E}$ and $\text{DISJ}_R \subseteq 2^{\mathcal{R}} \times \mathcal{R}$. The intended meaning of, say, $(\{E_1, \ldots, E_n\}, E) \in \text{DISJ}_E$ is '$E_1, \ldots, E_n$ are disjoint sub-entities of E.'

- COV is the union of two binary relations COV_E and COV_R, where $\text{COV}_E \subseteq 2^{\mathcal{E}} \times \mathcal{E}$ and $\text{COV}_R \subseteq 2^{\mathcal{R}} \times \mathcal{R}$. The intended meaning of, say, $(\{E_1, \ldots, E_n\}, E) \in \text{COV}_E$ is '$E_1, \ldots, E_n$ are covering sub-entities of E.'

We additionally require that the relations ISA_R, DISJ_R and COV_R may only be defined for relationships of the same arity.

In what follows we use E_1 ISA E_2 as a shortcut for $(E_1, E_2) \in \text{ISA}$ (similarly for ISA_E and ISA_R) and $\{E_1, \ldots, E_n\}$ DISJ E as a shortcut for $(\{E_1, \ldots, E_n\}, E) \in \text{DISJ}$ (similarly for DISJ_E, DISJ_R, COV, COV_E and COV_R).

3.2 Semantics

The following definition specifies the set-theoretic semantics of ER_{full} schemas.

Definition 2 (ER_{full} **Semantics**). Let Σ be an ER_{full} conceptual schema and B_D, for $D \in \mathcal{D}$, a collection of disjoint countable sets called *basic domains*. An *interpretation* for Σ is a pair $\mathcal{B} = (\Delta^{\mathcal{B}} \cup \Lambda^{\mathcal{B}}, \cdot^{\mathcal{B}})$, where

- $\Delta^{\mathcal{B}}$ is a nonempty set, the *interpretation domain*;
- $\Lambda^{\mathcal{B}} = \bigcup_{D \in \mathcal{D}} \Lambda_D^{\mathcal{B}}$, with $\Lambda_D^{\mathcal{B}} \subseteq B_D$ for each $D \in \mathcal{D}$, is the *active domain* such that $\Delta^{\mathcal{B}} \cap \Lambda^{\mathcal{B}} = \emptyset$;
- $\cdot^{\mathcal{B}}$ is a function such that
    (i) $D^{\mathcal{B}} = \Lambda_D^{\mathcal{B}}$, for each $D \in \mathcal{D}$;
    (ii) $E^{\mathcal{B}} \subseteq \Delta^{\mathcal{B}}$, for each $E \in \mathcal{E}$;
    (iii) $R^{\mathcal{B}} \subseteq T_{\Delta^{\mathcal{B}}}(\nu)$, where $\nu = \{U_i \in \mathcal{U} \mid (U_i, E_i) \in \text{REL}(R)\}$, for each $R \in \mathcal{R}$;
    (iv) $A^{\mathcal{B}} \subseteq \Delta^{\mathcal{B}} \times \Lambda^{\mathcal{B}}$, for each $A \in \mathcal{A}$.

An interpretation $\mathcal{B}$ of a schema Σ is called a *legal database state* if it satisfies the following conditions:

1. For each $R \in \mathcal{R}$ with $\text{REL}(R) = \langle U_1 \colon E_1, \ldots, U_m \colon E_m \rangle$ and each $r \in R^{\mathcal{B}}$, we have $r = \langle U_1 \colon e_1, \ldots, U_m \colon e_m \rangle$ with $e_i \in E_i^{\mathcal{B}}$, for each $1 \leq i \leq m$. In the following, we adopt the convention to denote such a labelled tuple r as $(e_1, \ldots, e_m)$, and we may use $r[i]$ instead of $r[U_i]$ to denote the U_i/i-component of r—i.e., we simplify the notation by adopting for tuples a *positional* notation instead of the one based on role names.

2. For each $E \in \mathcal{E}$ with $\text{ATT}(E) = \langle A_1 \colon D_1, \ldots, A_h \colon D_h \rangle$, each $(e, a) \in \Delta^{\mathcal{B}} \times \Lambda^{\mathcal{B}}$ and each $1 \leq i \leq h$, if $(e, a) \in A_i^{\mathcal{B}}$ then $a \in D_i^{\mathcal{B}}$.

3. For each $R \in \mathcal{R}$ with $\text{REL}(R) = \langle U_1 \colon E_1, \ldots, U_m \colon E_m \rangle$ and each $1 \leq i \leq m$, if $\text{CARD}_R(R, U_i, E_i) = (\alpha, \beta)$ then, for all $e \in E_i^{\mathcal{B}}$,

$$\alpha \quad \leq \quad \sharp\{(e_1, \ldots, e_i, \ldots, e_m) \in R^{\mathcal{B}} \mid e_i = e\} \quad \leq \quad \beta. \qquad (2)$$

4. For each $E \in \mathcal{E}$ with $\text{ATT}(E) = \langle A_1 : D_1, \ldots, A_h : D_h \rangle$ and each $1 \leq i \leq h$, if $\text{CARD}_A(A_i, E) = (\alpha, \beta)$ then $\alpha \leq \sharp\{(e, a) \in A_i^{\mathcal{B}}\} \leq \beta$, for all $e \in E^{\mathcal{B}}$.

5. For each $R \in \mathcal{R}$ with $\text{REL}(R) = \langle U_1 : E_1, \ldots, U_m : E_m \rangle$, each $1 \leq i \leq m$ and each $E \in \mathcal{E}$ with $E \text{ISA} E_i$, if $\text{REF}(R, U_i, E) = (\alpha, \beta)$ then (2) holds for all $e \in E^{\mathcal{B}}$.

6. For all $E_1, E_2 \in \mathcal{E}$, if $E_1 \text{ ISA}_E E_2$ then $E_1^{\mathcal{B}} \subseteq E_2^{\mathcal{B}}$ (similarly for relationships).

7. For all $E, E_1, \ldots, E_n \in \mathcal{E}$, if $\{E_1, \ldots, E_n\} \text{ DISJ}_E E$ then $E_i^{\mathcal{B}} \subseteq E^{\mathcal{B}}$, for every $1 \leq i \leq n$, and $E_i^{\mathcal{B}} \cap E_j^{\mathcal{B}} = \emptyset$, for all $1 \leq i < j \leq n$ (similarly for relationships).

8. For all $E, E_1, \ldots, E_n \in \mathcal{E}$, $\{E_1, \ldots, E_n\} \text{ COV}_E E$ implies $E^{\mathcal{B}} = \bigcup_{i=1}^{n} E_i^{\mathcal{B}}$ (similarly for relationships).

3.3 Reasoning Problems

Reasoning tasks over conceptual schemas include verifying whether an entity, a relationship, or a schema is *consistent*, or checking whether an entity (relationship) *subsumes* another entity (relationship, respectively). The model-theoretic semantics associated with a conceptual schema allows us to define formally the following reasoning tasks:

Definition 3 (Reasoning services). Let Σ be an ER_{full} schema.

Schema consistency. Σ is *consistent* if there exists a legal database state $\mathcal{B}$ for Σ such that $E^{\mathcal{B}} \neq \emptyset$, for some entity $E \in \mathcal{E}$.
Strong (schema) consistency. Σ is *strongly consistent* if there exists a legal database state $\mathcal{B}$ for Σ such that $E^{\mathcal{B}} \neq \emptyset$, for *every* entity $E \in \mathcal{E}$.
Entity consistency. An entity $E \in \mathcal{E}$ is *consistent w.r.t. a schema Σ* if there exists a legal database state $\mathcal{B}$ for Σ such that $E^{\mathcal{B}} \neq \emptyset$.
Relationship consistency. A relationship $R \in \mathcal{R}$ is *consistent w.r.t. a schema Σ* if there exists a legal database state $\mathcal{B}$ for Σ such that $R^{\mathcal{B}} \neq \emptyset$.
Entity subsumption. An entity $E_1 \in \mathcal{E}$ *subsumes* an entity $E_2 \in \mathcal{E}$ w.r.t. a schema Σ if $E_2^{\mathcal{B}} \subseteq E_1^{\mathcal{B}}$, for every legal database state $\mathcal{B}$ for Σ.
Relationships subsumption. A relationship $R_1 \in \mathcal{R}$ *subsumes* a relationship $R_2 \in \mathcal{R}$ w.r.t. a schema Σ if $R_2^{\mathcal{B}} \subseteq R_1^{\mathcal{B}}$, for every legal database state $\mathcal{B}$ for Σ.

The reasoning tasks of Schema/Entity/Relationship consistency and Entity subsumption are reducible to each other. Indeed, that Entity subsumption is equivalent to Entity satisfiability is shown in [5]. Schema consistency can be reduced to Entity consistency by extending Σ as follows: let O^* be a fresh entity symbol, $\mathcal{E}^* = \mathcal{E} \cup \{O^*\}$ and $\text{COV}^* = \text{COV} \cup \{(\mathcal{E}, O^*)\}$. Clearly, Σ is consistent iff O^* is consistent w.r.t. Σ^*. For the converse reduction Σ is extended as follows: let O^* be a fresh entity symbol and R_E a fresh relationship symbol, $\mathcal{E}^* = \mathcal{E} \cup \{O^*\}$, $\text{COV}^* = \text{COV} \cup \{(\mathcal{E}, O^*)\}$, $\mathcal{R}^* = \mathcal{R} \cup \{R_E\}$, $\text{REL}(R_E) = \langle U_1 : E, U_2 : O^* \rangle$, $\text{CARD}_R(R_E, U_2, O^*) = (1, \infty)$. Clearly, E is consistent w.r.t. Σ iff Σ^* is consistent.

Relationship consistency can be reduced to Entity consistency by extending Σ as follows: let O^* be a fresh entity symbol, $\mathcal{E}^* = \mathcal{E} \cup \{O^*\}$, $\text{ISA}_E^* = \text{ISA}_E \cup \{(O^*, E)\}$ and REF^* extends REF so that $\text{REF}^*(R, U, O^*) = (1, \beta)$, where E is an entity with

$(U, E) \in \text{REL}(R)$ and β is such that $\text{CARD}_R(R, U, E) = (\alpha, \beta)$. Relationship R is consistent w.r.t. Σ iff entity O^* is consistent w.r.t. Σ^*. For the converse reduction, let R_E be a fresh relationship symbol with $\text{REL}(R_E) = \langle U_1 \colon E, U_2 \colon E \rangle$. Then E is consistent iff R_E is consistent.

Finally, we note that, in absence of the covering constructor, Schema consistency cannot be reduced to a single instance of Entity consistency, though it can be reduced to several Entity consistency checks.

4 Reasoning over ER_{isaR} Schemas

The modelling language ER_{isaR} is the subset of ER_{full} without the Booleans between relationships (i.e., $\text{DISJ}_R = \emptyset$ and $\text{COV}_R = \emptyset$) but with the possibility to express ISA between them. In this section we show that reasoning in ER_{isaR} is an EXPTIME-complete problem. The upper bound follows from [5]. The lower bound is established by reducing concept satisfiability w.r.t. $\mathcal{ALC}$ knowledge bases, which is known to be EXPTIME-complete [2], to entity consistency w.r.t. ER_{isaR} conceptual schemas.

We remind the reader that $\mathcal{ALC}$ *concepts* C are defined as follows:

$$ C \quad ::= \quad A_i \quad | \quad \neg C \quad | \quad C_1 \sqcap C_2 \quad | \quad C_1 \sqcup C_2 \quad | \quad \exists P_i.C \quad | \quad \forall P_i.C, $$

where the last two constructors are interpreted in $\mathcal{I}$ of the form (1) by taking

$$ (\exists P_i.C)^{\mathcal{I}} = \{x \in \Delta \mid \exists y \in \Delta \, ((x, y) \in P_i^{\mathcal{I}} \wedge y \in C^{\mathcal{I}})\}, $$
$$ (\forall P_i.C)^{\mathcal{I}} = \{x \in \Delta \mid \forall y \in \Delta \, ((x, y) \in P_i^{\mathcal{I}} \rightarrow y \in C^{\mathcal{I}})\}. $$

An $\mathcal{ALC}$ *knowledge base* is a finite set of $\mathcal{ALC}$ concept inclusions $C_1 \sqsubseteq C_2$. It is easy to show (see, e.g., [5, Lemma 5.1]) that one can convert, in a satisfiability preserving way, an $\mathcal{ALC}$ KB $\mathcal{K}$ into a *primitive* KB $\mathcal{K}'$ that contains only axioms of the form:

$$ A \sqsubseteq B, \quad A \sqsubseteq \neg B, \quad A \sqsubseteq B \sqcup B', \quad A \sqsubseteq \forall R.B, \quad A \sqsubseteq \exists R.B, $$

where A, B, B' are concept names and R is a role name, and the size of $\mathcal{K}'$ is linear in the size of $\mathcal{K}$. Thus, concept satisfiability w.r.t. primitive $\mathcal{ALC}$ KBs is EXPTIME-complete [5].

Let $\mathcal{K}$ be a primitive $\mathcal{ALC}$ KB. We illustrate a satisfiable preserving mapping from $\mathcal{K}$ into an ER_{isaR} schema $\Sigma(\mathcal{K})$: the first three forms of axioms are dealt with in a way similar to [5]. Axioms of the form $A \sqsubseteq \forall R.B$ are encoded in [5] using disjointness and covering (along with ISA) between relationships, which are unavailable in ER_{isaR}. In order to stay within ER_{isaR}, we propose to use reification of $\mathcal{ALC}$ roles (which are binary relationships) to encode the last two forms of axioms. This approach is illustrated in Fig. 1: in (a), $A \sqsubseteq \forall R.B$ is encoded by reifying the binary relationship R with the entity C_R so that the functional relationships R_1 and R_2 give the first and second component of the reified R, respectively; a similar encoding is used to capture $A \sqsubseteq \exists R.B$ in (b). The following lemma shows that ISA between relationships—and so conceptual schemas in ER_{isaR}—are enough to encode $\mathcal{ALC}$ axioms.
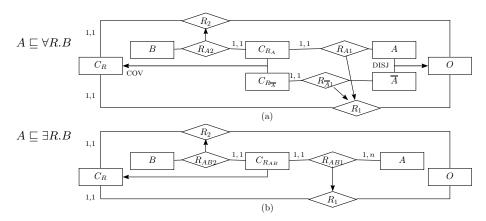
Fig. 1. Encoding axioms: (a) $A \sqsubseteq \forall R.B$; (b) $A \sqsubseteq \exists R.B$.

Lemma 1. *A concept name E is satisfiable w.r.t. a primitive $\mathcal{ALC}$ KB $\mathcal{K}$ iff the entity E is consistent w.r.t. the ER_{isaR} schema $\Sigma(\mathcal{K})$.*

Proof. ($\Leftarrow$) Let $\mathcal{B} = (\Delta^{\mathcal{B}}, \cdot^{\mathcal{B}})$ be a legal database for $\Sigma(\mathcal{K})$ such that $E^{\mathcal{B}} \neq \emptyset$. We construct a model $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ of $\mathcal{K}$ with $E^{\mathcal{I}} \neq \emptyset$ by taking $\Delta^{\mathcal{I}} = \Delta^{\mathcal{B}}$, $A^{\mathcal{I}} = A^{\mathcal{B}}$, for all concept names A in $\mathcal{K}$, and $R^{\mathcal{I}} = (R_1^- \circ R_2)^{\mathcal{B}}$, for all role names R in $\mathcal{K}$, where $\circ$ denotes the binary relation composition. Clearly, $E^{\mathcal{I}} \neq \emptyset$. Let us show that $\mathcal{I}$ is indeed a model of $\mathcal{K}$. The cases of axioms of the form $A \sqsubseteq B$, $A \sqsubseteq \neg B$ and $A \sqsubseteq B \sqcup B'$ are treated as in [5]. Let us consider the remaining two cases.

Case $A \sqsubseteq \forall R.B$. Let $o \in A^{\mathcal{I}}$ and $o' \in \Delta^{\mathcal{I}}$ with $(o, o') \in R^{\mathcal{I}}$. We show that $o \in (\forall R.B)^{\mathcal{I}}$. Since $R^{\mathcal{I}} = (R_1^- \circ R_2)^{\mathcal{B}}$, there is $o'' \in \Delta^{\mathcal{B}}$ with $(o, o'') \in (R_1^-)^{\mathcal{B}}$ and $(o'', o') \in R_2^{\mathcal{B}}$. Then $o'' \in C_R^{\mathcal{B}}$ and, by the covering constraint, $o'' \in C_{R_A}^{\mathcal{B}} \cup C_{R_{\overline{A}}}^{\mathcal{B}}$. We claim that $o'' \in C_{R_A}^{\mathcal{B}}$. Indeed, suppose otherwise; then $o'' \in C_{R_{\overline{A}}}^{\mathcal{B}}$, and so there is a unique $a \in \Delta^{\mathcal{B}}$ such that $(o'', a) \in R_{\overline{A}1}^{\mathcal{B}}$ and $a \in \overline{A}^{\mathcal{B}}$; it follows from $R_{\overline{A}1}^{\mathcal{B}} \subseteq R_1^{\mathcal{B}}$ and the cardinality constraint on C_R that $a = o$, contrary to $o \in A^{\mathcal{B}} = A^{\mathcal{I}}$ and the disjointness of A and $\overline{A}$. Since $o'' \in C_{R_A}^{\mathcal{B}}$, there is a unique $b \in \Delta^{\mathcal{B}}$ with $(o'', b) \in R_{A2}^{\mathcal{B}}$ and $b \in B^{\mathcal{B}}$. From $R_{A2}^{\mathcal{B}} \subseteq R_2^{\mathcal{B}}$ and the cardinality constraint on C_R, we conclude that $b = o'$. Thus, $o' \in B^{\mathcal{B}} = B^{\mathcal{I}}$ and $o \in (\forall R.B)^{\mathcal{I}}$.

Case $A \sqsubseteq \exists R.B$. Let $o \in A^{\mathcal{I}}$. Since $o \in A^{\mathcal{I}} = A^{\mathcal{B}}$, there is $o' \in \Delta^{\mathcal{B}}$ with $(o, o') \in (R_{AB1}^-)^{\mathcal{B}}$ and $o' \in C_{R_{AB}}^{\mathcal{B}}$. As $R_{AB1}^{\mathcal{B}} \subseteq R_1^{\mathcal{B}}$, we have $(o, o') \in (R_1^-)^{\mathcal{B}}$, and, as $o' \in C_{R_{AB}}^{\mathcal{B}}$, there is $o'' \in \Delta^{\mathcal{B}}$ such that $(o', o'') \in R_{AB2}^{\mathcal{B}} \subseteq R_2^{\mathcal{B}}$ and $o'' \in B^{\mathcal{B}} = B^{\mathcal{I}}$. Thus, as $R^{\mathcal{I}} = (R_1^- \circ R_2)^{\mathcal{B}}$, we obtain $(o, o'') \in R^{\mathcal{I}}$ and $o'' \in B^{\mathcal{I}}$, i.e. $o \in (\exists R.B)^{\mathcal{I}}$.

($\Rightarrow$) Let $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ be an $\mathcal{ALC}$ model of $\mathcal{K}$ such that $E^{\mathcal{I}} \neq \emptyset$. We construct a legal database state $\mathcal{B} = (\Delta^{\mathcal{B}}, \cdot^{\mathcal{B}})$ for $\Sigma(\mathcal{K})$ such that $E^{\mathcal{B}} \neq \emptyset$. Let $\Delta^{\mathcal{B}} = \Delta^{\mathcal{I}} \cup \Gamma$, where Γ is the disjoint union of the $\Delta_R = \{(o, o') \in \Delta^{\mathcal{I}} \mid (o, o') \in R^{\mathcal{I}}\}$, for all $\mathcal{ALC}$ role names R. We set $A^{\mathcal{B}} = A^{\mathcal{I}}$ and $\overline{A}^{\mathcal{B}} = (\neg A)^{\mathcal{I}}$, for all concept names A, $O^{\mathcal{B}} = \Delta^{\mathcal{I}}$, for the entity O, and $C_R^{\mathcal{B}} = \Delta_R$, for all $\mathcal{ALC}$ role names R.

Next, for every $\mathcal{ALC}$ axiom of the form $A \sqsubseteq \forall R.B$, we set

9

- $C_{R_A}^{\mathcal{B}} = \{(o, o') \in \Delta_R \mid o \in A^{\mathcal{I}}\}$, $C_{R_{\overline{A}}}^{\mathcal{B}} = \{(o, o') \in \Delta_R \mid o \in (\neg A)^{\mathcal{I}}\}$,
- $R_1^{\mathcal{B}} = \{((o, o'), o) \in \Delta_R \times \Delta^{\mathcal{I}} \mid (o, o') \in R^{\mathcal{I}}\}$,
- $R_2^{\mathcal{B}} = \{((o, o'), o') \in \Delta_R \times \Delta^{\mathcal{I}} \mid (o, o') \in R^{\mathcal{I}}\}$,
- $R_{A1}^{\mathcal{B}} = \{((o, o'), o) \in R_1^{\mathcal{B}} \mid o \in A^{\mathcal{I}}\}$, $R_{\overline{A}1}^{\mathcal{B}} = \{((o, o'), o) \in R_1^{\mathcal{B}} \mid o \in (\neg A)^{\mathcal{I}}\}$,
- $R_{A2}^{\mathcal{B}} = \{((o, o'), o') \in R_2^{\mathcal{B}} \mid o \in A^{\mathcal{I}}\}$,

and, for every $\mathcal{ALC}$ axiom of the form $A \sqsubseteq \exists R.B$, we set

- $C_{R_{AB}}^{\mathcal{B}} = \{(o, o') \in \Delta_R \mid o \in A^{\mathcal{I}}$ and $o' \in B^{\mathcal{I}}\}$,
- $R_1^{\mathcal{B}} = \{((o, o'), o) \in \Delta_R \times \Delta^{\mathcal{I}} \mid (o, o') \in R^{\mathcal{I}}\}$,
- $R_2^{\mathcal{B}} = \{((o, o'), o') \in \Delta_R \times \Delta^{\mathcal{I}} \mid (o, o') \in R^{\mathcal{I}}\}$,
- $R_{AB1}^{\mathcal{B}} = \{((o, o'), o) \in R_1^{\mathcal{B}} \mid (o, o') \in C_{R_{AB}}^{\mathcal{B}}\}$.
- $R_{AB2}^{\mathcal{B}} = \{((o, o'), o') \in R_2^{\mathcal{B}} \mid (o, o') \in C_{R_{AB}}^{\mathcal{B}}\}$.

It is now easy to show that $\mathcal{B}$ is a legal database state for $\Sigma(\mathcal{K})$ and $E^{\mathcal{B}} \neq \emptyset$.

Since reasoning over $\mathcal{ALC}$ knowledge bases is an EXPTIME-complete problem [2] and ER_{isaR} is a sub-language of ER_{full}, which is also EXPTIME-complete [5], we obtain the following result:

Theorem 2. *Reasoning over ER_{isaR} schemas is* EXPTIME-*complete.*

5 Reasoning over ER_{bool} Schemas

Denote by ER_{bool} the sublanguage of ER_{full} without ISA and the Booleans between relationships (i.e., $\text{ISA}_R = \emptyset$, $\text{DISJ}_R = \emptyset$ and $\text{COV}_R = \emptyset$). We also impose an extra restriction on REL: reusing the same role symbol by different relations is not allowed. More precisely, there is no $U \in \mathcal{U}$ such that $(U, E') \in \text{REL}(R')$ and $(U, E'') \in \text{REL}(R'')$, for some *distinct* $R', R'' \in \mathcal{R}$ and some $E', E'' \in \mathcal{E}$. This restriction does make sense in ER_{bool}, since the language does not allow for sub-relationships.

We first define a polynomial translation τ of ER_{bool} schemas into *DL-Lite$_{bool}$* knowledge bases. Then we show that an entity E is consistent w.r.t. an ER_{bool} schema Σ iff the translation of the entity, $\overline{E}$, is satisfiable w.r.t. the knowledge base $\tau(\Sigma)$. The latter problem is known to be in NP (Theorem 1).

Let Σ be an ER_{bool} schema. Given an entity, domain or relationship symbol N from $\mathcal{E} \cup \mathcal{D} \cup \mathcal{R}$, let $\overline{N}$ be a *DL-Lite$_{bool}$* concept name. Similarly, for an attribute or role symbol $N \in \mathcal{A} \cup \mathcal{U}$, let $\overline{N}$ be a *DL-Lite$_{bool}$* role name. The translation $\tau(\Sigma)$ is defined as the following set of *DL-Lite$_{bool}$* concept inclusions:

$$\tau(\Sigma) \;=\; \tau_{dom} \;\cup\; \bigcup_{R \in \mathcal{R}} \left[\tau_{rel}^R \cup \tau_{card_R}^R \cup \tau_{ref}^R \right] \;\cup\; \bigcup_{E \in \mathcal{E}} \left[\tau_{att}^E \cup \tau_{card_A}^E \right] \;\cup$$

$$\bigcup_{\substack{E_1, E_2 \in \mathcal{E} \\ E_1 \text{ISA} E_2}} \tau_{isa}^{E_1, E_2} \;\cup\; \bigcup_{\substack{E_1,...,E_n, E \in \mathcal{E} \\ \{E_1,...,E_n\} \text{DISJ} E}} \tau_{disj}^{\{E_1,...,E_n\}, E} \;\cup\; \bigcup_{\substack{E_1,...,E_n, E \in \mathcal{E} \\ \{E_1,...,E_n\} \text{COV} E}} \tau_{cov}^{\{E_1,...,E_n\}, E},$$

where

10

- $\tau_{dom} = \big\{ \overline{D} \sqsubseteq \neg \overline{X} \mid D \in \mathcal{D},\ X \in \mathcal{E} \cup \mathcal{R} \cup \mathcal{D},\ D \neq X \big\}$;

- $\tau_{rel}^{R} = \big\{ \overline{R} \sqsubseteq \exists \overline{U},\ \geq 2\,\overline{U} \sqsubseteq \bot,\ \exists \overline{U} \sqsubseteq \overline{R},\ \exists \overline{U}^{-} \sqsubseteq \overline{E} \mid (U, E) \in \mathrm{REL}(R) \big\}$;

- $\tau_{card_R}^{R} = \big\{ \overline{E} \sqsubseteq\ \geq \alpha\,\overline{U}^{-} \mid (U, E) \in \mathrm{REL}(R), \mathrm{CARD}_R(R, U, E) = (\alpha, \beta), \alpha \neq 0 \big\}$
  $\cup \big\{ \overline{E} \sqsubseteq\ \leq \beta\,\overline{U}^{-} \mid (U, E) \in \mathrm{REL}(R), \mathrm{CARD}_R(R, U, E) = (\alpha, \beta), \beta \neq \infty \big\}$;

- $\tau_{ref}^{R} = \big\{ \overline{E} \sqsubseteq\ \geq \alpha\,\overline{U}^{-} \mid (U, E) \in \mathrm{REL}(R), \mathrm{REF}(R, U, E) = (\alpha, \beta), \alpha \neq 0 \big\}$
  $\cup \big\{ \overline{E} \sqsubseteq\ \leq \beta\,\overline{U}^{-} \mid (U, E) \in \mathrm{REL}(R), \mathrm{REF}(R, U, E) = (\alpha, \beta), \beta \neq \infty \big\}$;

- $\tau_{att}^{E} = \big\{ \exists \overline{A}^{-} \sqsubseteq \overline{D} \mid (A, D) \in \mathrm{ATT}(E) \big\}$;

- $\tau_{card_A}^{E} = \big\{ \overline{E} \sqsubseteq\ \geq \alpha\,\overline{A} \mid (A, D) \in \mathrm{ATT}(E), \mathrm{CARD}_A(A, E) = (\alpha, \beta), \alpha \neq 0 \big\}$
  $\cup \big\{ \overline{E} \sqsubseteq\ \leq \beta\,\overline{A} \mid (A, D) \in \mathrm{ATT}(E), \mathrm{CARD}_A(A, E) = (\alpha, \beta), \beta \neq \infty \big\}$;

- $\tau_{isa}^{E_1, E_2} = \big\{ \overline{E_1} \sqsubseteq \overline{E_2} \big\}$;

- $\tau_{disj}^{\{E_1, \ldots, E_n\}, E} = \big\{ \overline{E_i} \sqsubseteq \overline{E} \mid 1 \leq i \leq n \big\} \cup \big\{ \overline{E_i} \sqsubseteq \neg \overline{E_j} \mid 1 \leq i < j \leq n \big\}$;

- $\tau_{cov}^{\{E_1, \ldots, E_n\}, E} = \big\{ \overline{E_i} \sqsubseteq \overline{E} \mid 1 \leq i \leq n \big\} \cup \big\{ \overline{E} \sqsubseteq \overline{E_1} \sqcup \cdots \sqcup \overline{E_n} \big\}$.

Clearly, the size of $\tau(\Sigma)$ is polynomial in the size of Σ (under the same coding of the numerical parameters).

Lemma 2. *An entity E is consistent w.r.t. an ER_{bool} schema Σ iff the concept $\overline{E}$ is satisfiable w.r.t. the* DL-Lite$_{bool}$ *KB $\tau(\Sigma)$.*

Proof. ($\Rightarrow$) Let $\mathcal{B} = (\Delta^{\mathcal{B}} \cup \Lambda^{\mathcal{B}}, \cdot^{\mathcal{B}})$ be a legal database state for Σ such that $E^{\mathcal{B}} \neq \emptyset$, where $\{B_D\}_{D \in \mathcal{D}}$ are the domain sets. Define a model $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ of $\tau(\Sigma)$ by taking $\Delta^{\mathcal{I}} = \Delta^{\mathcal{B}} \cup \Lambda^{\mathcal{B}} \cup \Gamma$, where Γ is the disjoint union of the $\Delta_R = \{(e_1, \ldots, e_m) \in R^{\mathcal{B}}\}$, for all relationships $R \in \mathcal{R}$, and setting $\overline{D}^{\mathcal{I}} = D^{\mathcal{B}}$, for every $D \in \mathcal{D}$, $\overline{E}^{\mathcal{I}} = E^{\mathcal{B}}$, for every $E \in \mathcal{E}$, $\overline{A}^{\mathcal{I}} = A^{\mathcal{B}}$, for every $A \in \mathcal{A}$, $\overline{R}^{\mathcal{I}} = \Delta_R$, for every $R \in \mathcal{R}$, and, for every $U \in \mathcal{U}$ such that there is $R \in \mathcal{R}$ with $\mathrm{REL}(R) = \langle U_1 \colon E_1, \ldots, U_m \colon E_m \rangle$ and $U = U_i$ for some i with $1 \leq i \leq m$,

$$\overline{U}^{\mathcal{I}} = \{((e_1, \ldots, e_m), e_i) \in \Delta_R \times \Delta^{\mathcal{B}} \mid (e_1, \ldots, e_m) \in R^{\mathcal{B}}\}. \tag{3}$$

Clearly, $\overline{E}^{\mathcal{I}} \neq \emptyset$. We now prove that $\mathcal{I}$ is indeed a model of $\tau(\Sigma)$. We guide the proof by considering the translation of the various statements in Σ.

1. We show $\mathcal{I} \models \tau_{dom}$. For any two distinct $D_1, D_2 \in \mathcal{D}$, we have $D_1^{\mathcal{B}} \cap D_2^{\mathcal{B}} = \emptyset$, and so $\mathcal{I} \models \overline{D_1} \sqsubseteq \neg \overline{D_2}$. For all $D \in \mathcal{D}$ and $E \in \mathcal{E}$, since $E^{\mathcal{B}} \subseteq \Delta^{\mathcal{B}}$, $D^{\mathcal{B}} \subseteq \Lambda^{\mathcal{B}}$ and $\Delta^{\mathcal{B}} \cap \Lambda^{\mathcal{B}} = \emptyset$, we have $\mathcal{I} \models \overline{D} \sqsubseteq \neg \overline{E}$. Next, for all $D \in \mathcal{D}$ and $R \in \mathcal{R}$, as $D^{\mathcal{B}} \subseteq \Lambda^{\mathcal{B}}$, $\overline{R}^{\mathcal{I}} = \Delta_R \subseteq \Gamma$ and $\Gamma \cap \Lambda^{\mathcal{B}} = \emptyset$, we have $\mathcal{I} \models \overline{D} \sqsubseteq \neg \overline{R}$.

2. $\mathrm{REL}(R) = \langle U_1 : E_1, \ldots, U_m : E_m \rangle$. Consider all axioms in $\tau_{rel}^{R} \cup \tau_{card_R}^{R} \cup \tau_{ref}^{R}$:

   (a) $\overline{R} \sqsubseteq \exists \overline{U_i}$. Let $r \in \overline{R}^{\mathcal{I}}$. Then r is of the form $(e_1, \ldots, e_m) \in R^{\mathcal{B}}$. By (3), $(r, e_i) \in \overline{U_i}^{\mathcal{I}}$, and so $r \in \exists \overline{U_i}^{\mathcal{I}}$.

(b) $\geq 2\,\overline{U_i} \sqsubseteq \bot$. Suppose that there are $(r, e), (r, e') \in \overline{U_i}^{\mathcal{I}}$ such that $e \neq e'$. By (3), r is of the form $(e_1, \ldots, e_m)$ and $e = e_i = e'$, contrary to $e \neq e'$.

(c) $\exists \overline{U_i}^{\,-} \sqsubseteq \overline{E_i}$. Let $e \in (\exists \overline{U_i}^{\,-})^{\mathcal{I}}$. Then $(r, e) \in \overline{U_i}^{\mathcal{I}}$ for some $r \in \Delta^{\mathcal{I}}$. Since U_i may be involved only in one relation (R in this case) and in view of (3), r is of the form $(e_1, \ldots, e_m) \in R^{\mathcal{B}}$ and $e_i = e$. By the semantics of R, $e \in E_i^{\mathcal{B}}$, from which $e \in \overline{E_i}^{\mathcal{I}}$.

(d) $\exists \overline{U_i} \sqsubseteq \overline{R}$. Let $r \in (\exists \overline{U_i})^{\mathcal{I}}$. Then $(r, e) \in \overline{U_i}^{\mathcal{I}}$ for some $e \in \Delta^{\mathcal{I}}$. Since U_i may be involved only in one relation (R in this case) and by (3), r is of the form $(e_1, \ldots, e_m) \in R^{\mathcal{B}}$ and $e = e_i$. Therefore, $r \in \overline{R}^{\mathcal{I}}$.

(e) $\overline{E} \sqsubseteq\, \geq \alpha\,\overline{U_i}^{\,-}$ (when $\mathrm{CARD}_R(R, U_i, E_i) = (\alpha, \beta)$ and $\alpha \neq 0$). Let $e \in \overline{E_i}^{\mathcal{I}}$. Then $e \in E_i^{\mathcal{B}}$. We have $\sharp\{(e_1, \ldots, e_m) \in R^{\mathcal{B}} \mid e_i = e\} \geq \alpha$ and, by (3), we obtain $\sharp\{r \mid (r, e) \in \overline{U_i}^{\mathcal{I}}\} \geq \alpha$, from which $e \in (\geq \alpha\overline{U_i}^{\,-})^{\mathcal{I}}$.

(f) $\overline{E} \sqsubseteq\, \leq \beta\,\overline{U_i}^{\,-}$ (when $\mathrm{CARD}_R(R, U_i, E_i) = (\alpha, \beta)$ and $\beta \neq \infty$). The proof is similar to the previous case.

(g) $\overline{E} \sqsubseteq\, \geq \alpha\,\overline{U_i}^{\,-}$ (when $\mathrm{REF}(R, U_i, E_i) = (\alpha, \beta)$ and $\alpha \neq 0$). The proof is similar to case 2e.

(h) $\overline{E} \sqsubseteq\, \leq \beta\,\overline{U_i}^{\,-}$ (when $\mathrm{REF}(R, U_i, E_i) = (\alpha, \beta)$ and $\beta \neq \infty$). The proof is similar to case 2e.

3. $\mathrm{ATT}(E) = \langle A_1 : D_1, \ldots, A_h : D_h \rangle$. Let us consider all axioms in $\tau_{att}^E \cup \tau_{card_A}^E$:

(a) $\exists \overline{A_i}^{\,-} \sqsubseteq \overline{D_i}$. Let $a \in (\exists \overline{A_i}^{\,-})^{\mathcal{I}}$. Then there is $e \in \Delta^{\mathcal{I}}$ such that $(e, a) \in \overline{A_i}^{\mathcal{I}}$. As $\overline{A_i}^{\mathcal{I}} = A_i^{\mathcal{B}}$, we have $e \in \Delta^{\mathcal{B}}$ and $a \in \Lambda^{\mathcal{B}}$. It follows that $a \in D_i^{\mathcal{B}}$.

(b) $\overline{E} \sqsubseteq\, \geq \alpha\,\overline{A_i}$ (when $\mathrm{CARD}_A(A_i, E) = (\alpha, \beta)$ and $\alpha \neq 0$). Let $e \in \overline{E}^{\mathcal{I}}$. Then $e \in E^{\mathcal{B}}$. Thus, $\sharp\{a \mid (e, a) \in A^{\mathcal{B}}\} \geq \alpha$ and $\sharp\{a \mid (e, a) \in A^{\mathcal{I}}\} \geq \alpha$. Therefore, $e \in (\geq \alpha\,\overline{A_i})^{\mathcal{I}}$.

(c) $\overline{E} \sqsubseteq\, \leq \beta\,\overline{A_i}$ (when $\mathrm{CARD}_A(A_i, E) = (\alpha, \beta)$ and $\beta \neq \infty$). The proof is similar to the previous case.

4. E_1 ISA E_2. We have $\overline{E_1}^{\mathcal{I}} = E_1^{\mathcal{B}} \subseteq E_2^{\mathcal{B}} = \overline{E_2}^{\mathcal{I}}$, and so $\mathcal{I} \models \tau_{isa}^{E_1, E_2}$.

5. $\{E_1, \ldots, E_n\}$ DISJ E. We have $E_i^{\mathcal{B}} \subseteq E^{\mathcal{B}}$, for $1 \leq i \leq n$, and $E_i^{\mathcal{B}} \cap E_j^{\mathcal{B}} = \emptyset$ for $1 \leq i < j \leq n$. Hence, $\mathcal{I} \models \tau_{disj}^{\{E_1, \ldots, E_n\}, E}$.

6. $\{E_1, \ldots E_n\}$ COV E. Similarly to the previous case.

Thus, $\mathcal{I} \models \tau(\Sigma)$.

($\Leftarrow$) Let $\mathcal{T} = (\Delta^{\mathcal{T}}, \cdot^{\mathcal{T}})$ be a model of $\tau(\Sigma)$ such that $\overline{E}^{\mathcal{I}} \neq \emptyset$. Without loss of generality, we may assume that $\mathcal{T}$ is a *tree model* (see, e.g., [12, 6]). We construct domain sets $\{B_D\}_{D \in \mathcal{D}}$ and a legal database state $\mathcal{B} = (\Delta^{\mathcal{B}} \cup \Lambda^{\mathcal{B}}, \cdot^{\mathcal{B}})$ for the ER_{bool} schema Σ by taking $B_D = \Lambda_D^{\mathcal{B}} = D^{\mathcal{B}} = \overline{D}^{\mathcal{T}}$, for $D \in \mathcal{D}$, $\Lambda^{\mathcal{B}} = \bigcup_{D \in \mathcal{D}} \Lambda_D^{\mathcal{B}}$ and $\Delta^{\mathcal{B}} = \Delta^{\mathcal{T}} \setminus \Lambda^{\mathcal{B}}$; further we set $E^{\mathcal{B}} = \overline{E}^{\mathcal{T}}$, for every $E \in \mathcal{E}$, $A^{\mathcal{B}} = \overline{A}^{\mathcal{T}} \cap (\Delta^{\mathcal{B}} \times \Lambda^{\mathcal{B}})$, for every $A \in \mathcal{A}$, and, for every $R \in \mathcal{R}$ with $\mathrm{REL}(R) = \langle U_1 : E_1, \ldots, U_m : E_m \rangle$, we set

$$R^{\mathcal{B}} = \big\{(e_1, \ldots, e_m) \in T_{\Delta^{\mathcal{T}}}(\{U_1, \ldots, U_m\}) \mid$$
$$\exists r \in \overline{R}^{\mathcal{T}} \text{ such that } (r, e_i) \in \overline{U_i}^{\mathcal{T}} \text{ for } 1 \leq i \leq m\big\}.$$

Observe that the function $\cdot^{\mathcal{B}}$ is as required by Definition 2 and $E^{\mathcal{B}} \neq \emptyset$. We show now that $\mathcal{B}$ satisfies every assertion of the ER_{bool} schema Σ.

1. $\text{REL}(R) = \langle U_1 \colon E_1, \ldots, U_m \colon E_m \rangle$. Let $(e_1, \ldots, e_m) \in R^{\mathcal{B}}$. Then there exists $r \in \overline{R}^{\mathcal{T}}$ such that $(r, e_i) \in \overline{U_i}^{\mathcal{T}}$, for $1 \leq i \leq m$. Since $\mathcal{T} \models \exists \overline{U_i}^- \sqsubseteq \overline{E_i}$, we obtain $e_i \in \overline{E_i}^{\mathcal{T}}$, and so $e_i \in E_i^{\mathcal{B}}$, for $1 \leq i \leq m$.

2. $\text{ATT}(E) = \langle A_1 : D_1, \ldots, A_h : D_h \rangle$. Let $(e, a_i) \in \Delta^{\mathcal{B}} \times \Lambda^{\mathcal{B}}$ with $(e, a_i) \in A_i^{\mathcal{B}}$, for $1 \leq i \leq h$. Then $(e, a_i) \in \overline{A_i}^{\mathcal{T}}$. As $\mathcal{T} \models \exists \overline{A_i}^- \sqsubseteq \overline{D_i}$, we have $a_i \in \overline{D_i}^{\mathcal{T}}$, from which $a_i \in D_i^{\mathcal{B}} \subseteq \Lambda^{\mathcal{B}}$.

3. $\text{CARD}_R(R, U, E) = (\alpha, \beta)$. Then we have $\text{REL}(R) = \langle U_1 \colon E_1, \ldots, U_m \colon E_m \rangle$ such that $U_i = U$ and $E_i = E$, for some U_i and E_i, $1 \leq i \leq m$. We have to show that, for every $e \in E^{\mathcal{B}}$,

$$\alpha \leq \sharp\{(e_1, \ldots, e_m) \in R^{\mathcal{B}} \mid e_i = e\} \leq \beta.$$

Consider the lower and upper bounds.

   (a) We may assume that $\alpha \neq 0$. Since $\mathcal{T} \models \overline{E} \sqsubseteq \geq \alpha \overline{U}^-$ and $E^{\mathcal{B}} = \overline{E}^{\mathcal{T}}$, there exist at least α distinct $r_1, \ldots, r_\alpha \in \Delta^{\mathcal{T}}$ such that $(r_j, e) \in \overline{U}^{\mathcal{T}}$, for $1 \leq j \leq \alpha$. Since $\mathcal{T} \models \exists \overline{U} \sqsubseteq \overline{R}$, we have $r_1, \ldots, r_\alpha \in \overline{R}^{\mathcal{T}}$. And since $\mathcal{T} \models \overline{R} \sqsubseteq \exists \overline{U_i}$ and $\mathcal{T} \models \geq 2 \overline{U_i} \sqsubseteq \bot$, for all $1 \leq i \leq m$, there are uniquely determined $e_k^j \in \Delta^{\mathcal{T}}$ such that $(r_j, e_k^j) \in \overline{U_k}^{\mathcal{T}}$ and $e_i^j = e$, for all $1 \leq j \leq \alpha$ and $1 \leq k \leq m$. Since $\mathcal{T}$ is a tree-like model, we have $e_k^j \neq e_{k'}^{j'}$ whenever $k \neq i$, $k' \neq i$ and either $k \neq k'$ or $j \neq j'$. Therefore, we have shown that exactly one tuple corresponds to each object in $\overline{R}^{\mathcal{T}}$ and vice versa. Then, by construction, $(e_1^j, \ldots, e_m^j) \in R^{\mathcal{B}}$ and $e_i^j = e$, for all $1 \leq j \leq \alpha$. It follows that $\sharp\{(e_1, \ldots, e_m) \in R^{\mathcal{B}} \mid e_i = e\} \geq \alpha$.

   (b) We may assume that $\beta \neq \infty$. The proof is similar to the previous item.

4. $\text{CARD}_A(A, E) = (\alpha, \beta)$. Let $e \in E^{\mathcal{B}} = \overline{E}^{\mathcal{T}}$. Consider the lower and upper bounds:

   (a) We may assume $\alpha \neq 0$. Since $\mathcal{T} \models \overline{E} \sqsubseteq \geq \alpha \overline{A}$ and $\mathcal{T} \models \exists \overline{A}^- \sqsubseteq \overline{D}$, for some D with $(A, D) \in \text{ATT}(E)$, we have $\sharp\{a \in D^{\mathcal{B}} \mid (e, a) \in \overline{A}^{\mathcal{T}}\} \geq \alpha$. Finally, as $A^{\mathcal{B}} = \overline{A}^{\mathcal{T}} \cap (\Delta^{\mathcal{B}} \times \Lambda^{\mathcal{B}})$, we obtain $\sharp\{a \mid (e, a) \in A^{\mathcal{B}}\} \geq \alpha$.

   (b) We may assume $\beta \neq \infty$. The proof is similar to the previous case.

5. $\text{REF}(R, U, E) = (\alpha, \beta)$. The proof is the same as in case 3.

6. $E_1 \text{ ISA } E_2$. This holds in $\mathcal{B}$ since $\mathcal{T} \models \overline{E_1} \sqsubseteq \overline{E_2}$ and $E_i^{\mathcal{B}} = \overline{E_i}^{\mathcal{T}}$, for $i \in \{1, 2\}$.

7. $\{E_1, \ldots, E_n\} \text{ DISJ } E$. This holds in $\mathcal{B}$ since $\mathcal{T} \models \overline{E_i} \sqsubseteq \overline{E}$, for all $1 \leq i \leq n$, and $\mathcal{T} \models \overline{E_i} \sqsubseteq \neg \overline{E_j}$, for all $1 \leq i < j \leq n$, and $E_i^{\mathcal{B}} = \overline{E_i}^{\mathcal{T}}$, for $1 \leq i \leq n$.

8. $\{E_1, \ldots E_n\} \text{ COV } E$. Similar to the previous case.

It follows from this lemma and the mutual reducibility between the various reasoning problems in ER_{bool} that we have the following complexity result:

13

Theorem 3. *Reasoning over ER_{bool} conceptual schemas is* NP-*complete.*

Proof. The upper bound follows from Lemma 2 and Theorem 1. To prove NP-hardness we provide a polynomial reduction of the 3SAT problem, which is known to be NP-complete, to the entity consistency problem. Let an instance of 3SAT be given by a set ϕ of 3-clauses $c_i = a_i^1 \vee a_i^2 \vee a_i^3$ over some finite set Λ of literals. We define an ER_{bool} schema Σ_ϕ as follows:

- the signature $\mathcal{L}$ of Σ_ϕ is given by $\mathcal{E} = \{\overline{a} \mid a \in \Lambda\} \cup \{\overline{c} \mid c \in \phi\} \cup \{\overline{\phi}, \top\}, \mathcal{A} = \emptyset,$
  $\mathcal{R} = \emptyset, \mathcal{U} = \emptyset, \mathcal{D} = \emptyset$;
- $\overline{\phi}$ ISA $\overline{c}$, for all $c \in \phi$;
- $(\mathcal{E} \setminus \{\top\})$ COV $\top$, $\quad \{\overline{a}, \overline{\neg a}\}$ COV $\top$, for all $a \in \Lambda$,
  $\{\overline{a_i^1}, \overline{a_i^2}, \overline{a_i^3}\}$ COV $\overline{c_i}$, for all $c_i \in \phi$, $c_i = a_i^1 \vee a_i^2 \vee a_i^3$;
- $\{\overline{a}, \overline{\neg a}\}$ DISJ $\top$, for all $a \in \Lambda$;
- ATT, REL, CARD$_R$, CARD$_A$, REF are empty functions.

Now we show the following claim:

Claim. ϕ is satisfiable iff the entity $\overline{\phi}$ is consistent w.r.t. the schema Σ_ϕ.

($\Rightarrow$) Let $\mathcal{J} \models \phi$. Define a legal database state $\mathcal{B} = (\Delta^{\mathcal{B}}, \cdot^{\mathcal{B}})$ by taking $\Delta^{\mathcal{B}} = \{o\}$, $\top^{\mathcal{B}} = \{o\}$ and, for every $\overline{E} \in \mathcal{E} \setminus \{\top\}$, $\overline{E}^{\mathcal{B}} = \{o\}$ if $\mathcal{J} \models E$ and $\overline{E}^{\mathcal{B}} = \emptyset$ if $\mathcal{J} \not\models E$. We show that $\mathcal{B}$ is indeed a legal database state for Σ_ϕ. Since $\mathcal{J} \models \phi$, we have $\mathcal{J} \models c_i$ for all $c_i \in \phi$, and, by construction, $\overline{c_i}^{\mathcal{B}} = \{o\}$. This means that every ISA assertion in Σ_ϕ is satisfied by $\mathcal{B}$. Consider now some $c_i \in \phi$. Then $\mathcal{J} \models a_i^k$ for at least one of a_i^1, a_i^2 or a_i^3, which means that $\overline{a_i^k}^{\mathcal{B}} = \{o\}$. It follows that the assertion $\{\overline{a_i^1}, \overline{a_i^2}, \overline{a_i^3}\}$ COV $\overline{c_i}$ holds in $\mathcal{B}$. The assertion $(\mathcal{E} \setminus \{\top\})$ COV $\top$ holds, since $\overline{E}^{\mathcal{B}} \subseteq \{o\}$, $\overline{\phi}^{\mathcal{B}} = \{o\}$ and $\top^{\mathcal{B}} = \{o\}$, for every $\overline{E} \in \mathcal{E} \setminus \{\top\}$. It should also be clear that every assertion $\{\overline{a}, \overline{\neg a}\}$ COV $\top$, for $a \in \Lambda$, holds in $\mathcal{B}$. Since only one of $a, \neg a$ is satisfied by $\mathcal{J}$, the other one will be interpreted in $\mathcal{B}$ as the empty set, so every assertion in DISJ holds, too. Thus, $\mathcal{B}$ is a legal database state for Σ_ϕ, with $\overline{\phi}^{\mathcal{B}} \neq \emptyset$.

($\Leftarrow$) Let $\mathcal{B} = (\Delta^{\mathcal{B}}, \cdot^{\mathcal{B}})$ be a legal database state for Σ_ϕ such that $o \in \overline{\phi}^{\mathcal{B}}$, for some $o \in \Delta^{\mathcal{B}}$. Construct a model $\mathcal{J}$ for ϕ by taking, for every propositional variable p in ϕ, $\mathcal{J} \models p$ iff $o \in \overline{p}^{\mathcal{B}}$. We show that $\mathcal{J} \models \phi$. Indeed, as $o \in \overline{\phi}^{\mathcal{B}}$ and $\overline{\phi}$ ISA $\overline{c_i}$, we have $o \in \overline{c_i}^{\mathcal{B}}$, for $1 \leq i \leq n$. Since, for every c_i, we have $\{\overline{a_i^1}, \overline{a_i^2}, \overline{a_i^3}\}$ COV $\overline{c_i}$, there is a_i^k in c_i such that $o \in (\overline{a_i^k})^{\mathcal{B}}$. Now, if a_i^k is a variable then, by the construction of $\mathcal{J}$, we have $\mathcal{J} \models a_i^k$, and so $\mathcal{J} \models c_i$. Otherwise, $a_i^k = \neg p$ and, since $\{\overline{a_i^k}, \overline{p}\}$ DISJ $\top$, $o \notin \overline{p}^{\mathcal{B}}$. Therefore, by the construction of $\mathcal{J}$, $\mathcal{J} \not\models p$, i.e., $\mathcal{J} \models a_i^k$, and so $\mathcal{J} \models c_i$.

6 Reasoning over ER_{ref} Schemas

Denote by ER_{ref} the modelling language without the Booleans and ISA between relationships, but with the possibility to express ISA and disjointness between entities (i.e.,

$\text{ISA}_R = \emptyset$, $\text{COV}_R = \emptyset$, $\text{DISJ}_R = \emptyset$ and $\text{COV}_E = \emptyset$). Thus, ER_{ref} is essentially ER_{bool} without the covering constructor.

In this section we show that checking entity consistency in ER_{ref} is an NLOGSPACE-complete problem. Consider the reduction τ from Section 5. It is not difficult to check that τ is *logspace bounded*. At the same time, for every ER_{ref} schema Σ, the knowledge base $\tau(\Sigma)$ is a *DL-Lite$_{krom}$* knowledge base, because we do not have τ_{cov} in this case. Thus, as a consequence of Lemma 2, the problem of entity consistency for ER_{ref} can be logspace reduced to the NLOGSPACE-complete problem of concept satisfiability w.r.t. *DL-Lite$_{krom}$* knowledge bases [1]. So the entity consistency problem w.r.t. ER_{ref} schemas is in NLOGSPACE as well.

To establish the lower bound, we consider the reachability problem in oriented graphs, or the MAZE problem, which is known to be NLOGSPACE-hard; see, e.g., [15]. Let $G = (V, E, s, t)$ be an instance of MAZE, where s, t are the initial and terminal vertices of (V, E), respectively. We can encode this instance in ER_{ref} using the following schema Σ_G:

$$\overline{u} \text{ ISA } \overline{v}, \quad \text{for all } (u, v) \in E, \quad \text{and} \quad \{\overline{s}, \overline{t}\} \text{ DISJ } \overline{O},$$

where $\overline{O}$ is a fresh entity. Clearly, we have the following:

Claim. The terminal node t is reachable from s in $G = (V, E, s, t)$ iff the entity $\overline{s}$ is *not* consistent w.r.t. Σ_G.

As NLOGSPACE=CONLOGSPACE (by the Immerman-Szelepcsényi theorem; see, e.g., [15]) and the above reduction is logspace bounded, it follows that the problem of entity consistency in ER_{ref} is NLOGSPACE-hard. This result coupled with the membership in NLOGSPACE showed above gives us the following complexity result:

Theorem 4. *The entity consistency problem for ER_{ref} is* NLOGSPACE-*complete.*

7   Conclusions

This paper provides new complexity results for reasoning over Extended Entity-Relationship (EER) models with different modelling constructors. Starting from the EXPTIME result [5] for reasoning over the fully-fledged EER language, we prove that the same complexity holds even if we drop the Boolean constructors (disjointness and covering) on relationships. This result shows that ISA between relationships (with the Booleans on entities) is powerful enough to capture EXPTIME-hard problems. To illustrate that the presence of relationship hierarchies is a major source of complexity in reasoning, we show that avoiding them makes reasoning in ER_{bool} an NP-complete problem. Another source of complexity is covering constraints: indeed, we show that without relationship hierarchies and covering constraints reasoning problem is NLOGSPACE-complete.

The paper also establishes a tight correspondence between conceptual modelling languages and the *DL-Lite* family of description logics. Such a correspondence shows the usefulness of *DL-Lite* for representing and reasoning over conceptual models and ontologies.

References

1. Alessandro Artale, Diego Calvanese, Roman Kontchakov, and Michael Zakharyaschev. *DL-Lite* in the light of first-order logic. In *Proc. of the 22nd Nat. Conf. on Artificial Intelligence (AAAI 2007)*, 2007.

2. Franz Baader, Diego Calvanese, Deborah McGuinness, Daniele Nardi, and Peter F. Patel-Schneider, editors. *The Description Logic Handbook: Theory, Implementation and Applications*. Cambridge University Press, 2003.

3. Carlo Batini, Stefano Ceri, and Shamkant B. Navathe. *Conceptual Database Design, an Entity-Relationship Approach*. Benjamin and Cummings Publ. Co., 1992.

4. Sean Bechhofer, Frank van Harmelen, Jim Hendler, Ian Horrocks, Deborah L. McGuinness, Peter F. Patel-Schneider, and Lynn Andrea Stein. OWL Web Ontology Language reference. W3C Recommendation, February 2004. Available at http://www.w3.org/TR/owl-ref/.

5. Daniela Berardi, Diego Calvanese, and Giuseppe De Giacomo. Reasoning on UML class diagrams. *Artificial Intelligence*, 168(1–2):70–118, 2005.

6. Patrick Blackburn, Maarten de Rijke, and Yde Venema. *Modal Logic*, volume 53 of *Cambridge Tracts in Theoretical Computer Science*. Cambridge University Press, 2001.

7. Diego Calvanese and Giuseppe De Giacomo. Expressive description logics. In Baader et al. [2], chapter 5, pages 178–218.

8. Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. DL-Lite: Tractable description logics for ontologies. In *Proc. of the 20th Nat. Conf. on Artificial Intelligence (AAAI 2005)*, pages 602–607, 2005.

9. Diego Calvanese, Giuseppe De Giacomo, Domenico Lembo, Maurizio Lenzerini, and Riccardo Rosati. Data complexity of query answering in description logics. In *Proc. of the 10th Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR 2006)*, pages 260–270, 2006.

10. Diego Calvanese and Maurizio Lenzerini. On the interaction between ISA and cardinality constraints. In *Proc. of the 10th IEEE Int. Conf. on Data Engineering (ICDE'94)*, pages 204–213, Houston (Texas, USA), 1994. IEEE Computer Society Press.

11. Diego Calvanese, Maurizio Lenzerini, and Daniele Nardi. Unifying class-based representation formalisms. *J. of Artificial Intelligence Research*, 11:199–240, 1999.

12. Alexander Chagrov and Michael Zakharyaschev. *Modal Logic*, volume 35 of *Oxford Logic Guides*. 1997.

13. Giuseppe Di Battista and Maurizio Lenzerini. Deductive entity-relationship modeling. *IEEE Trans. on Knowledge and Data Engineering*, 5(3):439–450, 1993.

14. Ramez A. ElMasri and Shamkant B. Navathe. *Fundamentals of Database Systems*. Addison Wesley Publ. Co., 5th edition, 2007.

15. Dexter Kozen. *Theory of Computation*. Springer, 2006.

16. Maurizio Lenzerini and Paolo Nobili. On the satisfiability of dependency constraints in entity-relationship schemata. *Information Systems*, 15(4):453–461, 1990.