The Complexity of Temporal Description Logics with Rigid Roles and Restricted TBoxes: In Quest of Saving a Troublesome Marriage

Víctor Gutiérrez-Basulto, Jean Christoph Jung, and Thomas Schneider

Department of Computer Science, Universität Bremen
{victor, jeanjung, ts}@cs.uni-bremen.de

1 Introduction

Temporal description logics (TDLs) extend classical DLs, providing built-in means to represent and reason about temporal aspects of knowledge. The importance of TDLs stems from the need of relevant applications to capture temporal and dynamic aspects of knowledge, e.g., in medical and life science ontologies, which are very large but still demand efficient reasoning, such as SNOMED CT and FMA [9], and the gene ontology (GO) [20]. A natural task is to model *dynamic* knowledge about patient histories against *static* medical knowledge (e.g., about diseases): e.g., the temporal concept $C := \mathbf{E}\Diamond\exists\,\mathsf{requiresTransfusion}.\top$ describes a patient who may need a blood transfusion in the future, and the axiom $\mathsf{Anemic} \sqsubseteq C$ says that this applies to anemic people. In contrast, $\mathsf{Anemia} \sqsubseteq \mathsf{Disorder}$ represents static knowledge.

A notable approach to designing TDLs is to combine DLs with temporal logics commonly used in software/hardware verification such as LTL, CTL$^{(*)}$, and to provide a two-dimensional product-like semantics [19, 11, 17]. The combination allows various design choices, e.g., we can restrict the scope of temporal operators to certain types of entities (such as concepts, roles, axioms), or declare some DL concepts or roles as rigid, meaning that their interpretation will not change over time. The need for rigid roles in TDL applications, e.g., in biomedical ontologies to accurately capture life-time relations, has been identified [7]. For example, the role $\mathsf{hasBloodType}$ should be rigid since a human's blood type does not change during their lifetime.

Alas, TDLs based on the Boolean-complete DL $\mathcal{ALC}$ with rigid roles cannot be effectively used since they become undecidable when temporal operators are applied to concepts and a general TBox is allowed [11, 15]. This is the case even if we severely restrict the temporal operators available and use the sub-Boolean DL $\mathcal{EL}$, whose standard reasoning problems are tractable, instead of $\mathcal{ALC}$ [1, 15]. In the light of these results, several efforts have been devoted to design decidable TDLs with rigid roles [3, 2]; e.g., decidability can be recovered by using a lightweight DL component based on *DL-Lite*. Both the $\mathcal{EL}$ and *DL-Lite* families underlie prominent profiles of the OWL standard.

Interestingly, no research has been yet devoted to TDLs based on $\mathcal{EL}$ in the presence of restricted TBoxes, such as classical TBoxes, which consist solely of definitions of the form $A \equiv C$ with A atomic and unique, or acyclic TBoxes, which additionally forbid syntactic cycles in definitions. This is surprising since in the presence of general TBoxes TDLs based on $\mathcal{EL}$ tend to be as complex as the $\mathcal{ALC}$ variant [3, 13, 15].

These considerations lead us to investigating TDLs with rigid roles based on $\mathcal{EL}$ and the (branching-time) CTL allowing for temporal concepts and acyclic TBoxes. We are convinced that TDLs designed in this way are suitable for temporal extensions of biomedical ontologies: large parts of SNOMED CT and GO are acyclic $\mathcal{EL}$-TBoxes.

Our main contributions are algorithms for standard reasoning problems and (mostly tight) complexity bounds. We begin by showing that the combination of CTL and $\mathcal{ALC}$ with empty and acyclic TBoxes is decidable. Our nonelementary upper bound is optimal even when the set of temporal operators is restricted to $\mathbf{E}\Diamond$ ("possibly eventually") or $\mathbf{E}\bigcirc$ ("possibly next"). We then replace $\mathcal{ALC}$ with $\mathcal{EL}$ and maintain the restriction to $\mathbf{E}\Diamond$, $\mathbf{E}\bigcirc$ and empty TBoxes. We particularly show that the resulting TDLs are decidable in PTIME with one of the two operators, and coNP-complete with both. To this aim, we employ canonical models, together with expansion vectors [16] in the case with both $\mathbf{E}\Diamond$, $\mathbf{E}\bigcirc$. Next, we lift the PTIME upper bound to the case of acyclic TBoxes, employing a completion algorithm in the style of those for $\mathcal{EL}$ and extensions, [5]. Finally, we show that the combination of $\mathbf{E}\Diamond$ with $\mathbf{A}\Box$ ("always globally") and acyclic TBoxes leads to a PSPACE-complete TDL, again employing a completion algorithm. An overview of existing and new results is given in Table 1, where CTL_X^Y denotes the combination of the DL X with the fragment of CTL restricted to the temporal operators Y. In particular, all the new results hold even if rigid concepts are also included.

Rigid roles? TBoxes	no general	yes general	yes acyclic	yes empty
$\text{CTL}_{\mathcal{ALC}}$	=EXPTIME [13]	undecidable [15]	nonelementary, decidable (1)	nonelem., decidable (1)
$\text{CTL}_{\mathcal{EL}}^{\mathbf{E}\Diamond}$	≤PTIME [13]	nonelementary [15]	≤PTIME (6)	≤PTIME (6)
$\text{CTL}_{\mathcal{EL}}^{\mathbf{E}\bigcirc}$	≤PTIME [13]	undecidable [15]	≤PTIME (6)	≤PTIME (6)
$\text{CTL}_{\mathcal{EL}}^{\mathbf{E}\bigcirc,\mathbf{E}\Diamond}$	=EXPTIME [13, 15]	undecidable [15]	≥coNP, (2) ≤coNEXPTIME (5)	=coNP (2)
$\text{CTL}_{\mathcal{EL}}^{\mathbf{E}\Diamond,\mathbf{A}\Box}$	=PSPACE [13]	nonelementary [15]	=PSPACE (9)	≤PSPACE (9)

Table 1. Previous and new complexity results. ≥ hardness, ≤ membership, = completeness. (n) refers to our Theorem or Corollary n.

The relatively low complexity that we obtain for $\mathcal{EL}$-based TDLs over restricted TBoxes are in sharp contrast with the undecidability and nonelementary lower bounds known for the same logics over general TBoxes [15]. With the restriction to acyclic TBoxes, we will thus identify the first computationally well-behaved TDLs with rigid roles based on $\mathcal{EL}$ and classical temporal logics.

Additional technical notions and proofs are in a report: http://tinyurl.com/ijcai15tdl

2 Preliminaries

We introduce $\text{CTL}_{\mathcal{ALC}}$, a TDL based on the classical DL $\mathcal{ALC}$. Let $\mathsf{N_C}$ and $\mathsf{N_R}$ be countably infinite sets of *concept names* and *role names*, respectively. We assume that $\mathsf{N_C}$ and $\mathsf{N_R}$ are partitioned into two countably infinite sets: $\mathsf{N_C^{rig}}$ and $\mathsf{N_C^{loc}}$ of *rigid concept* *names* and *local concept names*, respectively; and, $\mathsf{N}_\mathsf{R}^\mathsf{rig}$ and $\mathsf{N}_\mathsf{R}^\mathsf{loc}$ of *rigid role names* and *local role names*, respectively. CTL$_{\mathcal{ALC}}$-*concepts* C are defined by the grammar

$$C := \top \mid A \mid \neg C \mid C \sqcap D \mid \exists r.C \mid \mathbf{E}\bigcirc C \mid \mathbf{E}\Box C \mid \mathbf{E}(C\mathcal{U}D)$$

where A ranges over N_C, r over N_R. We use standard DL abbreviations [6] and temporal abbreviations $\mathbf{E}\Diamond C$, $\mathbf{A}\Box C$, $\mathbf{A}\Diamond C$ and $\mathbf{A}(C\,\mathcal{U}\,D)$ [10].

The semantics of classical DLs, such as $\mathcal{ALC}$, is given in terms of *interpretations* of the form $\mathcal{I} = (\Delta, \cdot^\mathcal{I})$, where Δ is a non-empty set called the *domain* and $\cdot^\mathcal{I}$ is an *interpretation function* that maps each $A \in \mathsf{N}_\mathsf{C}$ to a subset $A^\mathcal{I} \subseteq \Delta$ and each $r \in \mathsf{N}_\mathsf{R}$ to a binary relation $r^\mathcal{I} \subseteq \Delta \times \Delta$. The semantics of CTL$_{\mathcal{ALC}}$ is given in terms of temporal interpretations based on infinite trees [15]: A *temporal interpretation* based on an infinite tree $T = (W, E)$ is a structure $\mathfrak{I} = (T, (\mathcal{I}_w)_{w\in W})$ such that, for each $w \in W$, $\mathcal{I}_w$ is a DL interpretation with domain Δ; and, $r^{\mathcal{I}_w} = r^{\mathcal{I}_{w'}}$ and $A^{\mathcal{I}_w} = A^{\mathcal{I}_{w'}}$ for all $r \in \mathsf{N}_\mathsf{R}^\mathsf{rig}$, $A \in \mathsf{N}_\mathsf{C}^\mathsf{rig}$ and $w, w' \in W$. We usually write $A^{\mathfrak{I},w}$ instead of $A^{\mathcal{I}_w}$. The stipulation that all worlds share the same domain is called the *constant domain assumption (CDA)*. For Boolean-complete TDLs, CDA is the most general: increasing, decreasing and varying domains can all be reduced to it [11, Prop. 3.32]. For the sub-Boolean logics studied here, CDA is not w.l.o.g. Indeed, we identify a logic in which reasoning with increasing domains cannot be reduced to the constant domain case.

We now define the semantics of CTL$_{\mathcal{ALC}}$-concepts. A *path* in $T = (W, E)$ starting at a node w is an infinite sequence $\pi = w_0w_1w_2\cdots$ with $w_0 = w$ and $(w_i, w_{i+1}) \in E$. We write $\pi[i]$ for w_i, and use $\mathsf{Paths}(w)$ to denote the set of all paths starting at the node w. The mapping $\cdot^{\mathfrak{I},w}$ is extended from concept names to CTL$_{\mathcal{ALC}}$-concepts as follows.

$$\top^{\mathfrak{I},w} = \Delta \qquad (C \sqcap D)^{\mathfrak{I},w} = C^{\mathfrak{I},w} \cap D^{\mathfrak{I},w}$$

$$\begin{aligned}
(\exists r.C)^{\mathfrak{I},w} &= \{d \in \Delta \mid \exists e\,.\,(d,e) \in r^{\mathfrak{I},w} \wedge e \in C^{\mathfrak{I},w}\}\\
(\mathbf{E}\bigcirc C)^{\mathfrak{I},w} &= \{d \mid \exists \pi \in \mathsf{Paths}(w)\,.\,d \in C^{\mathfrak{I},\pi[1]}\}\\
(\mathbf{E}\Box C)^{\mathfrak{I},w} &= \{d \mid \exists \pi \in \mathsf{Paths}(w)\,.\,\forall j \geq 0\,.\,d \in C^{\mathfrak{I},\pi[j]}\}\\
(\mathbf{E}(C\mathcal{U}D))^{\mathfrak{I},w} &= \{d \mid \exists \pi \in \mathsf{Paths}(w)\,.\,\exists j \geq 0\,.\,(d \in D^{\mathfrak{I},\pi[j]} \wedge (\forall k < j\,.\,d \in C^{\mathfrak{I},\pi[k]}))\}
\end{aligned}$$

An *acyclic CTL$_{\mathcal{ALC}}$-TBox* $\mathcal{T}$ is a finite set of *concept definitions (CDs)* $A \equiv D$ with $A \in \mathsf{N}_\mathsf{C}$ and D a CTL$_{\mathcal{ALC}}$ concept, such that (1) no two CDs have the same left-hand side, and (2) there are no CDs $A_1 \equiv C_1, \ldots, A_k \equiv C_k$ in $\mathcal{T}$ such that A_{i+1} occurs in C_i for $1 \leq i \leq k$, where $A_{k+1} = A_1$.

A temporal interpretation $\mathfrak{I}$ is a *model* of a concept C if $C^{\mathfrak{I},\varepsilon} \neq \emptyset$; it is a model of an acyclic TBox $\mathcal{T}$, written $\mathfrak{I} \models \mathcal{T}$, if $A^{\mathfrak{I},w} = C^{\mathfrak{I},w}$ for all $A \equiv C \in \mathcal{T}$ and $w \in W$; it is a model of a *concept inclusion* $C \sqsubseteq D$, written $\mathfrak{I} \models C \sqsubseteq D$, if $C^{\mathfrak{I},w} \subseteq D^{\mathfrak{I},w}$ for all $w \in W$.

The two main reasoning tasks we consider are concept satisfiability and subsumption. A concept C is *satisfiable* relative to an acyclic TBox $\mathcal{T}$ if there is a common model of C and $\mathcal{T}$. A concept D *subsumes* a concept C relative to an acyclic TBox $\mathcal{T}$, written $\mathcal{T} \models C \sqsubseteq D$, if $\mathfrak{I} \models C \sqsubseteq D$ for all models $\mathfrak{I}$ of $\mathcal{T}$. If $\mathcal{T}$ is empty, we write $\models C \sqsubseteq D$.

3 First Observations

We start by observing that the combination of CTL and $\mathcal{ALC}$ with rigid roles relative to empty and acyclic TBoxes is decidable and inherently nonelementary. In a nutshell, we show the upper bounds using a variant of the quasimodel technique [11, Thm. 13.6]; the lower bound follows from the fact that satisfiability for the product modal logics $\mathsf{S4}\times\mathsf{K}$ and $\mathsf{K}\times\mathsf{K}$ is inherently nonelementary [12]. Indeed, the fragment of $\mathrm{CTL}_{\mathcal{ALC}}$ allowing $\mathbf{E}\Diamond$ ($\mathbf{E}\bigcirc$) as the only temporal operator is a notational variant of $\mathsf{S4}\times\mathsf{K}$ ($\mathsf{K}\times\mathsf{K}$) [15].

Theorem 1. *Concept satisfiability relative to acyclic and empty TBoxes for* $\mathrm{CTL}_{\mathcal{ALC}}$ *with rigid roles is decidable and inherently nonelementary.*

With Theorem 1 and the third column of Table 1 in mind, we particularly set as our goal the identification of elementary (ideally tractable) TDLs. To this aim, we study combinations of (fragments of) CTL with the lightweight DL $\mathcal{EL}$. $\mathrm{CTL}_{\mathcal{EL}}$ is the fragment of $\mathrm{CTL}_{\mathcal{ALC}}$ that disallows the constructor $\neg$ (and thus the abbreviations $C \sqcup D$, $\forall r.C$, $\mathbf{A}\Box, \dots$). The standard reasoning problem for $\mathrm{CTL}_{\mathcal{EL}}$, as for $\mathcal{EL}$, is concept subsumption since each concept and TBox are trivially satisfiable. In what follows we consider various fragments of $\mathrm{CTL}_{\mathcal{EL}}$ obtained by restricting the available temporal operators. We denote the fragments by putting the allowed operators as a superscript. In this context, we view each of the operators $\mathbf{E}\Diamond$, $\mathbf{A}\Box$ as primitive instead of as an abbreviation.

In order to keep the presentation of our main results accessible, in Sections 5-6, we concentrate on the case where only rigid role names and local concept names are present. Later on, in Section 7, we explain how to deal with the general case.

4 $\mathrm{CTL}^{\mathbf{E}\bigcirc,\mathbf{E}\Diamond}_{\mathcal{EL}}$ relative to the Empty TBox

We begin by investigating the complexity of subsumption relative to the empty TBox for a TDL whose subsumption relative to general TBoxes is undecidable: $\mathrm{CTL}^{\mathbf{E}\bigcirc,\mathbf{E}\Diamond}_{\mathcal{EL}}$.

Theorem 2. *Concept subsumption relative to the empty TBox is* coNP*-complete for* $\mathrm{CTL}^{\mathbf{E}\bigcirc,\mathbf{E}\Diamond}_{\mathcal{EL}}$ *with rigid roles and in* PTime *for* $\mathrm{CTL}^{\mathbf{E}\bigcirc}_{\mathcal{EL}}$ *and* $\mathrm{CTL}^{\mathbf{E}\Diamond}_{\mathcal{EL}}$ *with rigid roles.*

coNP-hardness is obtained by embedding $\mathcal{EL}$ plus transitive closure into $\mathrm{CTL}^{\mathbf{E}\bigcirc,\mathbf{E}\Diamond}_{\mathcal{EL}}$; the jump in complexity comes from the ability to express disjunctions, e.g., $\models \mathbf{E}\Diamond C \sqsubseteq C \sqcup \mathbf{E}\bigcirc\mathbf{E}\Diamond C$. We next explain coNP-membership; the PTime results are a byproduct and improved later.

We proceed in two steps: first we provide a characterization of $\models C \sqsubseteq D$ where C is an $\mathrm{CTL}^{\mathbf{E}\bigcirc}_{\mathcal{EL}}$-concept and D an $\mathrm{CTL}^{\mathbf{E}\bigcirc,\mathbf{E}\Diamond}_{\mathcal{EL}}$-concept. Next we generalize this characterization to $\mathrm{CTL}^{\mathbf{E}\bigcirc,\mathbf{E}\Diamond}_{\mathcal{EL}}$-concepts C.

Given a $\mathrm{CTL}^{\mathbf{E}\bigcirc}_{\mathcal{EL}}$-concept C, the *description tree* $t_C = (V_C, L_C, E_C)$ *for* C is a labeled graph corresponding to C's syntax tree; we denote its *root* by x_C. For example, if $C = \mathbf{E}\bigcirc(\exists r.A \sqcap \exists s.B)$, then t_C is given in Figure 1, left.

For plain $\mathcal{EL}$, we have $\models C \sqsubseteq D$ if and only if there is a homomorphism from t_D to t_C, which can be tested in polynomial time [8]. This criterion cannot directly be transferred to $\mathrm{CTL}^{\mathbf{E}\bigcirc}_{\mathcal{EL}}$ because t_C does not explicitly represent all pairs of worlds and domain elements whose existence is implied by t_C, e.g., for $\models \mathbf{E}\bigcirc\exists r.A \sqsubseteq \exists r.\mathbf{E}\bigcirc A$ with r rigid, there is no homomorphism from t_D to t_C. We overcome this problem by transforming t_C into a *canonical model* $\mathfrak{I}_C$ of C, i.e., (1) its distinguished root is an instance of C and (2) $\mathfrak{I}_C$ homomorphically embeds into every model of C. The

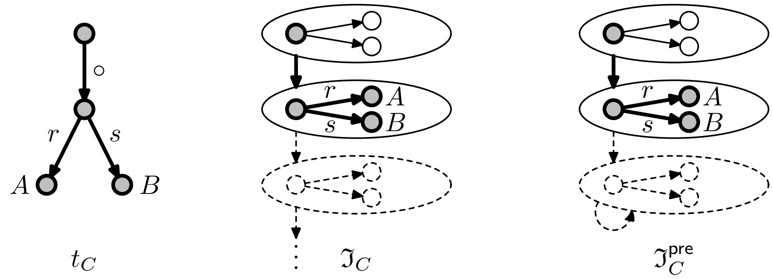

Fig. 1. Description tree t_C, canonical model $\mathfrak{I}_C$, finite representation $\mathfrak{I}_C^{\mathsf{pre}}$ for $C = \mathbf{E}\bigcirc(\exists r.A \sqcap \exists s.B)$

construction of $\mathfrak{I}_C$ from t_C is straightforward: for every node with an incoming $\bigcirc$-edge (r-edge, r being a role) create a fresh world (domain element); for the root x_C create *both* a world and domain element. The temporal relation and the interpretation of r and concept names is read off E_C and L_C. To transform (W, R) into an infinite tree, we add an infinite path of fresh worlds to every world without R-successor. The canonical model for the above concept C is shown in Fig. 1, center; the infinite path of worlds is dashed.

From (1), (2), and the preservation properties of homomorphisms, we obtain:

Lemma 3. *For all* $\mathrm{CTL}_{\mathcal{EL}}^{\mathbf{E}\bigcirc}$*-concepts* C *and all* $\mathrm{CTL}_{\mathcal{EL}}^{\mathbf{E}\bigcirc,\mathbf{E}\Diamond}$*-concepts* D*, we have* $\models C \sqsubseteq D$ *if and only if* $x_C \in D^{\mathfrak{I}_C, x_C}$.

Now $x_C \in D^{\mathfrak{I}_C, x_C}$ can be verified by model-checking D in world x_C and element x_C of $\mathfrak{I}_C^{\mathsf{pre}}$, which is the polynomial-sized modification of $\mathfrak{I}$ where the lastly added infinite path of worlds is replaced by a single loop, see Fig. 1, right. Since $\mathfrak{I}_C$ is the unraveling of $\mathfrak{I}_C^{\mathsf{pre}}$ into the temporal dimension, $\mathfrak{I}_C$ and $\mathfrak{I}_C^{\mathsf{pre}}$ satisfy the same concepts in their roots. Theorem 2 for $\mathrm{CTL}_{\mathcal{EL}}^{\mathbf{E}\bigcirc}$ thus follows. The $\mathrm{CTL}_{\mathcal{EL}}^{\mathbf{E}\Diamond}$ part can be obtained by representing every $\mathbf{E}\Diamond$ in C by a $\bigcirc$-edge in t_C and adapting the notion of a homomorphism.

For $\mathrm{CTL}_{\mathcal{EL}}^{\mathbf{E}\bigcirc,\mathbf{E}\Diamond}$, we use expansion vectors introduced in [16], applied to the temporal dimension. Let C be a $\mathrm{CTL}_{\mathcal{EL}}^{\mathbf{E}\bigcirc,\mathbf{E}\Diamond}$-concept with n occurrences of $\mathbf{E}\Diamond$. An *expansion vector for* C is an n-tuple $U = (u_1, \ldots, u_n)$ of integers $u_i \geqslant 0$. Intuitively, U fixes a specific number of temporal steps taken for each $\mathbf{E}\Diamond$ in C when constructing t_C and $\mathfrak{I}_C$. More precisely, we denote with $C[U]$ the $\mathrm{CTL}_{\mathcal{EL}}^{\mathbf{E}\bigcirc}$-concept obtained from C by replacing the i-th occurrence of $\mathbf{E}\Diamond$ with $(\mathbf{E}\bigcirc)^{u_i}$, i.e., i times $\mathbf{E}\bigcirc$. For example, if $C = \mathbf{E}\Diamond\exists r.\mathbf{E}\Diamond(A \sqcap \mathbf{E}\bigcirc B)$ and $U = (2, 0)$, then $C[U] = \mathbf{E}\bigcirc\mathbf{E}\bigcirc\exists r.(A \sqcap \mathbf{E}\bigcirc B)$.

Let $\mathbb{U}_C^m = \{(u_1, \ldots, u_n) \mid u_i \leqslant m \text{ for all } i\}$. We denote with $\mathsf{tdepth}(D)$ the nesting depth of temporal operators in D. We use expansion vectors with entries bounded by $\mathsf{tdepth}(D)$ to reduce $\not\models C \sqsubseteq D$ for $\mathrm{CTL}_{\mathcal{EL}}^{\mathbf{E}\bigcirc,\mathbf{E}\Diamond}$ to the case where C is from $\mathrm{CTL}_{\mathcal{EL}}^{\mathbf{E}\bigcirc}$.

Lemma 4. *For all* $\mathrm{CTL}_{\mathcal{EL}}^{\mathbf{E}\bigcirc,\mathbf{E}\Diamond}$*-concepts* C, D*, we have* $\models C \sqsubseteq D$ *if and only if* $\models C[\overline{U}] \sqsubseteq D$ *for all* $\overline{U} \in \mathbb{U}_C^{\mathsf{tdepth}(D)+1}$.

Together with Lemma 3, this yields the desired polynomial-time guess-and-check procedure for deciding $\models C \sqsubseteq D$.

5 $\mathrm{CTL}_{\mathcal{EL}}^{\mathbf{E}\bigcirc}$ and $\mathrm{CTL}_{\mathcal{EL}}^{\mathbf{E}\Diamond}$ relative to Acyclic TBoxes

The results of Theorem 2 transfer to acyclic TBoxes with an exponential blowup due to unfolding [18], that is:

Corollary 5. *Concept subsumption relative to acyclic* $\mathrm{CTL}_{\mathcal{EL}}^{\mathbf{E}\bigcirc,\mathbf{E}\Diamond}$*-TBoxes with rigid roles is in* coNEXPTIME.

For the subfragments $\mathrm{CTL}_{\mathcal{EL}}^{\mathbf{E}\bigcirc}$ and $\mathrm{CTL}_{\mathcal{EL}}^{\mathbf{E}\Diamond}$, we can even show polynomial complexity:

Theorem 6. *Concept subsumption relative to acyclic* $\mathrm{CTL}_{\mathcal{EL}}^{\mathbf{E}\bigcirc}$*- and* $\mathrm{CTL}_{\mathcal{EL}}^{\mathbf{E}\Diamond}$*-TBoxes with rigid roles is in* PTIME.

We first concentrate on the $\mathbf{E}\Diamond$ case and explain below how to deal with the $\mathbf{E}\bigcirc$ one. We focus w.l.o.g. on subsumption between concept *names* and assume that the input TBox is in normal form (NF), i.e., each axiom is of the shape $A \equiv A_1 \sqcap A_2$, $A \equiv \mathbf{E}\Diamond A_1$, or $A \equiv \exists r.A_1$, where $A_i \in \mathsf{N_C} \cup \{\top\}$ and $r \in \mathsf{N_R}$. As usual, a subsumption-equivalent TBox in NF can be computed in polynomial time [4]. We use CN and ROL to denote the sets of concept names and roles occurring in $\mathcal{T}$.

To prove a PTIME upper bound, we devise a completion algorithm in the style of those known for $\mathcal{EL}$ and (two-dimensional) extensions, cf. [5, 14], which build an abstract representation of the 'minimal' model of the input TBox $\mathcal{T}$ (in the sense of Horn logic). The main difficulty is that different occurrences of the same concept name in the TBox cannot all be treated uniformly (as it is the case for, say, $\mathcal{EL}$), due to the two-dimensional semantics. Instead, we have to carefully choose witnesses for $\mathbf{E}\Diamond A$ and $\exists r.A$, respectively. Our algorithm constructs a graph $G = (W, E, Q, R)$ based on a set W, a binary relation E, a mapping Q that associates with each $A \in \mathsf{CN}$ and each $w \in W$ a subset $Q(A, w) \subseteq \mathsf{CN}$, and a mapping R that associates with each rigid role $r \in \mathsf{ROL}$ a relation $R(r) \subseteq \mathsf{CN} \times W \times \mathsf{CN} \times W$. For brevity, we write $(A, w) \xrightarrow{r} (B, w')$ instead of $(A, w, B, w') \in R(r)$ and denote with E^* the reflexive, transitive closure of E.

The algorithm for deciding subsumption initializes G as follows. For all $r \in \mathsf{ROL}$, set $R(r) = \emptyset$. Set $W = \mathsf{CN}\times\mathsf{CN}\cup\{\mathbf{E}\Diamond A \mid A \in \mathsf{CN}\}$. Set $E = \{(\mathbf{E}\Diamond A, AA), (AB, A\top) \mid A, B \in \mathsf{CN}\}$. For all $A \in \mathsf{CN}$, set $Q(A, w) = \{\top, B\}$ if $w = AB$ and $Q(A, w) = \{\top\}$ otherwise.

Intuitively, the unraveling of (W, E) is the temporal tree underlying the minimal model and the mappings Q and R contain condensed information on how to interpret concepts and roles, respectively. More specifically, the data stored in $Q(A, \cdot)$ describes the temporal evolution of an instance of A. For example, $Q(A, AA)$ collects all concept names B such that $\mathcal{T} \models A \sqsubseteq B$; likewise, $Q(A, \mathbf{E}\Diamond A)$ captures everything that follows from $\mathbf{E}\Diamond A$. Finally, $Q(A, AB)$ contains concept names that are implied by B given that B appears in the temporal evolution of an instance of A, i.e., $B' \in Q(A, AB)$ implies $\mathcal{T} \models A \sqcap \mathbf{E}\Diamond B \sqsubseteq \mathbf{E}\Diamond(B \sqcap B')$.

After initialization, the algorithm extends G by applying the completion rules depicted in Figure 2 in three phases. In the first phase – also called FORWARD-phase, since definitions $A \equiv C \in \mathcal{T}$ are read as $A \sqsubseteq C$ – rules **F1**-**F3** are exhaustively applied in order to generate a fusion-like representation by adding witness-worlds and witness-existentials as demanded. Most notably, rule **F2** introduces a pointer to the structure representing the temporal evolution of an instance of B'.

F1	If $B \in Q(A, AA')$ & $B \equiv \mathbf{E}\Diamond B' \in \mathcal{T}$, then add (AA', AB') to E
F2	If $B \in Q(A, w)$ and $B \equiv \exists r.B' \in \mathcal{T}$, then set $(A, w) \xrightarrow{r} (B', B'B')$
F3	If $B \in Q(A, w)$ & $B \equiv A_1 \sqcap A_2 \in \mathcal{T}$, then add A_1, A_2 to $Q(A, w)$
C1	If $(BB, w) \in E$ and $(A, w') \xrightarrow{r} (B, BB)$, then add (w', w) to E
C2	If $(A, w) \xrightarrow{r} (B, BB)$, then **a)** $(A, w') \xrightarrow{r} (B, \mathbf{E}\Diamond B)$ for all $w' \neq w$ with $(w', w) \in E^*$ **b)** $(A, w') \xrightarrow{r} (B, w')$ for all w' with $(w', w) \notin E^*$
B1	If $B \in Q(A, w)$, $(w', w) \in E^*$, and $A' \equiv \mathbf{E}\Diamond B \in \mathcal{T}$, then add A' to $Q(A, w')$
B2	If $A \in Q(B, w)$, $(A', w') \xrightarrow{r} (B, w)$, and $A'' \equiv \exists r.A \in \mathcal{T}$ then add A'' to $Q(A', w')$
B3	If $A_1, A_2 \in Q(B, w)$ & $A \equiv A_1 \sqcap A_2 \in \mathcal{T}$ then add A to $Q(B, w)$

Fig. 2. Completion rules

Subsequently, G is extended to conform with the constant domain assumption and reflect rigidity of roles by exhaustively applying rules **C1**, **C2**. Here read **C2** as '*if two points are connected via r in some world, then they should be connected in all worlds*.' Note that $Q(B, \mathbf{E}\Diamond B)$ is used as a representative for the entire "past" of B in part **a)**.

In the final phase, BACKWARD-completion rules **B1**-**B3** are exhaustively applied in order to respect the 'backwards'-direction of definitions, i.e., definitions $A \equiv C \in \mathcal{T}$ are read as $A \sqsupseteq C$. This separation into a FORWARD and BACKWARD phase is sanctioned by acyclicity of the TBox. In fact, one run through each phase is enough; note that no new tuples are added to E or R in the BACKWARD-phase.

The following lemma shows correctness of our algorithm.

Lemma 7. *Let $\mathcal{T}$ be an acyclic $\mathrm{CTL}_{\mathcal{EL}}^{\mathbf{E}\Diamond}$-TBox in normal form. Then for all $A, B \in \mathsf{CN}$, we have $\mathcal{T} \models A \sqsubseteq B$ iff, after exhaustive rule application, $B \in Q(A, AA)$.*

To prove "$\Leftarrow$", we show that (a certain unraveling of) G "embeds" into every model of A and $\mathcal{T}$. For this purpose, we need to adapt the notion of a homomorphism to temporal interpretations and rigid roles. For "$\Rightarrow$", we construct from G a model $\mathfrak{I}$ of $\mathcal{T}$ with $d \in A^{\mathfrak{I},w} \setminus B^{\mathfrak{I},w}$ for some d, w. The algorithm runs in polynomial time: the size of the data structures W, E, and R is clearly polynomial and the mapping $Q(\cdot, \cdot)$ is extended in every rule application, so the algorithm stops after polynomially many steps.

Finally, we sketch two modifications of the algorithm such that it works for $\mathbf{E}\bigcirc$ instead of $\mathbf{E}\Diamond$. First, we have to use a non-transitive version of **B1**. Second, and a bit more subtly, we have to replace $\mathbf{E}\Diamond A \in W$ with $\mathbf{E}\bigcirc^k A$, $1 \leq k \leq |\mathcal{T}|$ to capture what is implied by $\mathbf{E}\bigcirc^k A$; more precisely, $B' \in Q(A, \mathbf{E}\bigcirc^k A)$ implies $\mathcal{T} \models \mathbf{E}\bigcirc^k A \sqsubseteq B'$, where $\mathbf{E}\bigcirc^k$ denotes $\mathbf{E}\bigcirc \cdots \mathbf{E}\bigcirc$ k times.

We next show that there is a jump in the complexity if increasing domains are considered instead of constant ones. Intuitively, this can be explained by the fact that increasing domains allow rigid roles to mimic the behaviour of the $\mathbf{A}\Box$-operator. In the next section, we show that adding $\mathbf{A}\Box$ to $\{\mathbf{E}\Diamond\}$ indeed leads to PSPACE-hardness.

Theorem 8. *Concept subsumption relative to acyclic $\mathrm{CTL}_{\mathcal{EL}}^{\mathbf{E}\bigcirc}$- and $\mathrm{CTL}_{\mathcal{EL}}^{\mathbf{E}\Diamond}$-TBoxes with rigid roles and increasing domains is* PSPACE*-hard.*

6 $\mathrm{CTL}_{\mathcal{EL}}^{\mathbf{E}\Diamond,\mathbf{A}\Box}$ relative to Acyclic TBoxes

We now add $\mathbf{A}\Box$ and observe an increase in complexity over acyclic TBoxes.

Theorem 9. *Concept subsumption relative to acyclic* $\mathrm{CTL}_{\mathcal{EL}}^{\mathbf{E}\Diamond,\mathbf{A}\Box}$*-TBoxes with rigid roles is* PSPACE*-complete.*

The lower bound is obtained via a reduction from QBF validity. For the upper bound, we again consider w.l.o.g. subsumption between concept *names* and assume that the acyclic TBox is in normal form, i.e., axioms are of the shape $A \equiv A_1 \sqcap A_2$, $A \equiv \mathbf{E}\Diamond A_1$, $A \equiv \mathbf{A}\Box A_1$, or $A \equiv \exists r.A_1$, where $A_i \in \mathsf{N_C} \cup \{\top\}$ and $r \in \mathsf{N_R}$. We also restrict ourselves again to only rigid roles. CN and ROL are used as before.

In contrast to the previous section, we cannot maintain the entire minimal model in memory since the added operator $\mathbf{A}\Box$ can be used to enforce models of exponential size. Instead, we will compute all concepts implied by the input concept A (the left-hand side of the subsumption to be checked) by iteratively visiting relevant parts of the minimal model. Our main tool for doing so are *traces*.

Definition 10. *A* trace *is a tuple* (σ, E, R) *where* σ *is a sequence* $(d_0, w_0) \cdots (d_n, w_n)$ *such that for all* $0 \leq i < n$ *one of the following is true. (1)* $d_i = d_{i+1}$ *and* $(w_i, w_{i+1}) \in E$*. (2)* $w_i = w_{i+1}$ *and* $(d_i, d_{i+1}) \in R(r)$ *for some* $r \in \mathsf{ROL}$.

Intuitively, traces represent paths through temporal interpretations, which in each step follow either the temporal relation (Def. 10, 1) or a DL relation r (2); so, in a pair (d, w), d can be thought of as a domain element and w as a world.

Our algorithm, whose basic structure is given by Alg. 1, enumerates on input $\mathcal{T}, A, B$, in a systematic tableau-like way, all traces that *must* appear in every model of A and $\mathcal{T}$. Note that in the context of Algorithm 1 a trace is used as the basis for inducing a richer structure that conforms with the constant domain assumption and captures rigidity; see Example 11 below. The algorithm also maintains an additional mapping Q that labels each point (d, w) of the trace (and all the induced points) with a set $Q(d, w) \subseteq \mathsf{CN}$. The set $Q(d, w)$ captures all concept names that are satisfied in the minimal model at points represented by (d, w).

Algorithm 1: Subsumption in $\mathrm{CTL}_{\mathcal{EL}}^{\mathbf{E}\Diamond,\mathbf{A}\Box}$

Input: Acyclic TBox $\mathcal{T}$, concept names A, B
Output: `true` if $\mathcal{T} \models A \sqsubseteq B$, `false` otherwise

$\sigma := (d_0, w_0)$; $Q(d_0, w_0) := \{A, \top\}$;
$E := \emptyset$; $R(r) := \emptyset$ for all $r \in \mathsf{ROL}$;
`expand`(σ, E, R);
return `true` if $B \in Q(d_0,w_0)$, `false` otherwise;
procedure `expand` (σ, E, R) **:**
    `complete` (σ, E, R, Q);
    if (σ, Q) *is periodic at* (i, j) **then**
        add (w_{j-1}, w_i) to E;
        `truncate`;
        `complete` (σ, E, R, Q);
        return;
    $(d, w) :=$ last element of σ;
    foreach $A \in Q(d, w)$ *with* $A \equiv \exists r.B \in \mathcal{T}$ **do**
        $Q(d', w) = \{B, \top\}$ for a fresh d';
        add (d, d') to $R(r)$;
        `expand` $(\sigma \cdot (d', w), E, R)$;
    foreach $A \in Q(d, w)$ *with* $A \equiv \mathbf{E}\Diamond B \in \mathcal{T}$ **do**
        $Q(d, w') = \{B, \top\}$ for a fresh w';
        add (w, w') to E;
        `expand` $(\sigma \cdot (d, w'), E, R)$;

R1	If $A \equiv A_1 \sqcap A_2 \in \mathcal{T}$ and $A \in Q_*(\cdot)$, then add A_1, A_2 to $Q_*(\cdot)$
R2	If $A \equiv A_1 \sqcap A_2 \in \mathcal{T}$ and $A_1, A_2 \in Q_*(\cdot)$, then add A to $Q_*(\cdot)$
R3	If $(d, d') \in R(r)$, $B \in Q(d', w)$, $A \equiv \exists r.B \in \mathcal{T}$, then add A to $Q(d, w)$
R4	If $B \in Q(d, w)$, $(w', w) \in E^*$, $A \equiv \mathbf{E}\Diamond B \in \mathcal{T}$, then add A to $Q(d, w')$
R5	If $B \in Q(d, w)$, $(w, w') \in E^*$, $B \equiv \mathbf{A}\Box A \in \mathcal{T}$, then add B, A to $Q(d, w')$
R6	If $(d, d') \in R(r)$, $B \in Q_{\mathsf{cert}}(d')$, $A \equiv \exists r.B \in \mathcal{T}$, then add A to $Q_{\mathsf{cert}}(d)$
R7	If $B \in Q_{\mathsf{cert}}(d)$, $A \equiv \mathbf{A}\Box B \in \mathcal{T}$, then add A to $Q_{\mathsf{cert}}(d)$
R8	If $B \in Q_{\mathsf{cert}}(d)$, then add B to $Q(d, w)$ for all w
R9	If $B \in Q_{\mathbf{A}\Box}(d, w)$, $A \equiv \mathbf{A}\Box B \in \mathcal{T}$, add A to $Q(d, w)$
R10	If $A \in Q(d, w)$, $A \equiv \mathbf{A}\Box B \in \mathcal{T}$, add A, B to $Q_{\mathbf{A}\Box}(d, w)$
R11	If $(d, d') \in R(r)$, $B \in Q_{\mathbf{A}\Box}(d', w)$, $A \equiv \exists r.B \in \mathcal{T}$, then add A to $Q_{\mathbf{A}\Box}(d, w)$
R12	If $A \in Q_{\mathbf{A}\Box}(d, w)$, $A \equiv \mathbf{E}\Diamond B \in \mathcal{T}$, w' added due to $A \in Q(d, w)$ in Line 18, $B' \in Q(d, w')$, $A' \equiv \mathbf{E}\Diamond B' \in \mathcal{T}$, then add A' to $Q_{\mathbf{A}\Box}(d, w)$

Fig. 3. Saturation rules. In **R1**, **R2** the set $Q_*(\cdot)$ ranges over all $Q(d, w)$, $Q_{\mathsf{cert}}(d)$, and $Q_{\mathbf{A}\Box}(d, w)$.

The basics of Algorithm 1 are the following. In Lines 1 and 2, it creates a trace consisting of a single point representing A and initializes the necessary data structures. In Line 3, the systematic expansion is set off. When that is finished, the algorithm just returns whether or not B (the right-hand of the subsumption) has been added during the expansion. As for the `expand` procedure:

– in Line 6 and 10, the algorithm updates the mapping Q;
– Line 7 contains some termination condition; and finally,
– the loops in Lines 13 & 17 enumerate all $\exists r.B$ and $\mathbf{E}\Diamond B$ that appear in the set $Q(d, w)$ of the last trace element and expand the trace to witness these concepts.

This basic description of the algorithm leaves open several points: *(i)* the precise behavior of the subroutine `complete`, *(ii)* when a trace is *periodic*, and *(iii)* what happens inside the `truncate` command in Line 9. Let us start with describing the subroutine `complete`. It uses additional mappings $Q_{\mathsf{cert}}(d) \subseteq \mathsf{CN}$ and $Q_{\mathbf{A}\Box}(d, w) \subseteq \mathsf{CN}$, which intuitively contain all the concept names that d satisfies *certainly*, i.e., in all worlds, and starting from world w, respectively. It proceeds in two steps. (1) Initialize undefined $Q(d, w)$ and $Q_{\mathsf{cert}}(d)$ with $\{\top\}$, and undefined $Q_{\mathbf{A}\Box}(d, w)$ with $Q_{\mathsf{cert}}(d)$. (2) Apply rules **R1**-**R12** in Figure 3 to $Q(\cdot)$, $Q_{\mathsf{cert}}(\cdot)$ and $Q_{\mathbf{A}\Box}(\cdot)$.

The number of rules is indeed scarily high; however, they can be divided into four digestible groups: **R1** and **R2** are used to ensure that all sets Q_* are closed under conjunction; **R3**-**R5** are used to complete $Q(\cdot)$. Note that **R1**-**R4** are already known from the algorithm of the previous section. Furthermore, **R6**-**R8** are used to deal with $Q_{\mathsf{cert}}(\cdot)$; and **R9**-**R12** to update $Q_{\mathbf{A}\Box}(\cdot)$. As an example of the interplay between the different mappings take **R9**: If B is certain for d starting in world w and $A \equiv \mathbf{A}\Box B$, then we also know that d satisfies A in w; and **R11** for the interplay between temporal operators and rigid roles: indeed, for r rigid, $\models \exists r.\mathbf{A}\Box B \sqsubseteq \mathbf{A}\Box\exists r.B$.

Example 11. Let $\mathcal{T} = \{A \equiv \mathbf{E}\Diamond A_1,\ A_1 \equiv \exists r.B,\ B \equiv \mathbf{E}\Diamond A_1\}$ be the input TBox; and $\mathcal{T} \models A \sqsubseteq A_1$ is to be checked. Figure 4 (left) shows the trace initiated at (d_0, w_0) with $Q(d_0, w_0) = \{\top, A\}$, and further expanded in Lines 13 and 17. The trace, as mentioned above, induces a richer structure, reflecting rigid roles and the constant domain assumption; see Fig. 4 (center). This richer structure is then `completed` to properly enrich the types $Q(d, w)$ of each element. In particular, during completion, further concept names are added to the corresponding types (Fig. 4, right). One can now easily see that $\mathcal{T} \models A \sqsubseteq A_1$ indeed holds. Furthermore, note that $\mathcal{T} \not\models A \sqsubseteq A_1$, if r is local or increasing domains are assumed. This is the case since, in both cases, the r-connection is not necessarily present in the 'root world'.

For the termination condition in Line 7, we take the following definition of periodicity.

Definition 12. *A trace* (σ, E, R) *together with a mapping* Q *is called* periodic at (i, j) *if* $\sigma = (d_0, w_0) \cdots (d_n, w_n)$*,* $i < j$*,* $d_i = d_j = d_n$*, and* $Q(d_i, w_i) = Q(d_j, w_j)$*.*

This means that during the evolution of element $d = d_i = d_j$, we find two different worlds w_i, w_j such that d has the same type in w_i and w_j. We can stop expanding worlds appearing after w_j since their behavior is already captured by the successors of w_i. If a trace periodic at (i, j) is found, we add an edge (w_{j-1}, w_i) to E reflecting the periodic behavior, see Line 8. Then, in `truncate`, the trace is shortened to $(d_0, w_0) \cdots (d_{j-1}, w_{j-1})$ and the relations E and $R(r)$, $r \in \mathsf{ROL}$, and the mappings $Q, Q_{\mathsf{cert}}, Q_{\mathbf{A}\Box}$ are restricted to those d and w that appear in the shortened trace.

Lemma 13. *On every input* $\mathcal{T}, A, B$*, Alg. 1 terminates and returns* `true` *iff* $\mathcal{T} \models A \sqsubseteq B$*.*

For termination, consider a trace with suffix $(d, w_1) \cdots (d, w_n)$ and let $A_1, \ldots, A_n$ be the concept names such that $\mathbf{E}\Diamond A_i$ lead to w_i, see Line 17 of Alg. 1. It is not difficult to show that if $A_i = A_j$ for $i < j$, then $Q(d, w_i) \subseteq Q(d, w_j)$ after application of `complete`. Since $Q(d, w) \subseteq \mathsf{CN}$, there are no infinite (strictly) increasing sequences. Hence, the expansion in Lines 17ff. will not indefinitely be applied. Also, the expansion in Lines 13ff. stops due to acyclicity of the TBox. Together, this guarantees termination.

Correctness is shown similar to Lemma 7. For "⇒", we show that every trace together with the labeling so far computed in Q can be embedded into every model of A and $\mathcal{T}$. For "⇐", we present a model of $\mathcal{T}$ witnessing $\mathcal{T} \not\models A \sqsubseteq B$.

We finish the proof of Theorem 9 by noting that the termination argument indeed yields a polynomial bound on the length of the traces encountered by Alg. 1.

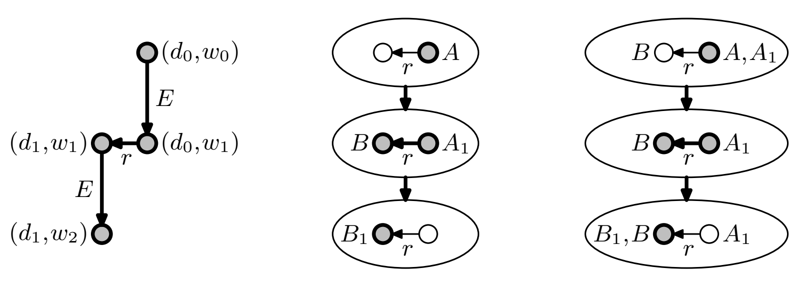

Fig. 4. An example trace and the induced structure

7 Local Roles and Rigid Concepts

One can easily extend the above algorithms so as to deal with local roles. In fact, e.g., in Section 5 only **B4** below needs to be added to the BACKWARD-rules in Figure 3. Note that **F2** is only applied to rigid roles and **C2** is therefore not applied to local ones. Clearly, the algorithm in Section 6 can be extended with a similar rule.

B4 If $A \in Q(B, w)$, $A \equiv \exists r.A'$, $B' \in Q(A', A'A')$, $B'' \equiv \exists r.B' \in \mathcal{T}$, add B'' to $Q(B, w)$
RC If $B \in Q(A, w)$, $B \in \mathsf{CN}_{\mathsf{rig}}$, then add B to $Q(A, w')$, $\forall w' \in W$
R13 If $B \in Q(d, w)$ or $B \in Q_{\mathbf{A}\Box}(d, w)$ & $B \in \mathsf{CN}_{\mathsf{rig}}$, then add B to $Q_{\mathsf{cert}}(d)$

A rigid concept has a constant interpretation over time. In the first example of Section 1, the concept Disorder should be rigid because we regard medical knowledge as static. PatientWithDisorder should be local because a disease history has begin and end.

In the presence of general TBoxes, rigid concepts can be simulated by rigid roles: replace each rigid concept name A with $\exists r_A.\top$, where r_A is a fresh rigid role. Alas, this simulation does not work in the context of acyclic TBoxes: the result of replacing A with $\exists r_A.\top$ in a CD $A \equiv D$ is no longer a CD. Still, our algorithms can be extended, without increasing the complexity, to consider rigid concepts: e.g., the algorithm in Section 5 can be extended by adding **RC** above to the FORWARD *and* BACKWARD rules – $\mathsf{CN}_{\mathsf{rig}}$ denotes the set of rigid concepts occurring in the input TBox. Note that the intermediate phase remains the same, i.e., rules **C1** and **C2** are neither extended nor modified.

Rigid concepts can analogously be included in Section 6 by adding a new rule **R13** above (recall: intuitively, $Q_{\mathsf{cert}}(d)$ contains the concepts that hold for d in any world).

In the empty TBox case rigid roles can again simulate rigid concepts as above.

8 Conclusions and Future Work

We have initiated the investigation of TDLs based on $\mathcal{EL}$ allowing for rigid roles and restricted TBoxes. We indeed achieved our main goal: we identified fragments of the combination of CTL and $\mathcal{EL}$ that have elementary, some even polynomial, complexity.

One important conclusion is that the use of acyclic TBoxes, instead of general ones, allows to design TDLs based on $\mathcal{EL}$ with dramatically better complexity than the $\mathcal{ALC}$ variant; e.g., for the fragment allowing only $\mathbf{E}\bigcirc$ the complexity drops from nonelementary to PTIME. As an important byproduct, the studied fragments of $\text{CTL}_{\mathcal{EL}}$ can be seen as *positive fragments* of product modal logics with elementary complexity, e.g., implication for the positive fragment of $\mathsf{K}\times\mathsf{K}$ is in PTIME.

Next, we plan to look at more expressive fragments of $\text{CTL}_{\mathcal{EL}}$ or at classical (cyclic) TBoxes, e.g., consider *non-convex* fragments, such as $\text{CTL}_{\mathcal{EL}}^{\mathbf{E}\bigcirc,\mathbf{E}\Diamond}$, with (a)cyclic TBoxes. We plan to incorporate temporal roles, too. It is also worth exploring how restricting TBoxes can help tame other TDLs with bad computational behavior over general TBoxes, such as TDLs based on LTL or the μ-calculus. We believe that the LTL case is technically easier than ours since it does not have the extra '$\frac{1}{2}$-dimension' introduced by branching.

Acknowledgements The first author was supported by the M8 PostDoc Initiative project TS-OBDA and the second one by the DFG project LU1417/1-1. We thank the anonymous reviewers for their detailed and constructive suggestions.

References

1. Artale, A., Kontchakov, R., Lutz, C., Wolter, F., Zakharyaschev, M.: Temporalising tractable description logics. In: Proc. TIME (2007)
2. Artale, A., Kontchakov, R., Ryzhikov, V., Zakharyaschev, M.: A cookbook for temporal conceptual data modelling with description logics. ACM Trans. Comput. Log. 15(3), 25 (2014)
3. Artale, A., Lutz, C., Toman, D.: A description logic of change. In: Proc. IJCAI (2007)
4. Baader, F.: Terminological cycles in a description logic with existential restrictions. In: Proc. IJCAI (2003)
5. Baader, F., Brandt, S., Lutz, C.: Pushing the $\mathcal{EL}$ envelope. In: Proc. IJCAI (2005)
6. Baader, F., Calvanese, D., McGuinness, D., Nardi, D., Patel-Schneider, P.F. (eds.): The Description Logic Handbook: Theory, Implementation, and Applications. Cambridge University Press (2003)
7. Baader, F., Ghilardi, S., Lutz, C.: LTL over description logic axioms. In: Proc. KR (2008)
8. Baader, F., Küsters, R., Molitor, R.: Computing least common subsumers in description logics with existential restrictions. In: Proc. IJCAI (1999)
9. Bodenreider, O., Zhang, S.: Comparing the representation of anatomy in the FMA and SNOMED CT. In: Proc. AMIA (2006)
10. Clarke, E.M., Grumberg, O., Peled, D.A.: Model Checking. MIT Press (1999)
11. Gabbay, D., Kurucz, A., Wolter, F., Zakharyaschev, M.: Many-dimensional modal logics: theory and applications, Studies in Logic, vol. 148. Elsevier (2003)
12. Göller, S., Jung, J.C., Lohrey, M.: The complexity of decomposing modal and first-order theories. ACM Trans. Comput. Log. 16(1), 9:1–9:43 (2015)
13. Gutiérrez-Basulto, V., Jung, J.C., Lutz, C.: Complexity of branching temporal description logics. In: Proc. ECAI (2012)
14. Gutiérrez-Basulto, V., Jung, J.C., Lutz, C., Schröder, L.: A closer look at the probabilistic description logic Prob-$\mathcal{EL}$. In: Proc. AAAI (2011)
15. Gutiérrez-Basulto, V., Jung, J.C., Schneider, T.: Lightweight description logics and branching time: a troublesome marriage. In: Proc. KR (2014)
16. Haase, C., Lutz, C.: Complexity of subsumption in the $\mathcal{EL}$ family of description logics: Acyclic and cyclic TBoxes. In: Proc. ECAI (2008)
17. Lutz, C., Wolter, F., Zakharyaschev, M.: Temporal description logics: A survey. In: Proc. TIME (2008)
18. Nebel, B.: Terminological reasoning is inherently intractable. Artif. Intell. 43(2), 235–249 (1990)
19. Schild, K.: Combining terminological logics with tense logic. In: Proc. EPIA (1993)
20. The Gene Ontology Consortium: Gene ontology: Tool for the unification of biology. Nature Genetics 25, 25–29 (2000)