Temporalising $\mathcal{EL}$ Concepts with Time Intervals

Jared Leo, Bijan Parsia, and Uli Sattler

The University of Manchester
Oxford Road, Manchester, M13 9PL, UK
{jared.leo,bijan.parsia,ulrike.sattler}@manchester.ac.uk

Abstract. We design and investigate a new interval based temporal description logic, $\mathcal{EL}_\lambda$, which is based on an Annotated Logic introduced by Kifer, AL, and motivated by life-science applications. We show how a subset of the logic can be captured as the $\mathcal{EL}$ fragment of AL, $\mathcal{EL}_{\mathsf{AL}}$, and then go on to show how we can extend this representation to capture further temporal entailments. We show that both $\mathcal{EL}_{\mathsf{AL}}$ and $\mathcal{EL}_\lambda$ maintain the same tractable complexity bounds for reasoning as $\mathcal{EL}$ and finally provide an example of how the logic can be utilised for the Drosophila developmental ontology.

1 Introduction

Description Logics (DLs) [4] are unable to capture simple temporal information and, as a step towards solving this problem, proposals for adding a temporal aspect to DLs have been considered. Some proposals [20, 2, 18] suggest the combination of DLs and Temporal Logics (TLs) [9, 11, 13, 19] to form Temporal Description Logics (TDLs), where a standard approach, first proposed by Schild [20], involves the combination of standard DLs, such as $\mathcal{ALC}$ [21] with standard TLs such as Linear Temporal Logic (LTL) [11, 13, 19] to form LTL$_{\mathcal{ALC}}$ [2, 18]. In this approach, temporal operators from LTL are added to $\mathcal{ALC}$ concept descriptions to build temporal statements. Other suggestions include using time points and intervals based on Allens 13 interval relations [1, 12], those developed for temporal query answering [7], or temporal data access [6].

Temporal aspects are showing up in numerous ontologies, particularly in life science and developmental ontologies. The OBO-relation ontology [22] is a widely used example of this. This ontology incorporates core upper-level relations such as *part of* as well as biology-specific relations such as *develops from*. Time information is present in all of the informal definitions of the relations. As an example, consider the definition of the relation *located in*:

> " C located_in C' if and only if: given any c that instantiates C at a time t, there is some c' such that: c' instantiates C' at time t and c located_in c' ". [22]

It is clear that temporal information is needed and wanted, but can not be captured due to limitations of current DLs. Temporal information is also present in many developmental ontologies in the bio-medical domain. These ontologies often express their time content in terms of *stages* (often referred to as Theiler stages) which are usually represented in a linear discrete time sequence. The temporal information of theses stages is usually stored as OWL annotations on classes and not included in the formal definition of the ontology. An example focussing on developmental stages is the Drosophila ontology [10] which describes developmental stages of the life cycle of a Drosophila (fruit) fly. Many of its classes are defined in terms of a chain of *develops_from* relations that indicates one stage coming after another, or, one class developing into another at some specific time point. A sequence of these stages is implicitly given through these chains. Again, it is clear that temporal information is present here, but more importantly, very specific time information is present where all time points are known.

It seems that the expressive powers of current TDLs are somewhat of an overkill for these applications that mention specific moments in time and stages. For example, the $\Diamond$ operator (some future moment) expresses time information where specific time points are not known (uncertainty). They also consider time to be relative and over an infinite sequence of time points, whereas the applications need more of an absolute time representation where we only need a few time points of a finite sequence.

In this paper we investigate and design two TDLs based on an interval representation of time using exact time points, with the aim to provide a suitable TDL for developmental ontologies. Our first attempt is based on an Annotated Logic (AL) introduced by Kifer [16] denoted as AL, where we combine the general semantics of AL, with $\mathcal{EL}$. We extend the semantics to encode annotation variables as time intervals to represent time. The second is a more general approach where we consider using a possible world semantics. With the two new logics, we hope to provide a more a succinct representation of time for stage based ontologies.

2 Preliminaries

$\mathcal{ELH}$ & $\mathcal{EL}$ We assume the reader to be familiar with the syntax and semantics of $\mathcal{ELH}$ and $\mathcal{EL}$ [3, 8]. We use an RBox to capture role inclusion axioms, instead of the traditional method of incorporating these axioms into the TBox. If $\mathcal{T}$ is a TBox, we use $\widetilde{\mathcal{T}}$ to be the signature of the TBox, i.e. the subset of N_{con} for which elements are directly used in $\mathcal{T}$, similarly for an ABox $\mathcal{A}$, an RBox $\mathcal{R}$ or a knowledge base $\mathcal{K}$.

First Order Logic We assume the reader to be familiar with first order logic, and will use the standard notion of a *valuation*, i.e., a mapping from free variables into interpretation domain elements and, for x a variable name, the x-variant of a valuation ν, i.e., the valuation ν' that behaves exactly like ν but for possibly mapping x to a different domain element.

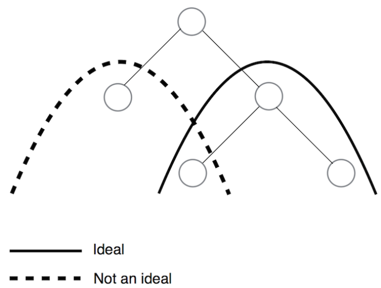

Fig. 1. An ideal over a semi lattice

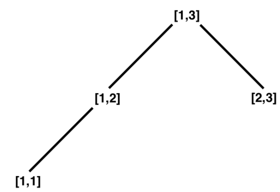

Fig. 2. A semilattice of intervals

3 Introducing $\mathcal{EL}_{\mathsf{AL}}$

3.1 Annotated Logics

Annotated Logics (ALs) are formalisms that have been applied to knowledge representations and expert systems, first introduced in [23] and later studied in [16, 15, 5]. ALs are also important to consider when incorporating time into DLs. Instead of combining temporal operators from TLs to concepts, ALs can 'annotate' parts of a formulae with time information, at an atomic level. Kifer in [16] introduced a general semantics for ALs (denoted as AL) based on ideals of semilattices, where annotations are elements of a semilattice with some ordering $\geq$ and the least upper bound operator $\sqcup$. Formulae of AL are simply first order logic formulae where atoms are annotated with elements of a given semilattice. W.l.o.g., we assume all atoms to be annotated with some lattice element. Next, we define the semantics of AL as described in [16], but without reference to Herbrand interpretations. Note that we assume an upper semilattice of annotations needs not be complete - we only assume the existence of a greatest element.

Definition 1. *An* ideal $\mathcal{ID}$ *of a semilattice* $\mathcal{SL}$ *is any subset* S *s.t.*

- $s \in S$ *and* $t \leq s \longrightarrow t \in S$ *and*
- $s, t \in S \longrightarrow s \sqcup t \in S$.

We use $\mathcal{ID}(\mathcal{SL})$ *for the set of all ideals of* $\mathcal{SL}$*. Furthermore, we assume that the lattice comes with a unary (pseudo) complement operator* $\neg$.

A generalized interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \mathcal{I}(\cdot), \mathcal{SL})$ *consists of a non-empty domain* $\Delta^{\mathcal{I}}$ *and a mapping* $\mathcal{I}(\cdot)$ *that maps (i) each constant symbol* a *to an element* $\mathcal{I}(a) \in \Delta^{\mathcal{I}}$ *and (ii) each* n*-ary predicate symbol* P *and each tuple* $a_1, \ldots, a_n \subseteq (\Delta^{\mathcal{I}})^n$ *to an ideal* $S \in \mathcal{ID}(\mathcal{SL})$*. The interpretation is extended to formulae as follows: let* $\mathcal{I}$ *be a generalized interpretation,* ν *a valuation,* $\alpha \in \mathcal{SL}$*,* A *a ground atom, and* F *a formula. Then:*

1. $\mathcal{I}, \nu \models A(t_1, \ldots, t_n) : \alpha$ *if* $\alpha \in \mathcal{I}(A(e_1, \ldots, e_n))$*, where* $e_i = \mathcal{I}(t_i)$ *if* t_i *is a constant and* $e_i = \nu(t_i)$ *if* t_i *is a variable,*

2. $\mathcal{I}, \nu \models \neg A(t_1, \ldots, t_n) : \alpha$ *if* $\neg(\alpha) \in \mathcal{I}(A(e_1, \ldots, e_n))$ *, where* $e_i = \mathcal{I}(t_i)$ *if* t_i *is a constant and* $e_i = \nu(t_i)$ *if* t_i *is a variable,*
3. $\mathcal{I}, \nu \models F^1 \wedge F^2$ *if* $\mathcal{I}, \nu \models F^1$ *and* $\mathcal{I}, \nu \models F^2$*,*
4. $\mathcal{I}, \nu \models \exists x.F$ *if* $\mathcal{I}, \nu' \models F(x)$ *for some x-variant* ν' *of* ν*,*
5. $\mathcal{I}, \nu \models \forall x.F$ *if* $\mathcal{I}, \nu' \models F(x)$ *for all x-variants* ν' *of* ν*.*

As usual, $\mathcal{I}$ is said to be a model of a formula F *if* $\mathcal{I} \models F$. To illustrate the semantics of AL when combined with a DL, let us first extend the semilattice to handle time intervals:

Definition 2. *An interval* λ *is of the form* $[x, y]$ *where* $x, y \in \mathbb{N}$ *and* $x \leq y$*. For an interval* $\lambda = [x, y]$*,* x *is the start point of* λ *and is denoted as* λ_s *and similarly* y *is the end point of* λ *and shall be denoted as* λ_e*. Let* $(\mathcal{SL}, \leq)$ *be a semi-lattice of intervals upwards closed by an interval denoted as* $\lambda^\top$*. Let* $\lambda^1, \lambda^2 \in \mathcal{SL}$*. The ordering* $\leq$ *is as follows:* $\lambda^1 < \lambda^2$ *iff* $\lambda^1_s \geq \lambda^2_s$ *and* $\lambda^1_e \leq \lambda^2_e$*. Let* $min(\mathcal{SL}) = \lambda_s$ *for* $\lambda \in \mathcal{SL}$ *s.t* $\nexists \lambda' \in \mathcal{SL}$ *and* $\lambda'_s < \lambda_s$*. Let* $max(\mathcal{SL}) = \lambda_e$ *for* $\lambda \in \mathcal{SL}$ *s.t* $\nexists \lambda' \in \mathcal{SL}$ *and* $\lambda'_e > \lambda_e$*.* $\lambda^\top = [\mathcal{SL}_{min}, \mathcal{SL}_{max}]$*.*

In this paper we focus on representing time as a discrete linear sequence, similar to the representation of time in LTL where we have exactly one future moment in time (discrete) and only one time sequence (linear). We do not state any assumption on the time between points x and $x + 1$. Consider the following AL formulae annotated with intervals (the semilattice for the annotations of the formulae is depicted in Figure 2):

1. $\forall x \left[B_{[1,1]}(x) \Rightarrow A_{[1,2]}(x)\right]$
2. $\forall x \left[\exists y \left(A_{[1,1]}(x) \Rightarrow R_{[1,3]}(x, y) \wedge C_{[1,2]}(y)\right)\right]$
3. $\forall x \left[\left(\exists y\, R_{[1,1]}(x, y) \wedge C_{[1,1]}(y)\right) \Rightarrow D_{[2,3]}(x)\right]$
4. $A_{[1,3]}(d)$

We interpret the formulae above as follows: (1) All instances of B at time 1 are instances of A at times 1 and 2, (2) All instances of A at time 1 have some R-relation at times 1, 2 and 3 to an individual who is an instance of C at times 1 and 2, (3) All instances who have an R successor at time 1 to some instance who is a C at time 1, are instances of D at times 2 and 3, (4) d is an instance of A at times 1, 2 and 3. To show how the lattice interacts with formulae and ideals, consider the following example. Using the lattice from Figure 2, and a formula $A_{[1,3]}(d)$, the ideal of $\mathcal{ID}([1, 3])$ is {[1,3], [1,2], [2,3], [1,1]}. Therefore we have, for any model $\mathcal{I}$ of a knowledge base, for any valuation ν, and for any $\lambda \in \mathcal{ID}([1, 3])$:

$$\mathcal{I}, \nu \models A_{[1,3]}(x) \Rightarrow A_\lambda(x) \tag{1}$$

We now combine the semantics of AL and $\mathcal{EL}$ to form $\mathcal{EL}_{\mathsf{AL}}$- our first attempt at temporalising $\mathcal{EL}$ with time intervals based on AL.

3.2 $\mathcal{EL}_{\mathsf{AL}}$- The $\mathcal{EL}$ fragment of AL

$\mathcal{EL}_{\mathsf{AL}}$ concept descriptions extend $\mathcal{EL}$ concept descriptions with the use of intervals (which we may refer to as labels) occurring in a semilattice $\mathcal{SL}$ and appearing on atomic concepts and roles. Concept descriptions in $\mathcal{EL}_{\mathsf{AL}}$ are defined according to the following definition:

Definition 3. *Let $\mathcal{SL}$ be a semilattice of intervals, $\lambda \in \mathcal{SL}$ an arbitrary interval, A an atomic concept, R an atomic role (role name) and C, D arbitrary concept descriptions. Then concept descriptions are formed in $\mathcal{EL}_{\mathsf{AL}}$ according to the following syntax rule:*

$$C, D \longrightarrow \top_{\lambda^\top} \mid A_\lambda \mid C \sqcap D \mid \exists R_\lambda.C$$

The semantics is given in terms of an interpretation:

Definition 4. *An interpretation $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$, consists of a non empty set $\Delta^{\mathcal{I}}$ which is the domain of $\mathcal{I}$, and a function $\cdot^{\mathcal{I}}$ that maps individuals to elements of $\Delta^{\mathcal{I}}$, each concept name $A_\lambda \subseteq \Delta^{\mathcal{I}}$ and each role name $R_\lambda \subseteq \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$. The function $\cdot^{\mathcal{I}}$ is inductively extended to arbitrary concepts by setting*

- $\top^{\mathcal{I}} = \Delta^{\mathcal{I}}$
- $(C \sqcap D)^{\mathcal{I}} = C^{\mathcal{I}} \cap D^{\mathcal{I}}$
- $(\exists R_{\lambda^1}.C)^{\mathcal{I}} = \{x \in \Delta^{\mathcal{I}} \mid \exists y \in C^{\mathcal{I}} \wedge (x, y) \in R^{\mathcal{I}}_{\lambda^1}\}$

as well as the following restrictions on the domain reflecting the general semantics from AL*:*

1. $A^{\mathcal{I}}_{\lambda^1} \subseteq A^{\mathcal{I}}_{\lambda^2} \longleftarrow \lambda^1 \geq \lambda^2$ *and* $A_{\lambda^1}, A_{\lambda^2} \in N_{con}$
2. $R^{\mathcal{I}}_{\lambda^1} \subseteq R^{\mathcal{I}}_{\lambda^2} \longleftarrow \lambda^1 \geq \lambda^2$ *and* $R_{\lambda^1}, R_{\lambda^2} \in N_{role}$

An $\mathcal{EL}_{\mathsf{AL}}$ knowledge base (KB) is made up of three parts: a terminological part called the TBox, denoted as $\mathcal{T}$, an assertional part called the ABox, denoted as $\mathcal{A}$ and a semilattice of time intervals $\mathcal{SL}$. The TBox and ABox are defined as usual. It is clear that the added restrictions on interpretations $\mathcal{I}$ of $\mathcal{EL}_{\mathsf{AL}}$ capture the semantics of AL because, the only additional implications we have are that when $\lambda^1 \leq \lambda^2$, $\forall x : A_{\lambda^2}(x) \rightarrow A_{\lambda^2}(x)$. We aim to show that subsumption in $\mathcal{EL}_{\mathsf{AL}}$ can be decided in polynomial time, and we do this by utilising an existing decision procedure for $\mathcal{ELH}$. We give a polynomial time reduction to show that any $\mathcal{EL}_{\mathsf{AL}}$ KB can be converted into a classical $\mathcal{ELH}$ KB s.t. they are model preserving. Our reduction consists of unfolding the restrictions (1 & 2) into the KB, enabling us to remove the lattice and have an equisatisfiable $\mathcal{ELH}$ KB.

3.3 From $\mathcal{EL}_{\mathsf{AL}}$ to $\mathcal{ELH}$

Reduction We unfold the restrictions on the domain of $\mathcal{EL}_{\mathsf{AL}}$ into an $\mathcal{ELH}$ KB, similar to the reduction found in [6], according to the following definition:

Definition 5. *Let K = ($\mathcal{T}$,$\mathcal{A}$,$\mathcal{SL}$) be an $\mathcal{EL}_{\mathsf{AL}}$ KB over the set N_{con} and N_{role}. The $\mathcal{ELH}$ KB $\mathsf{R}(K) = (\mathcal{T}', \mathcal{A}, \mathcal{R})$ is defined as follows:*

$$\mathcal{T}' := \mathcal{T} \cup \left\{ A_{\lambda^1} \sqsubseteq A_{\lambda^2} \mid A_{\lambda^1}, A_{\lambda^2} \in \widetilde{\mathcal{T}} \wedge \lambda^1 \geq \lambda^2 \right\}$$
$$\mathcal{R} := \left\{ R_{\lambda^1} \sqsubseteq R_{\lambda^2} \mid R_{\lambda^1}, R_{\lambda^2} \in \widetilde{\mathcal{T}} \wedge \lambda^1 \geq \lambda^2 \right\}$$

Since $\mathsf{R}(K)$ involves no semilattice, we treat labelled concept names as usual $\mathcal{ELH}$ concept names. It is clear that for any model $\mathcal{I}$ of K, $\mathcal{I}$ is also a model of $\mathsf{R}(K)$, as the only axioms added to K are those already satisfied in $\mathcal{I}$. It is also clear that for any model $\mathcal{I}$ of $\mathsf{R}(K)$, $\mathcal{I}$ is also a model of K since the added axioms in $\mathsf{R}(K)$ ensure that the additional constraints of $\mathcal{I}$ have been met.

Lemma 1. *Let K=($\mathcal{T}$,$\mathcal{A}$,$\mathcal{SL}$) be an $\mathcal{EL}_{\mathsf{AL}}$ KB. K and $\mathsf{R}(K)$ have the same models.*

The maximum number of additional axioms in $\mathsf{R}(K)$ is limited quadratically w.r.t $|K|$. The reduction can be computed deterministically and since there are only a quadratic number of additional axioms that can be added, the reduction only requires polynomial time.

Lemma 2. *Let K= ($\mathcal{T}$,$\mathcal{A}$,$\mathcal{SL}$) be an $\mathcal{EL}_{\mathsf{AL}}$ KB. $\mathsf{R}(K)$ can be computed in polynomial time w.r.t $|\widetilde{K}|$.*

Finally, as we are now left with an $\mathcal{ELH}$ KB, we can utilise the reasoning procedures from $\mathcal{ELH}$ to compute subsumption. It was shown in [8] that subsumption in $\mathcal{ELH}$ can be decided in polynomial time. Therefore, since we have a polynomial reduction from $\mathcal{EL}_{\mathsf{AL}}$ into $\mathcal{ELH}$, we can compute subsumption in $\mathcal{EL}_{\mathsf{AL}}$ in polynomial time.

Corollary 1. *Computing subsumption in $\mathcal{EL}_{\mathsf{AL}}$ is PTime-Complete.*

Limitations of $\mathcal{EL}_{\mathsf{AL}}$ According to the semantics of the semilattice from AL, the lattice really is just a set of labels, and even if we consider those labels to be intervals, the only temporal information they carry is that an interval is contained within another. Crucially, they do not capture an important aspect of temporal information involving overlapping intervals. Consider a semilattice of intervals with elements $\{[1,4],[2,5],[3,6],[1,7]\}$, and a TBox $\mathcal{T} := \{C_{[1,7]} \sqsubseteq A_{[1,4]} \sqcap A_{[3,6]}, A_{[2,5]} \sqsubseteq D_{[1,7]}\}$. Since the interval $[2,5]$ is not contained within the intervals $[1,4]$ or $[3,6]$, there is nothing to tell us that the interval is contained within the conjunction of the two, i.e we want $\mathcal{T}$ to entail that $C_{[1,7]} \sqsubseteq D_{[1,7]}$. It is not obvious how to recode the lattice to capture this. We could fix the issue by extending the domain constraints, however it is clear that the semantics of AL is not enough to capture minimal temporal information using our current ordering. Other issues arise when we consider the possibility of varying domains and possible worlds. In AL we are restricted to a single "world" with a constant domain, but TDLs can adopt a possible world semantics allowing the possibility of varying domains. It is not clear how to extend $\mathcal{EL}_{\mathsf{AL}}$ to capture this. To overcome these 2 problems, we move towards a new version of an interval based $\mathcal{EL}$ with a possible worlds semantics.

4 $\mathcal{EL}_{\lambda}$- A Possible World Semantics

We now consider temporalising $\mathcal{EL}$ with time intervals by adopting a possible worlds semantics [24]. We use the same definition for intervals as in Section 3.

4.1 Syntax and Semantics of $\mathcal{EL}_\lambda$

$\mathcal{EL}_\lambda$ concept descriptions are built in the same way as $\mathcal{EL}_{\mathsf{AL}}$ concept descriptions, however the intervals need no longer appear in a semilattice. In $\mathcal{EL}_\lambda$, the semantics of concept descriptions is defined in terms of a temporal interpretation with possible worlds. The possible worlds are a finite sequence of normal $\mathcal{EL}$ interpretations indexed by $i \in \mathbb{N}$.

Definition 6. *An $\mathcal{EL}_\lambda$ interpretation $\mathcal{I} = (\Delta^\mathcal{I}, \cdot^{\mathcal{I}_i}, \cdot^\mathcal{I})$ consists of a non-empty (constant) domain $\Delta^\mathcal{I}$ and a function $\cdot^{\mathcal{I}_i}$ that, for each index $i \in \{m, ..., n\} \subseteq \mathbb{N}$, maps each concept name A to a subset $A^{\mathcal{I}_i} \subseteq \Delta^\mathcal{I}$, each role name R to a subset $R^{\mathcal{I}_i} \subseteq \Delta^\mathcal{I} \times \Delta^\mathcal{I}$, where i is an index for each possible world. The function $\cdot^{\mathcal{I}_i}$ is inductively extended to arbitrary concepts by setting*

- $\top^{\mathcal{I}_i} = \Delta^\mathcal{I}$
- $(C \sqcap D)^{\mathcal{I}_i} := C^{\mathcal{I}_i} \cap D^{\mathcal{I}_i}$
- $(\exists R.C)^{\mathcal{I}_i} := \{e \in \Delta^\mathcal{I} \mid \exists f \in C^{\mathcal{I}_i} \wedge (e, f) \in R^{\mathcal{I}_i}\}$

*$\cdot^\mathcal{I}$ is a **global** function which maps each labelled concept name $A_{[x,y]}$ to $A^\mathcal{I}_{[x,y]} = \bigcap_{x \leq i \leq y} A^{\mathcal{I}_i}$, and each role $R_{[x,y]}$ to $R^\mathcal{I}_{[x,y]} = \bigcap_{x \leq i \leq y} R^{\mathcal{I}_i}$. The function $\cdot^\mathcal{I}$ is inductively extended to arbitrary concepts with labelled elements as follows:*

- $\top^\mathcal{I} = \Delta^\mathcal{I}$
- $(C \sqcap D)^\mathcal{I} := C^\mathcal{I} \cap D^\mathcal{I}$
- $(\exists R_\lambda.C)^\mathcal{I} := \{e \in \Delta^\mathcal{I} \mid \exists f \in C^\mathcal{I} \wedge (e, f) \in R^\mathcal{I}_\lambda\}$

Since we now have two types of concepts: A_λ and A (similarly with role names), we differentiate between the two by declaring N_{con} as the unlabelled counter parts of concepts occurring in N^λ_{con} (similarly with N_{role} and N^λ_{role}).

Lemma 3. *For any $\mathcal{I}$, (1) $A^\mathcal{I}_{[x,y]} \subseteq A^\mathcal{I}_{[u,v]}$ if $x \leq u \leq v \leq y$. (2) $A^\mathcal{I}_{[x,y]} \cap A^\mathcal{I}_{[u,v]} \subseteq A^\mathcal{I}_{[m,n]}$ if $x < u \leq y + 1 \leq v$, $m \geq x$, $n \leq v$. (Similarly, for each role name R_λ).*

Lemma 3.1 establishes that we capture the same constraints on the domain as $\mathcal{EL}_{\mathsf{AL}}$, whilst Lemma 3.2 overcomes the known limitations of $\mathcal{EL}_{\mathsf{AL}}$ by capturing the overlapping intervals correctly.

An $\mathcal{EL}_\lambda$ KB is built in the same way as an $\mathcal{EL}_{\mathsf{AL}}$ KB without the need of a semilattice.

4.2 Decision Procedure

As usual, we rely on an $\mathcal{EL}_\lambda$ TBox to be in normal form before computing subsumption.

Definition 7. *Let $\mathcal{T}$ be an $\mathcal{EL}_\lambda$-TBox over the set N^λ_{con} and N^λ_{role}. $\mathcal{T}$ is normalised iff $\mathcal{T}$ contains only GCIs of the forms:*

$$\begin{aligned} A_{\lambda^1} &\sqsubseteq B_{\lambda^2} \\ A_{\lambda^1} \sqcap B_{\lambda^2} &\sqsubseteq C_{\lambda^3} \\ A_{\lambda^1} &\sqsubseteq \exists R_{\lambda^2}.B_{\lambda^3} \\ \exists R_{\lambda^1}.A_{\lambda^2} &\sqsubseteq B_{\lambda^3} \end{aligned}$$

where $A_{\lambda^i}, B_{\lambda^i}, C_{\lambda^i}$ are atomic concept names from N^{λ}_{con} and R_{λ^i} is a role name from N^{λ}_{role}.

This normal form is similar to those seen in [3, 8]. We use a similar approach shown in [8] to transform any $\mathcal{EL}_\lambda$ TBox into a normalised version in polynomial time. Due to space constraints, the transformation proof can be found in [17].

Subsumer Sets We now show how we can compute subsumption relations for any labelled concept name occurring in an $\mathcal{EL}_\lambda$-TBox. Using a similar approach to [8], we compute this by building a set $S_*(A_\lambda)$ for every $A_\lambda \in N^{\lambda}_{con}$, for which each set contains labelled concept names that are subsumers of A_λ (similarly for each $R_\lambda \in N^{\lambda}_{role}$). The subsumer sets are defined as follows:

Definition 8. *Let $\mathcal{T}$ be an $\mathcal{EL}_\lambda$-TBox (normalized) over the set N^{λ}_{con} and N^{λ}_{role}. For every $A_\lambda \in N^{\lambda}_{con}$, and every $i \in \mathbb{N}$, the subsumer set $S_i(A_\lambda)$ is defined inductively by first applying rules* INIT0 *and* INIT1*, then for every $i \geq 0$, $S_{i+1}(A_\lambda)$ is defined by extending $S_i(A_\lambda)$ by exhaustive application of rules* CR0-CR5*. The subsumer set $S_*(A_\lambda)$, defined as the union of $\bigcup_{i \geq 0} S_i(A_\lambda)$. $S_{i+1}(A_\lambda)$ is complete if no more rules are applicable for any subsumer set (similarly for each $R_\lambda \in N_{role}$).*

- INIT0 $S_0(A_\lambda) := \{\top, A_\lambda\}$ for every $A_\lambda \in N^{\lambda}_{con}$,
- INIT1 $S_0(R_\lambda) := \{R_\lambda\}$ for every $R_\lambda \in N^{\lambda}_{role}$
- CR0 If $A_{[x,y]} \in S_i(C_{\lambda^1})$ and $D_{\lambda^2} \in S_i(A_{[u,v]})$ where $x \leq u \leq v \leq y$ and $D_{\lambda^2} \notin S_i(C_{\lambda^1})$ then $S_{i+1}(C_{\lambda^1}) := S_i(C_{\lambda^1}) \cup \{D_{\lambda^2}\}$
- CR1 If $A_{\lambda^1} \sqsubseteq B_{\lambda^2} \in \mathcal{T}$ and $A_{\lambda^1} \in S_i(C_{\lambda^3})$ and $B_{\lambda^2} \notin S_i(C_{\lambda^3})$ then $S_{i+1}(C_{\lambda^3}) := S_i(C_{\lambda^3}) \cup \{B_{\lambda^2}\}$.
- CR2 If $A_{\lambda^1} \sqcap B_{\lambda^2} \sqsubseteq C_{\lambda^3} \in \mathcal{T}$ and $A_{\lambda^1}, B_{\lambda^2} \in S_i(D_{\lambda^4})$ and $C_{\lambda^3} \notin S_i(D_{\lambda^4})$ then $S_{i+1}(D_{\lambda^4}) := S_i(D_{\lambda^4}) \cup \{C_{\lambda^3}\}$.
- CR3 If $A_{\lambda^1} \in S_i(B_{\lambda^2})$ and $A_{\lambda^1} \sqsubseteq \exists R_{\lambda^3}.C_{\lambda^4} \in \mathcal{T}$ and $D_{\lambda^5} \in S_i(C_{\lambda^4})$ and $P_{\lambda^6} \in S_i(R_{\lambda^3})$ and $\exists P_{\lambda^6}.D_{\lambda^5} \sqsubseteq E_{\lambda^7} \in \mathcal{T}$ and $E_{\lambda^7} \notin S_i(B_{\lambda^2})$ then $S_{i+1}(B_{\lambda^2}) := S_i(B_{\lambda^2}) \cup \{E_{\lambda^7}\}$.
- CR4 If $B_{\lambda^1} \in S_i(C_{[x,y]})$ and $C_{[i,j]} \in S_i(A_{\lambda^2})$ and $C_{[k,l]} \in S_i(A_{\lambda^2})$ and $i < k \leq j + 1 \leq l$ and $x \geq i$ and $y \leq l$ and $B_{\lambda^1} \notin S_i(A_{\lambda^2})$ then $S_{i+1}(A_{\lambda^2}) := S_i(A_{\lambda^2}) \cup \{B_{\lambda^1}\}$
- CR5 If $R_{[x,y]}, R_{[u,v]} \in N^{\mathcal{T}}_{role}$ and $x \leq u \leq v \leq y$ and $R_{[u,v]} \notin S_i(R_{[x,y]})$ then $S_{i+1}(R_{[x,y]}) := S_i(R_{[x,y]}) \cup \{R_{[u,v]}\}$

Theorem 1. *Let $\mathcal{T}$ be an $\mathcal{EL}_\lambda$-TBox over the set N^{λ}_{con} and N^{λ}_{role}. For every $F_{\lambda^1}, G_{\lambda^2} \in N^{\lambda}_{con}$ (or N^{λ}_{role}), it holds that $G_{\lambda^2} \in S_*(F_{\lambda^1})$ iff $\mathcal{T} \models F_{\lambda^1} \sqsubseteq G_{\lambda^2}$.*

Proof We first show the $\Rightarrow$ direction by proof of induction over n. Due to space constraints, we only include a proof for the most interesting rules. The full proof can be found in [17].

Claim: $G_{\lambda^2} \in S_*(F_{\lambda^1}) \Rightarrow \mathcal{T} \models F_{\lambda^1} \sqsubseteq G_{\lambda^2}$

$\mathbf{n = 0}$: If INIT0 added G_{λ^2} to $S_n(F_{\lambda^1})$, then $G_{\lambda^2}{=}F_{\lambda^1}$ or $G_{\lambda^2}{=}\top$, proving the claim holds.

n > 0: If CR4 added G_{λ^2} to $S_n(F_{\lambda^1})$ then there exists a concept $C_{[x,y]}$ where $G_{\lambda^2} \in S_{n-1}(C_{[x,y]})$ and $C_{[i,j]} \in S_{n-1}(F_{\lambda^1})$ and $C_{[k,l]} \in S_{n-1}(F_{\lambda^1})$ and $i < k \leq j+1 \leq l$ and $x \geq i$ and $y \leq l$ and $G_{[u,v]} \notin S_{n-1}(F_{\lambda^1})$. By induction hypothesis it holds that $\mathcal{T} \models C_{[x,y]} \sqsubseteq G_{\lambda^2}$, $\mathcal{T} \models F_{\lambda^1} \sqsubseteq C_{[i,j]}$ and $\mathcal{T} \models F_{\lambda^1} \sqsubseteq C_{[k,l]}$. From Lemma 1 and Lemma 2, $\mathcal{T} \models F_{\lambda^1} \sqsubseteq C_{[x,y]}$, therefore by transitivity of subsumption it holds that $\mathcal{T} \models F_{\lambda^1} \sqsubseteq G_{\lambda^2}$ proving the claim.

It suffices to show $\mathcal{T} \models F_{\lambda^1} \sqsubseteq G_{\lambda^2} \Rightarrow G_{\lambda^2} \in S_*(F_{\lambda^1})$. We approach this by proving the contraposition: **Claim**: $G_{\lambda^2} \notin S_*(F_{\lambda^1}) \Rightarrow \mathcal{T} \not\models F_{\lambda^1} \sqsubseteq G_{\lambda^2}$ where we build a canonical model $\mathcal{I}$ of $\mathcal{T}$ with a witness $x \in F^{\mathcal{I}}_{\lambda^1} \setminus G^{\mathcal{I}}_{\lambda^2}$. We construct the canonical model $\mathcal{I}$ according to the following definition:

- CM0 $\Delta^{\mathcal{I}} = \{a_{\lambda^1} \mid A_{\lambda^1} \in N^{\lambda}_{con}\}$
- CM1 $A^{\mathcal{I}_i} = \{b_{\lambda^1} \mid x \leq i \leq y \text{ and } A_{[x,y]} \in S_*(B_{\lambda^1})\}$
- CM2 $R^{\mathcal{I}_i} = \{(a_{\lambda^3}, b_{\lambda^4}) \mid x \leq i \leq y \text{ and } C_{\lambda^1} \sqsubseteq \exists P_{\lambda^2}.B_{\lambda^4} \text{ and } C_{\lambda^1} \in S_*(A_{\lambda^3})$ and $R_{[x,y]} \in S_*(P_{\lambda^2})\}$

We first show that $\mathcal{I}$ is in fact a valid model of $\mathcal{T}$. Since $\mathcal{T}$ is normalised, it suffices to show that the model is valid for each of the four possible axioms in $\mathcal{T}$. Again, due to space constraints, we only include a proof for the most interesting axioms. The full proof can be found in [17].

- $A_{\lambda^1} \sqsubseteq B_{\lambda^2} \in \mathcal{T} \longrightarrow A^{\mathcal{I}}_{\lambda^1} \subseteq B^{\mathcal{I}}_{\lambda^2}$. Let $y_{\lambda^3} \in A^{\mathcal{I}}_{\lambda^1}$. By definition of CM1 $A_{\lambda^1} \in S_*(Y_{\lambda^3})$. By non-applicability of CR1 $B_{\lambda^2} \in S_*(Y_{\lambda^3})$, thus by definition of CM1 $y_{\lambda^3} \in B^{\mathcal{I}}_{\lambda^2}$.

As we have shown that $\mathcal{I}$ is a model of $\mathcal{T}$, it remains to show that $F^{\mathcal{I}}_{\lambda^1} \not\subseteq G^{\mathcal{I}}_{\lambda^2}$ when $G_{\lambda^2} \notin S_*(F_{\lambda^1})$ by finding a witness $y_{\lambda^3} \in F^{\mathcal{I}}_{\lambda^1} \setminus G^{\mathcal{I}}_{\lambda^2}$. Claim: $G_{\lambda^2} \notin S_*(F_{\lambda^1}) \Rightarrow F^{\mathcal{I}}_{\lambda^1} \not\subseteq G^{\mathcal{I}}_{\lambda^2}$. By definition of the canonical model, CM1 and CM2 are the only definitions that add witnesses to concept interpretations. We prove the claim for the first and include the second in [17]:

- CM1: Let $[x,y] = \lambda^2$. CM1 will only add an individual f_{λ^1} to $\bigcap_{x \leq i \leq y} G^{\mathcal{I}_i}$ if $G_{[x,y]} \in S_*(F_{\lambda^1})$. Since $G_{[x,y]} \notin S_*(F_{\lambda^1})$ the individual is not added.

Theorem 2. *Let $\mathcal{T}$ be an $\mathcal{EL}_\lambda$-TBox over the set N^{λ}_{con} and N^{λ}_{role}. For every $A_\lambda \in N^{\mathcal{T},\top}_{con}$, the subsumer sets $S_*(A_\lambda)$ can be computed in polynomial time.*

Proof Let $n = |\mathcal{T}|$ (#GCIs), $m = |N^{\lambda}_{con}|$ and i represent each iteration of the subsumer sets s.t $S_*(A_\lambda) = S_i(A_\lambda)$ once $S_i(A_\lambda) = S_{i-1}(A_\lambda)$ for each $A_\lambda \in N^{\lambda}_{con}$. The initialisation phase where $i = 0$ takes m steps to compute - adding $\top$ and A_λ to each set $S_0(A_\lambda)$. For $i > 0$, the sets $S_i(A)$ depend only on $S_{i-1}(A_\lambda)$ and GCIs in $\mathcal{T}$. Since $|S_i(A_\lambda)| \leq m$, there can be at most $|S_i(A_\lambda)| - m - 1$ rule applications that can be fired out of the n possible GCIs in $\mathcal{T}$. i is also bounded by m since there can be no more than m iterations, otherwise more than m concepts would have been added to a subsumer set. Computing $S_*(A_\lambda)$ for every $A_\lambda \in N^{\lambda}_{con}$ takes polynomial time w.r.t $|\mathcal{T}|$ and $|N^{\lambda}_{con}|$(similarly for each $R_\lambda \in N^{\lambda}_{role}$).

Although we have the same syntax as $\mathcal{EL}_{\mathsf{AL}}$, semantically we capture more information - specifically overlapping intervals - and keep the polynomial time bound without any further extension. There is also room for considering extensions with varying domains, since we adopt a possible world semantics.

5 Drosophila Development Ontology

The Drosophila Development ontology [10] describes developmental stages of the life cycle of the Drosophila (fruit) fly. The ontology shows temporal patterns through one of its most used properties *devlopsFrom*. If we interpret this property as a stage based relation it is our understanding that an axiom of the form $A \sqsubseteq \exists devlopsFrom.B$ should be interpreted as any instance of B at a time point (*stage*), develops into an A at the next time point. Notice that temporal information is completely qualitative, it only tells us about stages and give no information on how long each stage lasts or even a relative duration. There is also another property called *partOf* which we can also interpret a temporal pattern: if we have the axioms $A \sqsubseteq \exists partOf.B$ and $A \sqsubseteq \exists partOf.C$, then the existence of A depends on the time points at which both B and C exist. We now take a small subset of the ontology to demonstrate the expressivity of $\mathcal{EL}_\lambda$. We extract the subset by starting from the class *coalescence_ spermatid* and include classes that are connected via a *devlopsFrom* chain. We also include a class connected by the *partOf* property. The resulting ontology is shown below:

$$\begin{aligned}
agglomeration_spermatid &\sqsubseteq \exists.devlopsFrom.coalescence_spermatid\\
clew_spermatid &\sqsubseteq \exists.devlopsFrom.agglomeration_spermatid\\
onion_spermatid &\sqsubseteq \exists.devlopsFrom.clew_spermatid\\
leafblade_spermatid &\sqsubseteq \exists.devlopsFrom.onion_spermatid\\
leafblade_spermatid &\sqsubseteq spermatid\\
onion_spermatid &\sqsubseteq spermatid\\
clew_spermatid &\sqsubseteq spermatid\\
agglomeration_spermatid &\sqsubseteq spermatid\\
coalescence_spermatid &\sqsubseteq spermatid\\
spermatid &\sqsubseteq \exists.partOf.spermatocyte_cyst
\end{aligned}$$

We temporalise the ontology as follows. Considering *coalescence_ spermatid* is at the beginning of the *devlopsFrom* chain and nothing is contained within this class and it has no *partOf* relationships, we label this concept with the the interval $[0, 0]$, i.e a single stage. The next concept we temporalise is *agglomeration_ spermatid*. Since the ontology contains the axiom $agglom\_spermatid \sqsubseteq \exists.devlopsFrom.coalescence\_spermatid$, using similar reasoning to the labelling of *coalescence_ spermatid* we label *agglomeration_ spermatid* with the interval $[1, 1]$. We then replace the first axiom with $coalescence\_spermatid_{[0,0]} \sqsubseteq agglomeration\_spermatid_{[1,1]}$. We repeat this process for each axiom of the form $A \sqsubseteq \exists devlopsFrom.B$. We give the class *spermatid* the interval [0,4] since it must exist during the duration of its subclasses that range from 0 to 4. We give the *partOf* relation in the axiom $spermatid \sqsubseteq \exists.partOf.spermatocyte\_cyst$ the interval $[0, 4]$ since *spermatid* has the interval $[0, 4]$ and therefore its parts must exist at the same time. Since *spermatocyte_ cyst* does not occur in any other axioms we label this with the same interval. We are left with the ontology:

$$
\begin{aligned}
coalescence\_spermatid_{[0,0]} &\sqsubseteq agglomeration\_spermatid_{[1,1]} \\
agglomeration\_spermatid_{[1,1]} &\sqsubseteq clew\_spermatid_{[2,2]} \\
clew\_spermatid_{[2,2]} &\sqsubseteq onion\_spermatid_{[3,3]} \\
onion\_spermatid_{[3,3]} &\sqsubseteq leafblade\_spermatid_{[4,4]} \\
leafblade\_spermatid_{[4,4]} &\sqsubseteq spermatid_{[0,4]} \\
spermatid_{[0,4]} &\sqsubseteq \exists .partOf_{[0,4]}.spermatocyte\_cyst_{[0,4]}
\end{aligned}
$$

We achieve a more succinct and faithful representation by mapping the ontology to an exact time sequence using this temporalisation, and we offer a means to some form of identity across the possible worlds.

6 Summary and Outlook

We presented a new approach for an interval based temporalisation of DLs, focussing on the lightweight DL $\mathcal{EL}$, based on an AL introduced by Kifer [16] denoted as AL. Our first attempt, $\mathcal{EL}_{\mathsf{AL}}$, saw the combination of $\mathcal{EL}$ and AL to form $\mathcal{EL}_{\mathsf{AL}}$ which was based on a classical DL model. Our second approach, $\mathcal{EL}_\lambda$, extended this using a possible world semantics and overcame some shortcomings of the first. We proved that both logics maintained the same polynomial time complexity bound for reasoning (subsumption) as $\mathcal{EL}$. We then showed how $\mathcal{EL}_\lambda$ can be used in a life-science oriented example by taking the Drosophila Development Ontology [10] and converting it into a temporal version.

We will identify whether varying domains are needed in practise, and if so, we will investigate the effects of allowing varying domains in $\mathcal{EL}_\lambda$ and see if we still maintain the polynomial time bound. We also plan to introduce variables in intervals to indicate some level of uncertainty and to see how closely related, if at all, this would be when compared with LTL combinations of DLs, namely $\mathrm{LTL}_{\mathcal{EL}}$. We plan to communicate with the authors of the Drosophila Development Ontology [10] to discuss our work, and get feedback on the usefulness of the logic. We have also currently implemented a preliminary reasoner for $\mathcal{EL}_\lambda$, using the OWL API [14], which can be used with OWL 2 where intervals appear as OWL-annotations on class and role names. This will enable us to view the performance of reasoning once an example has been fully encoded. Finally, we hope to extend the possible world semantics to more expressive DLs such as $\mathcal{EL}^{++}$ [3] and $\mathcal{ALC}$ [21].

References

1. Allen, J.F.: Maintaining knowledge about temporal intervals. Commun. ACM 26(11), 832–843 (Nov 1983), http://doi.acm.org/10.1145/182.358434
2. Artale, A., Franconi, E.: A survey of temporal extensions of description logics. Ann. Math. Artif. Intell. 30(1-4), 171–210 (2000)
3. Baader, F., Brand, S., Lutz, C.: Pushing the el envelope. In: Proc. of IJCAI 2005. pp. 364–369. Morgan-Kaufmann Publishers (2005)
4. Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D., Patel-Schneider, P.F. (eds.): The Description Logic Handbook: Theory, Implementation, and Applications. Cambridge University Press (2003)
5. Baader, F., Knechtel, M., Peñaloza, R.: Context-dependent views to axioms and consequences of semantic web ontologies. J. Web Sem. 12, 22–40 (2012)
6. Bobillo, F., Straccia, U.: Reasoning with the finitely many-valued lukasiewicz fuzzy description logic sroiq. Inf. Sci. 181(4), 758–778 (2011)
7. Borgwardt, S., Lippmann, M., Thost, V.: Temporal query answering in dl-lite. In: Eiter, T., Glimm, B., Kazakov, Y., Krötzsch, M. (eds.) Description Logics. CEUR Workshop Proceedings, vol. 1014, pp. 80–92. CEUR-WS.org (2013)
8. Brandt, S.: Reasoning in elh w.r.t. general concept inclusion axioms. Tech. rep. (2004)
9. Clarke, E.M., Emerson, E.A., Sistla, A.P.: Automatic verification of finite-state concurrent systems using temporal logic specifications. ACM Trans. Program. Lang. Syst. 8(2), 244–263 (Apr 1986), http://doi.acm.org/10.1145/5397.5399
10. Costa, M., Reeve, S., Grumbling, G., Osumi-Sutherland, D.: The drosophila anatomy ontology. Journal of Biomedical Semantics 4(1), 32 (2013), http://www.jbiomedsem.com/content/4/1/32
11. Fisher, M.: An Introduction to Practical Formal Methods Using Temporal Logic. Wiley (2011), http://books.google.co.uk/books?id=zl6OLZv7d1kC
12. Frasincar, F., Milea, V., Kaymak, U.: towl: Integrating time in owl. In: Virgilio, R.D., Giunchiglia, F., Tanca, L. (eds.) Semantic Web Information Management, pp. 225–246. Springer
13. Gabbay, D., Hodkinson, I., Reynolds, M.: Temporal Logic: Mathematical Foundations and Computational Aspects. No. v. 1 in Oxford logic guides, Clarendon Press (1994), http://books.google.co.uk/books?id=oVEZAQAAIAAJ
14. Horridge, M., Bechhofer, S.: The owl api: A java api for owl ontologies. Semantic Web 2(1), 11–21 (2011)
15. Kifer, M., Lozinskii, E.L.: Ri: A logic for reasoning with inconsistency. In: LICS. pp. 253–262. IEEE Computer Society (1989)
16. Kifer, M., Subrahmanian, V.S.: Theory of generalized annotated logic programming and its applications. J. Log. Program. 12(3&4), 335–367 (1992)
17. Leo, J., Parsia, B., Sattler, U.: Temporalising el concepts with time intervals - technical report. Tech. Rep. 1, Department of Computer Science, The University of Manchester, Oxford Road, Manchester, M13 9PL, UK (April 2014), http://www.cs.man.ac.uk.~leoj/techreportELLambda.pdf
18. Lutz, C., Wolter, F., Zakharyaschev, M.: Temporal description logics: A survey. In: Demri, S., Jensen, C.S. (eds.) TIME. pp. 3–14. IEEE Computer Society (2008)
19. Pnueli, A.: The temporal logic of programs. In: FOCS. pp. 46–57. IEEE Computer Society (1977)
20. Schild, K.: Combining terminological logics with tense logic. In: Filgueiras, M., Damas, L. (eds.) EPIA. Lecture Notes in Computer Science, vol. 727, pp. 105–120. Springer (1993)

21. Schmidt-Schauß, M., Smolka, G.: Attributive concept descriptions with complements. Artif. Intell. 48(1), 1–26 (1991)
22. Smith, B., Ashburner, M., Rosse, C., Bard, J., Bug, W., Ceusters, W., Goldberg, L.J., Eilbeck, K., Ireland, A., Mungall, C.J., OBI Consortium, Leontis, N., Rocca-Serra, P., Ruttenberg, A., Sansone, S.A.A., Scheuermann, R.H., Shah, N., Whetzel, P.L., Lewis, S.: The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. Nature biotechnology 25(11), 1251–1255 (Nov 2007), `http://dx.doi.org/10.1038/nbt1346`
23. Subrahmanian, V.S.: On the semantics of quantitative logic programs. In: SLP. pp. 173–182 (1987)
24. Wolter, F., Zakharyaschev, M.: On the decidability of description logics with modal operators. In: KR. pp. 512–523 (1998)