Chapter in *Handbook of Time and Temporal Reasoning in Artificial Intelligence*, edited by Dov Gabbay, Michael Fisher and Lluis Vila, MIT Press, (forthcoming).

Temporal Description Logics

Alessandro Artale
Dept. of Computation, UMIST, UK
artale@co.umist.ac.uk

Enrico Franconi
Dept. of Computer Science, Univ. of Manchester, UK
franconi@cs.man.ac.uk
http://www.cs.man.ac.uk/~franconi/

(Version II, draft of November 1, 2000)

Abstract

Temporal extensions of Description Logics (DL) are relevant to capture the evolving behaviour of dynamic domains, and they have been extensively considered in the literature. Several approaches for representing and reasoning with time dependent concepts have been proposed. In this Chapter a summary of the temporal logic based approaches and of the concrete domain based approaches will be presented. The Chapter will be organised according to the adopted ontologies of time: point-based and interval-based.

1 Introduction

In the Description Logic literature, several approaches for representing and reasoning with time dependent concepts have been proposed. These temporal extensions differ from each others in different ways.

- They differ on the ontology of time, whether they adopt a point-based or an interval-based notion of time. Point-based temporal description logics are usually obtained from the combination with a tense logic [Gabbay *et al.*, 1994]. Interval-based temporal description logics are usually obtained from the combination with some restriction of the interval temporal logic $\mathcal{HS}$ [Halpern and Shoham, 1991], which is undecidable in its full power.

- They differ on the way of adding the temporal dimension, i.e., whether an explicit notion of time is adopted in which temporal operators are used to build new formulæ, or temporal information is only implicit in the language by embedding a *state-change* based language – e.g., by resorting to a STRIPS-like style of representation – to represent sequences of events; see, for example, [Devanbu and Litman, 1996].

- In the case of an explicit representation of time, there is a further distinction between an *external* and an *internal* point of view; this distinction has been introduced by Finger and Gabbay [1993].

  - In the *external* method the very same individual can have different "snapshots" in different moments of time that describe the various states of the individual at these times. In this case, a temporal logic can be seen in a modular way: while an atemporal part of the language describes the "static" aspects, the temporal part relates the different snapshots describing in such a way the "dynamic" aspects.

  - In the *internal* method the different states of an individual are seen as different individual components: an individual is a collection of temporal "parts" each one holding at a particular moment. An example of this is the temporal Description Logic based on concrete domains.

In this Chapter a general framework will be introduced encompassing only the explicit approaches. After having introduced Description Logics and their relationship with modal logics, Section 4 will consider the point-based tense logical extension of Description Logics, concluding with decidability and complexity results. Section 5 analyses the interval-based extension, which has a worst computational behaviour. The concrete domains approach will be covered in Section 6, and it will be compared with the previous ones. A complete survey of the approaches not covered in this Chapter can be found in [Artale and Franconi, 2000a].

2 Description Logics

In this Section we give a brief introduction to the $\mathcal{ALCQIFO}$ description logic, which will serve as the basic representation language for the non-temporal information. With respect to the formal apparatus, we will follow the standard concept language formalism whose extensions have been summarised in [Donini *et al.*, 1996; Calvanese *et al.*, 2000]. In this perspective, Description Logics are considered as a *structured* fragment of predicate logic.

The basic types of a concept language are *concepts*, *roles*, *features*, and *individual constants*. A concept is a description gathering the common properties among a collection of individuals; from a logical point of view it is a unary predicate ranging over the domain of individuals. Inter-relationships between these individuals are represented either by means of roles (which are interpreted as binary relations over the domain of individuals) or by means of features (which are interpreted as partial functions over the domain of individuals). Individual constants denote single individuals.

According to the syntax rules of Figure 1, *concepts* (denoted by the letters C and D) are built out of *atomic concepts* (denoted by the letter A), *roles* (denoted by the letter R), and *features* (denoted by the letter p), and *individual constants* (denoted by the letter a). Roles are built out of *atomic roles* (denoted by the letter T) and *features*. Features are built out of *atomic features* (denoted by the letter f). $\mathcal{ALC}$ is the minimal description language including full negation and disjunction – i.e., propositional calculus – and it is a notational variant of the propositional multi-modal logic $\mathbf{K_m}$ [Schild, 1991] (see Section 3). In this Section, we will consider the $\mathcal{ALCQIFO}$ Description Logic, extending $\mathcal{ALC}$ with qualified cardinality restrictions ($\mathcal{Q}$), inverse roles ($\mathcal{I}$), features ($\mathcal{F}$), and individual enumerations ($\mathcal{O}$). The top part of Figure 1 defines the $\mathcal{ALC}$ sublanguage, while the lower parts define its extensions. Both the abstract and the concrete syntax are shown in the Figure.

Let us now consider the formal semantics of $\mathcal{ALCQIFO}$. We define the *meaning* of concepts as sets of individuals—as for unary predicates—and the meaning of roles as sets of pairs of individuals—as for binary predicates.

$C, D \;\rightarrow$	$A \mid$	A	(atomic conc.)
	$\top \mid$	`top`	(top)
	$\bot \mid$	`bottom`	(bottom)
	$\neg C \mid$	(`not` C)	(complement)
	$C \sqcap D \mid$	(`and` $C\ D\ \ldots$)	(conjunction)
	$C \sqcup D \mid$	(`or` $C\ D\ \ldots$)	(disjunction)
	$\forall R.C \mid$	(`all` $R\ C$)	(univ. quantifier)
	$\exists R.C \mid$	(`some` $R\ C$)	(exist. quantifier)
	$\geqslant nR.C \mid$	(`atleast` $n\ R\ C$)	(min cardinality)
	$\leqslant nR.C \mid$	(`atmost` $n\ R\ C$)	(max cardinality)
	$p : C \mid$	(`in` $p\ C$)	(selection)
	$\uparrow p \mid$	(`undefined` p)	(undefinedness)
	$p \downarrow q \mid$	(`same` $p\ q$)	(agreement)
	$p \uparrow q \mid$	(`not-same` $p\ q$)	(disagreement)
	$\{a_1, \ldots, a_n\}$	(`one-of` $\mathtt{a_1} \ldots \mathtt{a_n}$)	(enumeration)
$R \;\rightarrow$	$T \mid$	T	(atomic role)
	$R^- \mid$	(`inverse` R)	(inverse role)
	p	p	(feature)
$p, q \;\rightarrow$	$f \mid$	f	(atomic feature)
	$p \circ q$	(`compose` $p\ q$)	(path)

Figure 1: Syntax rules for $\mathcal{ALCQIFO}$

Formally, an *interpretation* is a pair $\mathcal{I} = \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}} \rangle$ consisting of a set $\Delta^{\mathcal{I}}$ of individuals (the *domain* of $\mathcal{I}$) and a function $\cdot^{\mathcal{I}}$ (the *interpretation function* of $\mathcal{I}$) mapping every concept to a subset of $\Delta^{\mathcal{I}}$, every role to a subset of $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$, every feature to a partial function from $\Delta^{\mathcal{I}}$ to $\Delta^{\mathcal{I}}$, and every individual constant to an element of $\Delta^{\mathcal{I}}$, such that the equations in Figure 2 are satisfied. The additional unique name assumption should be fulfilled: $a^{\mathcal{I}} \neq b^{\mathcal{I}}$ if $a \neq b$. The semantics of the language can also be given by stating equivalences among expressions of the language and open First Order Logic formulæ. An atomic concept A, an atomic role T, an atomic feature f are

$$
\begin{aligned}
\top^{\mathcal{I}} &= \Delta^{\mathcal{I}} \\
\bot^{\mathcal{I}} &= \emptyset \\
(\neg C)^{\mathcal{I}} &= \Delta^{\mathcal{I}} \setminus C^{\mathcal{I}} \\
(C \sqcap D)^{\mathcal{I}} &= C^{\mathcal{I}} \cap D^{\mathcal{I}} \\
(C \sqcup D)^{\mathcal{I}} &= C^{\mathcal{I}} \cup D^{\mathcal{I}} \\
(\forall R.C)^{\mathcal{I}} &= \{x \in \Delta^{\mathcal{I}} \mid \forall y. R^{\mathcal{I}}(x,y) \Rightarrow C^{\mathcal{I}}(y)\} \\
(\exists R.C)^{\mathcal{I}} &= \{x \in \Delta^{\mathcal{I}} \mid \exists y. R^{\mathcal{I}}(x,y) \wedge C^{\mathcal{I}}(y)\} \\
\\
(\geqslant n R.C)^{\mathcal{I}} &= \{x \in \Delta^{\mathcal{I}} \mid \ \sharp\{y \in \Delta^{\mathcal{I}} \mid R^{\mathcal{I}}(x,y) \wedge C^{\mathcal{I}}(y)\} \geq n\} \\
(\leqslant n R.C)^{\mathcal{I}} &= \{x \in \Delta^{\mathcal{I}} \mid \ \sharp\{y \in \Delta^{\mathcal{I}} \mid R^{\mathcal{I}}(x,y) \wedge C^{\mathcal{I}}(y)\} \leq n\} \\
\\
(p : C)^{\mathcal{I}} &= \{x \in \mathsf{dom} p^{\mathcal{I}} \mid C^{\mathcal{I}}(p^{\mathcal{I}}(x))\} \\
(\uparrow p)^{\mathcal{I}} &= \Delta^{\mathcal{I}} \setminus \mathsf{dom} p^{\mathcal{I}} \\
(p \downarrow q)^{\mathcal{I}} &= \{x \in \mathsf{dom} p^{\mathcal{I}} \cap \mathsf{dom} q^{\mathcal{I}} \mid p^{\mathcal{I}}(x) = q^{\mathcal{I}}(x)\} \\
(p \uparrow q)^{\mathcal{I}} &= \{x \in \mathsf{dom} p^{\mathcal{I}} \cap \mathsf{dom} q^{\mathcal{I}} \mid p^{\mathcal{I}}(x) \neq q^{\mathcal{I}}(x)\} \\
\\
\{a_1, \ldots, a_n\}^{\mathcal{I}} &= \{a_1^{\mathcal{I}}, \ldots, a_n^{\mathcal{I}}\} \\
\\
(R^{-})^{\mathcal{I}} &= \{(x,y) \in \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \mid R^{\mathcal{I}}(y,x)\} \\
(p \circ q)^{\mathcal{I}} &= p^{\mathcal{I}} \circ q^{\mathcal{I}}
\end{aligned}
$$

Figure 2: Extensional semantics of $\mathcal{ALCQIFO}$

mapped respectively to the open formulæ $F_A(x) = A(x)$, $F_T(x) = T(x,y)$, $F_f(x) = f(x,y)$ – with f a functional relation, also written $f(x) = y$ – and x, y denoting the free variables; an individual constant a is represented by means of the corresponding constant a. Figure 3 gives the transformational semantics of $\mathcal{ALCQIFO}$ expressions in terms of equivalent FOL well-formed formulæ. A concept C, a role R and a path p correspond to the FOL open formulæ $F_C(x)$, $F_R(x,y)$, and $F_p(x,y)$, respectively. It is worth noting that, using the standard model-theoretic semantics, the extensional semantics of Figure 2 can be derived from the transformational semantics of Figure 3.

As an example of a concept, we can consider the concept of HAPPY FATHER, defined using the atomic concepts `Man, Doctor, Rich, Famous` and the roles `CHILD, FRIEND`. The concept HAPPY FATHER can be expressed in $\mathcal{ALCQIFO}$ as

$$
\begin{array}{rl}
\top^{\mathcal{I}} \sim & \mathsf{true} \\
\bot^{\mathcal{I}} \sim & \mathsf{false} \\
(\neg C)^{\mathcal{I}} \sim & \neg F_C(x) \\
(C \sqcap D)^{\mathcal{I}} \sim & F_C(x) \wedge F_D(x) \\
(C \sqcup D)^{\mathcal{I}} \sim & F_C(x) \vee F_D(x) \\
(\forall R.C)^{\mathcal{I}} \sim & \forall z. F_R(x,z) \Rightarrow F_C(z) \\
(\exists R.C)^{\mathcal{I}} \sim & \exists z. F_R(x,z) \wedge F_C(z) \\
& \\
(\geqslant n R.C)^{\mathcal{I}} \sim & \exists^{\geq n} z. F_R(x,z) \wedge F_C(z) \\
(\leqslant n R.C)^{\mathcal{I}} \sim & \exists^{\leq n} z. F_R(x,z) \wedge F_C(z) \\
& \\
(p : C)^{\mathcal{I}} \sim & \exists z. F_p(x,z) \wedge F_C(z) \\
(\uparrow p)^{\mathcal{I}} \sim & \neg \exists z. F_p(x,z) \\
(p \downarrow q)^{\mathcal{I}} \sim & \exists z. F_p(x,z) \wedge F_q(x,z) \\
(p \uparrow q)^{\mathcal{I}} \sim & \exists z_1, z_2. F_p(x,z_1) \wedge F_q(x,z_2) \wedge z_1 \neq z_2 \\
& \\
\{a_1, \ldots, a_n\}^{\mathcal{I}} \sim & x = a_1 \vee \ldots \vee x = a_n \\
& \\
(R^{-})^{\mathcal{I}} \sim & F_R(y,x) \\
(p \circ q)^{\mathcal{I}} \sim & \exists z. F_p(x,z) \wedge F_q(z,y)
\end{array}
$$

Figure 3: FOL semantics of $\mathcal{ALCQIFO}$

$$\mathtt{Man} \sqcap (\exists \mathtt{CHILD}.\top) \sqcap \forall \mathtt{CHILD}.(\mathtt{Doctor} \sqcap \exists \mathtt{FRIEND}.(\mathtt{Rich} \sqcup \mathtt{Famous})),$$

i.e., those men having some child and all of whose children are doctors having some friend who is rich or famous.

A *knowledge base*, in this context, is a finite set Σ of two types of *formulæ*: *terminological axioms* (TBox) and *assertional axioms* (ABox). For an atomic concept A, and (possibly complex) concepts C, D, terminological axioms are of the form $A \doteq C$ (concept definition), $A \sqsubseteq C$ (primitive concept definition), $C \sqsubseteq D$ (general inclusion statement). An interpretation $\mathcal{I}$ satisfies the formula $C \sqsubseteq D$ if and only if the interpretation of C is included in the interpretation of D, i.e., $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$. It is clear that the last kind of axiom is a generalisation of the first two: concept definitions of the type $A \doteq C$ – where A is an atomic concept – can be reduced to the pair of axioms $(A \sqsubseteq C)$ and $(C \sqsubseteq A)$. For example, the abovementioned concept could be used to explicitly define the concept HappyFather:

$$\texttt{HappyFather} \doteq \texttt{Man} \sqcap (\exists \texttt{CHILD}.\top) \ \sqcap \\ \forall \texttt{CHILD}.(\texttt{Doctor} \sqcap \exists \texttt{FRIEND}.(\texttt{Rich} \sqcup \texttt{Famous}))$$

Extensional knowledge is expressed by means of an ABox which is formed by a finite set of *assertional axioms*, i.e. formulæ on individual constants. An assertion is an axiom of the form $C(a)$, $R(a,b)$ or $p(a,b)$, where a and b are individual constants. A formula $C(a)$ is satisfied by an interpretation $\mathcal{I}$ iff $a^{\mathcal{I}} \in C^{\mathcal{I}}$, $P(a,b)$ is satisfied by $\mathcal{I}$ iff $(a^{\mathcal{I}}, b^{\mathcal{I}}) \in P^{\mathcal{I}}$, and $p(a,b)$ is satisfied by $\mathcal{I}$ iff $p^{\mathcal{I}}(a^{\mathcal{I}}) = b^{\mathcal{I}}$.

For example, the individual constant john, as defined by the following *ABox*, could be recognised as an HappyFather:

```
Man(john), CHILD(john,bill),
Doctor(bill), FRIEND(bill,peter), Rich(peter).
```

An interpretation $\mathcal{I}$ is a *model* of a knowledge base Σ iff every formula in Σ is satisfied by $\mathcal{I}$. If Σ has a model, then it is *satisfiable*; thus, checking for KB satisfiability is deciding whether there is at least one model for the knowledge base. Σ *logically implies* a formula ϕ (written $\Sigma \models \phi$) if ϕ is satisfied by every model of Σ. In particular, we say that a concept C is *subsumed* by a concept D in a knowledge base Σ (written $\Sigma \models C \sqsubseteq D$) if $C^{\mathcal{I}} \subseteq D^{\mathcal{I}}$ for every model $\mathcal{I}$ of Σ. We say that an individual constant a is an *instance of* a concept C in a knowledge base Σ (written $\Sigma \models C(a)$) if $a^{\mathcal{I}} \in C^{\mathcal{I}}$ for every model $\mathcal{I}$ of Σ.

For example, the concept

$$\texttt{Person} \sqcap (\exists \texttt{CHILD}.\texttt{Person})$$

denoting the PARENT class – i.e., the persons having at least a child which is a person – subsumes the concept HappyFather with respect to the following knowledge base Σ:

$$\texttt{Doctor} \doteq \texttt{Person} \sqcap \exists \texttt{DEGREE}.\texttt{Phd}, \\ \texttt{Man} \doteq \texttt{Person} \sqcap \texttt{sex} : \texttt{Male},$$

i.e., every happy father is also a person having at least one child, given the background knowledge that men are male persons, and that doctors are persons.

A concept C is satisfiable, given a knowledge base Σ, if there is at least one model $\mathcal{I}$ of Σ such that $C^{\mathcal{I}} \neq \emptyset$, i.e. $\Sigma \not\models C \doteq \bot$. For example, the concept

$(\exists \mathtt{CHILD}.\mathtt{Man}) \sqcap (\forall \mathtt{CHILD}.(\mathtt{sex} : \neg \mathtt{Male}))$

is unsatisfiable with respect to the above knowledge base Σ. In fact, an individual whose children are not male cannot have a child being a man.

Concept subsumption can be reduced to concept satisfiability since C is subsumed by D in Σ if and only if $(C \sqcap \neg D)$ is unsatisfiable in Σ.

The various sublanguages of $\mathcal{ALCQIFO}$ have different computational properties. Deciding knowledge base satisfiability, concept satisfiability, and logical implication in $\mathcal{ALCQIO}$ is NEXPTIME-complete [Tobies, 2000]. Deciding knowledge base satisfiability, concept satisfiability, and logical implication in $\mathcal{ALCQI}$, $\mathcal{ALCQO}$, $\mathcal{ALCIO}$ is EXPTIME-complete [Calvanese *et al.*, 2000]. Checking concept satisfiability, concept subsumption and instance checking with empty knowledge bases in $\mathcal{ALCF}$ is PSPACE-complete [Hollunder *et al.*, 1990]. Deciding knowledge base satisfiability, concept satisfiability, and logical implication in the full $\mathcal{ALCQIFO}$ is undecidable – it is already undecidable in $\mathcal{ALCF}$ [Lutz, 1999a].

3 Correspondence with Modal Logics

[Schild, 1991] proved the correspondence between $\mathcal{ALC}$ and the propositional normal multi-modal logic $\mathbf{K_m}$ [Goldblatt, 1983; Halpern and Moses, 1985]. $\mathbf{K_m}$ is the simplest normal multi-modal logic interpreted over Kripke structures: there are no restrictions on the accessibility relations. Informally, an $\mathcal{ALC}$ individual corresponds to a $\mathbf{K_m}$ possible world, and an $\mathcal{ALC}$ concept corresponds to a $\mathbf{K_m}$ propositional formula, which is interpreted as the set of possible worlds over which the formula holds. The existential and universal quantifiers correspond to the possibility and necessity operators over different accessibility relations R: $\Box_R\ \psi$ is interpreted as the set of all the possible worlds such that in every R-accessible world ψ holds; $\Diamond_R\ \psi$ is interpreted as the set of all the possible worlds such that in some R-accessible world ψ holds. Thus, roles correspond to the accessibility relations between worlds. Figure 4 shows the correspondence between an $\mathcal{ALC}$ concept C and a $\mathbf{K_m}$ propositional formula ψ_C. It is easy to see that both the notion of model and the reasoning problems in $\mathcal{ALC}$ have obvious counterparts in $\mathbf{K_m}$.

$\mathcal{ALC}$	$\mathbf{K_m}$
C denotes a set of individuals	ψ_C denotes a set of worlds
R denotes a set of pairs of individuals	R is an accessibility relation
A	A
$C \sqcap D$	$\psi_C \wedge \psi_D$
$C \sqcup D$	$\psi_C \vee \psi_D$
$\neg C$	$\neg \psi_C$
$\forall R.C$	$\Box_R\ \psi_C$
$\exists R.C$	$\Diamond_R\ \psi_C$

Figure 4: Correspondence between $\mathcal{ALC}$ concepts and $\mathbf{K_m}$ formulæ.

[Calvanese *et al.*, 2000] define a very expressive Description Logic – $\mathcal{ALCQI}_{reg}$ – which extends the expressivity of $\mathcal{ALCQI}$ with *regular expressions* over roles, and prove its correspondence with of *CPDL*, i.e. the propositional dynamic modal logic *PDL* with the converse operator, extended with *graded modalities*. Deciding knowledge base satisfiability, concept satisfiability, and logical implication in $\mathcal{ALCQI}_{reg}$ is EXPTIME-complete [Calvanese *et al.*, 2000]. The sublanguage $\mathcal{ALC}_{reg}$ which is in correspondence with *PDL* is often called $\mathcal{C}$, and $\mathcal{ALCQI}_{reg}$ is called $\mathcal{CIQ}$.

4 Point-based notion of time

This Section illustrates how to extend Description Logics with a point-based notion of time. In order to intuitively understand the meaning of the temporal operators that are being added to the Description Logic, let us consider as an example a simple definition of the temporally dependent concept `Mortal`:

$\texttt{Mortal} \doteq \texttt{LivingBeing} \sqcap \Diamond^{+}\neg\texttt{LivingBeing}$

The concept denotes the set of pairs $\langle t, a\rangle$ where a is a kind of `LivingBeing` at the time t, and there exists an instant $v > t$ where a is no more a `LivingBeing`.

The operator *universal future*, $\Box^+$, is the dual of $\Diamond^+$. Given a time point t, the concept $\Box^+ C$ denotes the set of individuals which are of kind C at every time $v > t$. With this operator, the definition of a mortal can be refined by saying that from a certain future time, $v > t$, he/she will never be alive again:

$\texttt{Mortal} \doteq \texttt{LivingBeing} \sqcap \Diamond^+ \Box^+ \neg \texttt{LivingBeing}$

This definition is still incomplete since does not tell anything about the time between t – when the mortal is alive – and v – when a mortal dies. At each time w with $t < w < v$, a mortal can be dead or alive. For this purpose the binary operator *until*, $\mathcal{U}$, can be used. At time t, the concept $C\,\mathcal{U}\,D$ denotes all those individuals which are of kind D at some time $v > t$ and which are of kind C for all times w with $t < w < v$. Thus, a mortal can be redefined as a living being who is alive *until* he dies:

$\texttt{Mortal} \doteq \texttt{LivingBeing} \sqcap (\texttt{LivingBeing}\,\mathcal{U}\,(\neg\texttt{LivingBeing} \sqcap \Box^+ \neg\texttt{LivingBeing}))$

More formally, complex temporal concepts can be expressed using the following syntax.

Definition 4.1 *The* tense-logical extension *of a concept language $\mathcal{L}$, called $\mathcal{L}_{\mathcal{US}}$, is the least set containing all concepts, roles and formulæ of $\mathcal{L}$, such that $C\,\mathcal{U}\,D, C\,\mathcal{S}\,D$ are concepts of $\mathcal{L}_{\mathcal{US}}$ if C and D are concepts of $\mathcal{L}_{\mathcal{US}}$, and such that $R_1\,\mathcal{U}\,R_2, R_1\,\mathcal{S}\,R_2$ are roles of $\mathcal{L}_{\mathcal{US}}$ if R_1 and R_2 are roles of $\mathcal{L}_{\mathcal{US}}$. If ϕ and ψ are formulæ of $\mathcal{L}_{\mathcal{US}}$ then so are $\neg\phi, \phi \wedge \psi, \phi\,\mathcal{U}\,\psi, \phi\,\mathcal{S}\,\psi$. The sublanguage of $\mathcal{L}_{\mathcal{US}}$ without temporal roles is called $\mathcal{L}^-_{\mathcal{US}}$.*

The $\mathcal{L}_{\mathcal{US}}$ semantics naturally extends with time the standard non-temporal semantics of $\mathcal{L}$ [Baader and Ohlbach, 1995; Wolter and Zakharyaschev, 1998]. A temporal structure $\mathcal{T} = (\mathcal{P}, <)$ is assumed, where $\mathcal{P}$ is a set of (time) points and $<$ is a strict linear order on $\mathcal{P}$. A $\mathcal{L}_{\mathcal{US}}$ *temporal interpretation* over $\mathcal{T}$ is a pair $\mathcal{M} = \langle \mathcal{T}, \mathcal{I} \rangle$, where $\mathcal{I}$ is a function associating to each t in $\mathcal{T}$ a standard non-temporal $\mathcal{L}$ interpretation, $\mathcal{I}(t) \doteq \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}(t)} \rangle$, such that it satisfies: the standard semantic definitions of $\mathcal{L}$ for each t in $\mathcal{T}$; the rigid designator hypothesis, $a^{\mathcal{I}(w)} = a^{\mathcal{I}(v)}$ for any $w, v \in \mathcal{T}$; plus the following[1]

$$(C\,\mathcal{U}\,D)^{\mathcal{I}(t)} = \{x \in \Delta^{\mathcal{I}} \mid \exists v.(v > t) \wedge D^{\mathcal{I}(v)}(x) \wedge \forall w.(t < w < v) \rightarrow C^{\mathcal{I}(w)}(x)\}$$
$$(C\,\mathcal{S}\,D)^{\mathcal{I}(t)} = \{x \in \Delta^{\mathcal{I}} \mid \exists v.(v < t) \wedge D^{\mathcal{I}(v)}(x) \wedge \forall w.(v < w < t) \rightarrow C^{\mathcal{I}(w)}(x)\}$$

[1]We omit the similar definitions for the temporal roles.

If the temporal structure is linear and discrete, it is possible to define the missing temporal operators: Existential Future ($\Diamond^+$), Existential Past ($\Diamond^-$), Universal Future ($\Box^+$), Universal Past ($\Box^-$), Next Instant ($\bigoplus$), Previous Instant ($\ominus$), as $\Diamond^+ C \doteq \top \,\mathcal{U}\, C$, $\Diamond^- C \doteq \top \,\mathcal{S}\, C$, $\bigoplus C \doteq \bot \mathcal{U} C$, $\ominus C \doteq \bot \mathcal{S} C$, $\Box^+ C \doteq \neg \Diamond^+ \neg C$, $\Box^- C \doteq \neg \Diamond^- \neg C$.

In addition, a language could be extended with *global roles* whose interpretation does not change in time: $R^{\mathcal{I}(w)} = R^{\mathcal{I}(v)}$ for any $w, v \in \mathcal{T}$.

Definition 4.2 *Given a formula ϕ, an interpretation $\mathcal{M} \doteq \langle \mathcal{T}, \mathcal{I} \rangle$, and a time point $t \in \mathcal{T}$, the satisfiability relation $\mathcal{M}, t \models \phi$ is defined inductively by:*

$$
\begin{array}{lll}
\mathcal{M}, t \models C \doteq D & \textit{iff} & C^{\mathcal{I}(t)} = D^{\mathcal{I}(t)} \\
\mathcal{M}, t \models C \sqsubseteq D & \textit{iff} & C^{\mathcal{I}(t)} \subseteq D^{\mathcal{I}(t)} \\
\mathcal{M}, t \models C(a) & \textit{iff} & C^{\mathcal{I}(t)}(a^{\mathcal{I}(t)}) \\
\mathcal{M}, t \models R(a, b) & \textit{iff} & R^{\mathcal{I}(t)}(a^{\mathcal{I}(t)}, b^{\mathcal{I}(t)}) \\
\mathcal{M}, t \models \phi \wedge \psi & \textit{iff} & \mathcal{M}, t \models \phi \wedge \mathcal{M}, t \models \psi \\
\mathcal{M}, t \models \neg \phi & \textit{iff} & \mathcal{M}, t \not\models \phi \\
\mathcal{M}, t \models \phi \,\mathcal{U}\, \psi & \textit{iff} & \exists v.(v > t) \wedge \mathcal{M}, v \models \psi \wedge \forall w.(t < w < v) \rightarrow \mathcal{M}, w \models \phi \\
\mathcal{M}, t \models \phi \,\mathcal{S}\, \psi & \textit{iff} & \exists v.(v < t) \wedge \mathcal{M}, v \models \psi \wedge \forall w.(v < w < t) \rightarrow \mathcal{M}, w \models \phi
\end{array}
$$

A formula ϕ is satisfiable *if there is an interpretation $\mathcal{M}$ and a time point t such that $\mathcal{M}, t \models \phi$. A formula is* valid *if for each time point $t \in \mathcal{T}$ then $\mathcal{M}, t \models \phi$.*

We show now how all the relevant reasoning problems can be reduced to satisfiability of formulæ. A concept C is *satisfiable* if there exists $\mathcal{M}$ and t such that $\mathcal{M}, t \models \neg(C \doteq \bot)$ – indeed, this means that there exists an interpretation $\mathcal{I}$ such that $C^{\mathcal{I}(t)} \neq \emptyset$ for some t. The *local logical implication* problem from a finite set of formulæ Σ (a KB) is defined as follows: $\Sigma \models^l \phi$, if for every $\mathcal{M}$ and every time point t then if $\mathcal{M}, t \models \Sigma$ then also $\mathcal{M}, t \models \phi$, i.e. $\Sigma \models^l \phi$ if and only if the formula $((\bigwedge \Sigma) \wedge \neg \phi)$ is not satisfiable. The *global logical implication* problem is defined as follows: $\Sigma \models \phi$, if for every interpretation $\mathcal{M}$ such that $\mathcal{M}, t \models \Sigma$ for every t then also $\mathcal{M}, t \models \phi$ for every t. Global logical implication is reducible to local logical implication (and then to a satisfiability problem): $\Sigma \models \phi$ if and only if $\Sigma \cup \{\Box^+ \psi \mid \psi \in \Sigma\} \cup \{\Box^- \psi \mid \psi \in \Sigma\} \models^l \phi$.
Note that – as expected – axioms in Description Logics are represented as valid formulæ in a temporal Description Logic, and the classical logical im-

plication problem in Description Logics is reformulated in a temporal Description Logics as a global logical implication.

Theorem 4.1 (Finite Model Property) *If $\mathcal{L}$ includes $\mathcal{ALC}$, then $\mathcal{L}_{\mathcal{U}}^{-}$ does not have the finite model property [Wolter and Zakharyaschev, 1998].*

Theorem 4.2 (Decidability)

1. *If $\mathcal{L}$ includes $\mathcal{ALC}$, then the problem of formula satisfiability for $\mathcal{L}_{\mathcal{U}}$ and $\mathcal{L}_{\mathcal{U}}^{-}$ with global roles is undecidable in any unbounded linear order [Wolter and Zakharyaschev, 1999a].*

2. *The problem of formula satisfiability is decidable for the following languages:*

   - *$\mathcal{CIQ}_{\mathcal{US}}^{-}$, $\mathcal{CIO}_{\mathcal{US}}^{-}$, $\mathcal{CQO}_{\mathcal{US}}^{-}$ and $\mathcal{DLR}_{\mathcal{US}}^{-}$ in $\langle \mathcal{N}, < \rangle$ and in $\langle \mathcal{Z}, < \rangle$ [Wolter and Zakharyaschev, 1999b; Artale and Franconi, 2000b],*
   - *$\mathcal{CIQ}_{\diamond}^{-}$, $\mathcal{CIO}_{\diamond}^{-}$ and $\mathcal{CQO}_{\diamond}^{-}$ in $\langle \mathcal{Q}, < \rangle$ [Wolter and Zakharyaschev, 1999b],*

3. *Concept satisfiability in $\mathcal{ALC}_{\mathcal{U}}^{-}$ with global roles with empty KBs, $\emptyset \not\models C \doteq \bot$, is decidable in $\langle \mathcal{N}, < \rangle$ [Wolter, 2000].*

Theorem 4.3 (Complexity)

1. *Concept satisfiability in $\mathcal{ALC}_{\mathcal{U}}^{-}$ with empty KBs, $\emptyset \not\models C \doteq \bot$, is PSPACE-complete in $\langle \mathcal{N}, < \rangle$ [Schild, 1993].*

2. *Concept satisfiability in $\mathcal{CF}_{\mathcal{U}}^{-}$ with empty KBs, $\emptyset \not\models C \doteq \bot$, is EXPTIME-complete in $\langle \mathcal{N}, < \rangle$ [Schild, 1993].*

3. *Formula satisfiability in $\mathcal{ALC}_{\mathcal{U}}^{-}$ and $\mathcal{ALCQI}_{\mathcal{US}}^{-}$ is EXPSPACE-complete in $\langle \mathcal{N}, < \rangle$ [Morurovic* et al.*, 2000; Artale and Franconi, 2000b].*

A tableau algorithm for formula satisfiability in $\mathcal{ALC}_{\mathcal{U}}^{-}$ with *expanding domains* has been developed in [Sturm and Wolter, 2000].

Let us consider now a fully worked out example using $\mathcal{ALCQI}_{\mathcal{US}}^{-}$, involving the concepts `Project`, `Employee`, `Manager`, and `Works-for`, and the roles `HAS-PRJ` and `HAS-EMP`.

- The concept $\texttt{Works-for}$ denotes the event of an employee working for a project:

  $$\begin{aligned}\texttt{Works-for} \sqsubseteq\ & \exists\texttt{HAS-PRJ}.\texttt{Project} \sqcap \exists\texttt{HAS-EMP}.\texttt{Employee}\ \sqcap \\ & \leqslant 1\texttt{HAS-PRJ}.\top \sqcap \leqslant 1\texttt{HAS-EMP}.\top\end{aligned}$$

- Every project has somebody working for it:

  $\texttt{Project} \sqsubseteq \exists\texttt{HAS-PRJ}^{-}.\texttt{Works-for}$

- Managers are employees:

  $\texttt{Manager} \sqsubseteq \texttt{Employee}$

- Managers are exactly those employees who do not work for a project:

  $\texttt{Manager} \doteq \forall\texttt{HAS-EMP}^{-}.\neg\texttt{Works-for}$

- A manager becomes qualified after a period when she/he was just an employee:

  $\texttt{Manager} \sqsubseteq \texttt{Qualified}\,\mathcal{S}\,(\texttt{Employee} \sqcap \neg\texttt{Manager})$

It turns out that the following formulæ are logically implied by Σ:

- For every project, there is at least an employee who is not a manager:

  $\Sigma \models \texttt{Project} \sqsubseteq \exists(\texttt{HAS-PRJ}^{-} \circ \texttt{HAS-EMP}).\neg\texttt{Manager}$

- A manager worked in a project before managing some (possibly different) project:

  $\Sigma \models \texttt{Manager} \sqsubseteq \Diamond^{-}\exists(\texttt{HAS-EMP}^{-} \circ \texttt{HAS-PRJ}).\texttt{Project}$

$\mathcal{ALCQI}_{\mathcal{US}}$ has been used to encode and reason with temporal conceptual data models in databases [Artale and Franconi, 1999].

5 Interval-based notion of time

This Section illustrates how to extend Description Logics with an interval-based notion of time. In order to intuitively understand the meaning of the temporal operators that are being added to the Description Logic, let us reconsider in this context the example of the temporally dependent concept $\texttt{Mortal}$:

Relation	Abbr.	Inverse
before(i, j)	b	a
meets(i, j)	m	mi
overlaps(i, j)	o	oi
starts(i, j)	s	si
during(i, j)	d	di
finishes(i, j)	f	fi

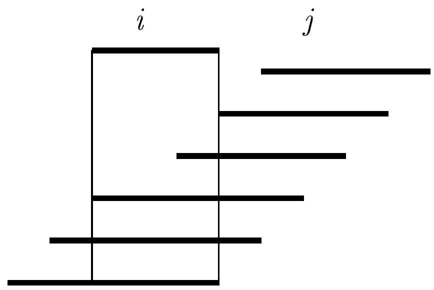

Figure 5: The Allen's interval relationships.

$\texttt{Mortal} \doteq \texttt{LivingBeing} \sqcap \langle \textsf{met-by} \rangle \neg \texttt{LivingBeing}$

The definition states that a `LivingBeing` at the reference interval will not be alive at some other interval *met by* the reference one. In this logic, the temporal operators make use of the Allen's interval algebra [Allen, 1991], as summarised in Figure 5.

Definition 5.1 *The* Allen's interval extension *of a concept language $\mathcal{L}$, called $\mathcal{L}_{\mathcal{A}}$, is the least set containing all concepts and roles of $\mathcal{L}$, such that $\langle\alpha\rangle C, [\alpha]C$ are concepts of $\mathcal{L}_{\mathcal{A}}$ if C is a concept of $\mathcal{L}_{\mathcal{A}}$, and α is one of the Allen's interval relations (Figure 5): before (*b*), meets (*m*), during (*d*), overlaps (*o*), starts (*s*), finishes (*f*), equal (=), after (*a*), met-by (*mi*), contains (*di*), overlapped-by (*oi*), started-by (*si*), finished-by (*fi*).*
$\mathcal{L}_{\mathcal{AF}}$ is the combination of the fragment of $\mathcal{L}_{\mathcal{A}}$ where existential temporal modalities $\langle\alpha\rangle C$ are only allowed at the top level of concepts and no universal temporal modalities $[\alpha]C$ are allowed, with global functional roles and with explicit variables in the language denoting temporal intervals à la Prior (see [Artale and Franconi, 1998] for details).

The $\mathcal{L}_{\mathcal{A}}$ semantics naturally extends with time the standard non-temporal semantics of $\mathcal{L}$. A linear and unbounded temporal structure $\mathcal{T} = (\mathcal{P}, <)$ is assumed, where $\mathcal{P}$ is a set of time points and $<$ is a strict linear order on $\mathcal{P}$. The *interval set* of a structure $\mathcal{T}$ is defined as the set $\mathcal{T}^{\star}_{<}$ of all closed proper intervals $[t_1, t_2] \doteq \{t \in \mathcal{P} \mid t_1 \leq t \leq t_2\}$ in $\mathcal{T}$. An $\mathcal{L}_{\mathcal{A}}$ *temporal interpretation* over $\mathcal{T}^{\star}_{<}$ is a pair $\mathcal{M} \doteq \langle \mathcal{T}^{\star}_{<}, \mathcal{I} \rangle$, where $\mathcal{I}$ is a function associating to each $i = [t_1, t_2] \in \mathcal{T}^{\star}_{<}$ a standard non-temporal $\mathcal{L}$ interpretation, $\mathcal{I}(i) \doteq \langle \Delta^{\mathcal{I}}, \cdot^{\mathcal{I}(i)} \rangle$, such that it satisfies the standard semantic definitions of $\mathcal{L}$ for each i in $\mathcal{T}^{\star}_{<}$, plus the following

$$
\begin{array}{lcl}
(\langle\alpha\rangle C)^{\mathcal{I}(i)} & = & \{x \in \Delta^{\mathcal{I}} \mid \exists j.\ \alpha(j,i) \wedge C^{\mathcal{I}(j)}(x)\} \\
([\alpha]C)^{\mathcal{I}(i)} & = & \{x \in \Delta^{\mathcal{I}} \mid \forall j.\ \alpha(j,i) \rightarrow C^{\mathcal{I}(j)}(x)\}
\end{array}
$$

where $\alpha(i,j)$ is understood from Figure 5. An interpretation $\mathcal{M} \doteq \langle \mathcal{T}^{\star}_{<}, \mathcal{I} \rangle$ is a *model* for a concept C if $C^{\mathcal{I}(i)} \neq \emptyset$ for some $i \in \mathcal{T}^{\star}_{<}$. If a concept has a model then it is *satisfiable*. C is subsumed by D, i.e. $C \sqsubseteq D$, if $C^{\mathcal{I}(i)} \subseteq D^{\mathcal{I}(i)}$ for every interpretation $\mathcal{M}$ and every $i \in \mathcal{T}^{\star}_{<}$.

Theorem 5.1 (Decidability)

1. *If $\mathcal{L}$ includes $\mathcal{ALC}$, then the problems of concept satisfiability and concept subsumption with empty KB for $\mathcal{L}_{\mathcal{A}}$ are undecidable in any unbounded linear order [Halpern and Shoham, 1991; Bettini, 1997].*

2. *The problems of concept satisfiability and concept subsumption with empty KB are decidable for $\mathcal{ALCF}_{\mathcal{AF}}$ in an unbound, dense, linear order [Artale and Franconi, 1998].*

Theorem 5.2 (Complexity)

1. *Concept satisfiability and concept subsumption with empty KB for $\mathcal{F}_{\mathcal{AF}}$ and $\mathcal{FU}_{\mathcal{AF}}$ [2] are NP-complete in an unbound, dense, linear order [Artale and Franconi, 1998],*

2. *Concept satisfiability with empty KB for $\mathcal{ALCF}_{\mathcal{AF}}$ is PSPACE-complete in an unbound, dense, linear order [Artale and Lutz, 1999].*

Let us consider now an example in the block domain using $\mathcal{ALCF}_{\mathcal{AF}}$. A stacking action involves two blocks (the parameters of the action), which should be both clear at the beginning; the central part of the action consists of holding one block; at the end, the blocks are one on top of the other, and the bottom one is no longer clear. The definition involves the concepts `Stack`, `Clear` and `Hold`, the feature `ON`, and the global functional roles *OBJECT1* and *OBJECT2*:

$$
\begin{array}{ll}
\texttt{Stack} \doteq & \langle \textsf{overlaps} \rangle \exists \mathit{OBJECT1}.\texttt{Clear} \sqcap \\
& \langle \textsf{finished-by} \rangle \exists \mathit{OBJECT2}.\texttt{Clear} \sqcap \\
& \langle \textsf{finishes} \rangle \exists \mathit{OBJECT1}.(\texttt{Hold} \sqcap \neg\texttt{Clear}) \sqcap \\
& \langle \textsf{met-by} \rangle (\mathit{OBJECT1} \circ \texttt{ON} \downarrow \mathit{OBJECT2} \sqcap \exists \mathit{OBJECT1}.\texttt{Clear})
\end{array}
$$

[2] Where $\mathcal{F}$ is the pure positive feature Description Logic, and $\mathcal{FU}$ extends it with disjunction; see [Artale and Franconi, 1998] for details.

The expression $\langle \mathsf{overlaps} \rangle \exists \mathit{OBJECT1}.\mathtt{Clear}$ means that the first parameter of the action should be a `Clear` block at an interval overlapping the reference interval of the `Stack` action. $\langle \mathsf{met\text{-}by} \rangle (\mathit{OBJECT1} \circ \mathtt{ON} \downarrow \mathit{OBJECT2} \sqcap \exists \mathit{OBJECT1}.\mathtt{Clear})$ indicates that at an interval met by the reference one the object on which $\mathit{OBJECT1}$ is placed is $\mathit{OBJECT2}$ and $\mathit{OBJECT1}$ is clear.

$\mathcal{ALCF}_{\mathcal{AF}}$ has been used to encode and reason with actions, events, and plans in Artificial Intelligence, in particular to solve the problem of *plan recognition* [Artale and Franconi, 1998].

6 Time as Concrete Domain

In the concrete domain extension of Description Logics [Baader and Hanschke, 1991], abstract individuals (i.e., elements of an abstract domain $\Delta^{\mathcal{I}}$) can now be related to values in a *concrete domain* (e.g., a temporal structure) via features. Furthermore, tuples of concrete values identified by such features can be constrained to satisfy an n-ary predicate over the concrete domain (e.g., an ordering relation).

Definition 6.1 (Concrete Domain) *A concrete domain is a pair* $\mathcal{D} = (\mathsf{dom}(\mathcal{D}), \mathsf{pred}(\mathcal{D}))$ *that consists of a set* $\mathsf{dom}(\mathcal{D})$ *(the domain), and a set of predicate symbols* $\mathsf{pred}(\mathcal{D})$*. Each predicate symbol* $P \in \mathsf{pred}(\mathcal{D})$ *is associated with an arity* n *and an* n*-ary relation* $P^{\mathcal{D}} \subseteq \mathsf{dom}(\mathcal{D})^n$*. A concrete domain* $\mathcal{D}$ *is called* admissible *iff (1)* $\mathsf{pred}(\mathcal{D})$ *is closed under negation and contains a unary predicate name* $\top_{\mathcal{D}}$ *for* $\mathsf{dom}(\mathcal{D})$*, and (2) satisfiability of finite conjunctions over* $\mathsf{pred}(\mathcal{D})$ *is decidable.*

Definition 6.2 *The* concrete domain extension *of a concept language* $\mathcal{L}$*, called* $\mathcal{L}_{\mathcal{D}}$*, is the least set containing all concepts, roles and formulæ of* $\mathcal{L}$*, such that* $\exists (p_1, \ldots, p_n).P$ *are concepts of* $\mathcal{L}_{\mathcal{D}}$ *if* p_i *are paths of atomic features of* $\mathcal{L}$ *and* P *is a* n*-ary predicate symbol of the concrete domain* $\mathcal{D}$*. If* $o_1 \ldots o_n$ *are names for concrete individuals of* $\mathcal{D}$ *and* P *is a* n*-ary predicate symbol of the concrete domain* $\mathcal{D}$ *then* $P(o_1, \ldots, o_n)$ *is an ABox formula of* $\mathcal{L}_{\mathcal{D}}$*.*
The further extension of $\mathcal{L}_{\mathcal{D}}$ *adding complex roles, called* $\mathcal{L}_{\mathcal{RD}}$*, is the least set containing all concepts, roles and formulæ of* $\mathcal{L}_{\mathcal{D}}$*, such that* $\exists (p_1, \ldots, p_n)(q_1, \ldots, q_m).P$ *are roles of* $\mathcal{L}_{\mathcal{RD}}$ *if* p_i*,* q_j *are paths of atomic features of* $\mathcal{L}_{\mathcal{D}}$ *and* P *is a* $(n+m)$*-ary predicate symbol of the concrete domain* $\mathcal{D}$*.*

The $\mathcal{L}_{\mathcal{RD}}$ semantics naturally extends with concrete domains the standard semantics of $\mathcal{L}$ [Baader and Hanschke, 1991]. A concrete domain

$\mathcal{D} = (\mathsf{dom}(\mathcal{D}), \mathsf{pred}(\mathcal{D}))$ is assumed, with $\mathsf{dom}(\mathcal{D}) \cap \Delta^{\mathcal{I}} = \emptyset$. A $\mathcal{L}_{\mathcal{RD}}$ *interpretation* extends the interpretation $\mathcal{I}$ of $\mathcal{L}$ by mapping every atomic feature f to a partial function $f^{\mathcal{I}} : \Delta^{\mathcal{I}} \to \Delta^{\mathcal{I}} \cup \mathsf{dom}(\mathcal{D})$, such that it satisfies the standard semantic definitions of $\mathcal{L}$ plus

$$
\begin{aligned}
(\exists (p_1, \ldots, p_n).P)^{\mathcal{I}} &= \{x \in \Delta^{\mathcal{I}} \mid \langle p_1^{\mathcal{I}}(x), \ldots, p_n^{\mathcal{I}}(x)\rangle \in P^{\mathcal{D}}\} \\
(\exists (p_1, \ldots, p_n)(q_1, \ldots, q_m).P)^{\mathcal{I}} &= \{(x, y) \in \Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}} \mid \\
&\qquad (p_1^{\mathcal{I}}(x), \ldots, p_n^{\mathcal{I}}(x), q_1^{\mathcal{I}}(y), \ldots, q_m^{\mathcal{I}}(y)) \in P^{\mathcal{D}}\}
\end{aligned}
$$

The semantics of the additional ABox statements involving concrete predicates follows the obvious intuition.

In this framework, assuming a concrete domain composed by temporal intervals and the Allen's predicates – proved admissible by [Lutz, 1999b] – the concept of `Mortal` can be defined as follows:

$$
\begin{aligned}
\mathtt{Mortal} \doteq\; & \mathtt{ALIVE\text{-}STATE} : \mathtt{LivingBeing} \sqcap \mathtt{DEAD\text{-}STATE} : (\neg \mathtt{LivingBeing}) \sqcap \\
& \exists (\mathtt{ALIVE\text{-}STATE} \circ \mathtt{HAS\text{-}TIME}, \mathtt{DEAD\text{-}STATE} \circ \mathtt{HAS\text{-}TIME}).\mathsf{meets}
\end{aligned}
$$

i.e., a mortal is any individual having the property of being alive at some temporal interval that meets some other temporal interval at which the same individual has the property of being dead.

Theorem 6.1 (Decidability and Complexity)

- *If $\mathcal{L}$ includes $\mathcal{ALC}$, then the problem of logical implication for $\mathcal{L}_{\mathcal{D}}$ is undecidable in any unbounded, dense, linear ordered concrete domain $\mathcal{D}$ [Baader and Hanschke, 1992].*

- *If $\mathcal{L}$ includes $\mathcal{ALC}$, then the problems of concept satisfiability, concept subsumption, and instance checking with empty TBox for $\mathcal{L}_{\mathcal{RD}}$ are undecidable [Haarslev* et al.*, 1998].*

- *The problems of concept satisfiability, concept subsumption, and instance checking with empty TBox for $\mathcal{ALCF}_{\mathcal{D}}$ are PSPACE-complete in any concrete domain $\mathcal{D}$, provided that satisfiability in $\mathcal{D}$ is in PSPACE [Lutz, 1999b].*

We introduce now an example which shows that the $\mathcal{ALC}_{\mathcal{D}}$ Description Logic is more suitable to describe properties of temporal objects (e.g., intervals) rather than properties of objects varying in time (like in the `Mortal` example). In [Baader and Hanschke, 1991] the Allen's interval relations is internally defined using the set of real numbers $\Re$ together with the predicates $\leq, \leq, \geq, \geq, =, \neq$ as the concrete admissible domain. The `Interval` concept can be defined as an ordered pair of real numbers by referring to the concrete predicate $\leq$ applied to the features LEFT-HAS-TIME and RIGHT-HAS-TIME:

$$\texttt{Interval} \doteq \exists(\texttt{LEFT-HAS-TIME}, \texttt{RIGHT-HAS-TIME}). \leq$$

Allen's relations are binary relations on two intervals and are represented by the `Pair` concept which uses the features FIRST and SECOND:

$$\texttt{Pair} \doteq \exists\texttt{FIRST}.\texttt{Interval} \sqcap \exists\texttt{SECOND}.\texttt{Interval}$$

Now Allen's relation can be easily defined as concepts:

$$\begin{array}{lcl}
\texttt{C-Equals} & \doteq & \texttt{Pair} \sqcap \exists(\texttt{FIRST} \circ \texttt{LEFT-HAS-TIME}, \texttt{SECOND} \circ \texttt{LEFT-HAS-TIME}). = \\
 & & \quad \sqcap \exists(\texttt{FIRST} \circ \texttt{RIGHT-HAS-TIME}, \texttt{SECOND} \circ \texttt{RIGHT-HAS-TIME}). = \\
\texttt{C-Before} & \doteq & \texttt{Pair} \sqcap \exists(\texttt{FIRST} \circ \texttt{RIGHT-HAS-TIME}, \texttt{SECOND} \circ \texttt{LEFT-HAS-TIME}). \leq \\
\texttt{C-Meets} & \doteq & \texttt{Pair} \sqcap \exists(\texttt{FIRST} \circ \texttt{RIGHT-HAS-TIME}, \texttt{SECOND} \circ \texttt{LEFT-HAS-TIME}). = \\
\ldots & &
\end{array}$$

As an example of $\mathcal{ALC}_{\mathcal{RD}}$, we can define the BEFORE role as being the counterpart of the concrete predicate before in the abstract domain, and use it for defining a new concept `NoBefore`, as the class of objects which do not have any BEFORE-related object:

$$\texttt{BEFORE} \equiv \exists(\texttt{HAS-TIME})(\texttt{HAS-TIME}).\mathsf{before}$$
$$\texttt{NoBefore} \doteq \forall\texttt{BEFORE}.\bot$$

Let us compare the above definition with a similar one in $\mathcal{ALC}_{\mathcal{A}}$

$$\texttt{NoBefore} \doteq [\mathsf{before}]\bot$$

Assuming that in both cases the temporal structure is isomorphic to the real numbers $\Re$, while the concept `NoBefore` in the concrete domain approach is satisfiable, denoting all the objects of the *abstract* domain having no BEFORE-related objects, the concept `NoBefore` in $\mathcal{ALC}_{\mathcal{A}}$ is clearly unsatisfiable. The reason is that in $\mathcal{ALC}_{\mathcal{RD}}$ we can only quantify over the abstract domain and not over the concrete one, i.e., we can only quantify over the abstract objects which may *possibly* have a specific temporal facet lifted up from the concrete domain. On the other hand, in $\mathcal{ALC}_{\mathcal{A}}$ both the abstract objects and the temporal elements are first-class citizens, resulting in a language where it is possible to quantify on both abstract objects and temporal elements.

A partial study on the relative expressive power between the languages $\mathcal{ALCF}_{\mathcal{AF}}$ and $\mathcal{ALCF}_{\mathcal{D}}$ has been conducted [Artale and Lutz, 1999]. In particular, it has been proved how the satisfiability of a $\mathcal{ALCF}_{\mathcal{AF}}$ concept can be reduced to the satisfiability of some corresponding concept in the language $\mathcal{ALCF}_{\mathcal{D}}$. The limit of this result is that the encoding preserves only satisfiability, and it does not clarify the real relationships between the two languages with respect to the problems of subsumption and logical implication.

References

[Allen, 1991] James F. Allen. Temporal reasoning and planning. In James F. Allen, Henry A. Kautz, Richard N. Pelavin, and Josh D. Tenenberg, editors, *Reasoning about Plans*, chapter 1, pages 2–68. Morgan Kaufmann, 1991.

[Artale and Franconi, 1998] Alessandro Artale and Enrico Franconi. A temporal description logic for reasoning about actions and plans. *J. of Artificial Intelligence Research*, 9:463–506, 1998.

[Artale and Franconi, 1999] Alessandro Artale and Enrico Franconi. Temporal ER modeling with description logics. In *Proc. of the International Conference on Conceptual Modeling (ER'99)*. Springer-Verlag, 1999.

[Artale and Franconi, 2000a] Alessandro Artale and Enrico Franconi. A survey of temporal extensions of description logics. *Annals of Mathematics and Artificial Intelligence*, 2000. To appear.

[Artale and Franconi, 2000b] Alessandro Artale and Enrico Franconi. Temporal description logics for conceptual modelling. Technical report, Department of Computer Science, University of Manchester, UK, July 2000.

[Artale and Lutz, 1999] Alessandro Artale and Carsten Lutz. A correspondence between temporal description logics. In *Proc. of the 1999 Description Logic Workshop (DL'99)*, pages 145–149, 1999.

[Baader and Hanschke, 1991] Franz Baader and Philipp Hanschke. A scheme for integrating concrete domains into concept languages. In *Proc. of the 12 th IJCAI*, pages 446–451, Sidney, Australia, 1991.

[Baader and Hanschke, 1992] Franz Baader and Philipp Hanschke. Extensions of concept languages for a mechanical engineering applications. In *Proc. 16th German Annual Conference on Artificial Intelligence (GWAI'92)*, pages 132–143, 1992.

[Baader and Ohlbach, 1995] F. Baader and H.J. Ohlbach. A multidimensional terminological knowledge representation language. *J. of Applied Non-Classical Logics*, 5:153–197, 1995.

[Bettini, 1997] Claudio Bettini. Time dependent concepts: Representation and reasoning using temporal description logics. *Data and Knowledge Engineering*, 22(1):1–38, 1997.

[Calvanese *et al.*, 2000] Diego Calvanese, Giuseppe De Giacomo, Maurizio Lenzerini, and Daniele Nardi. Reasoning in expressive description logics. In Alan Robinson and Andrei Voronkov, editors, *Handbook of Automated Reasoning*. Elsevier Science Publishers, Amsterdam, 2000. To appear.

[Devanbu and Litman, 1996] Premkumar T. Devanbu and Diane J. Litman. Taxonomic plan reasoning. *Artificial Intelligence*, 84:1–35, 1996.

[Donini *et al.*, 1996] F. Donini, M. Lenzerini, D. Nardi, and A. Schaerf. Reasoning in description logics. In G. Brewka, editor, *Principles of Knowledge Representation and Reasoning*, pages 193–238. Studies in Logic, Language and Information, CLSI Publications, 1996.

[Finger and Gabbay, 1993] M. Finger and D. Gabbay. Adding a temporal dimension to a logic system. *"Journal of Language and Information"*, 1:203–233, 1993.

[Gabbay *et al.*, 1994] D. Gabbay, I. Hodkinson, and M. Reynolds. *Temporal Logic*. Oxford University Press, 1994.

[Goldblatt, 1983] R. Goldblatt. *Logics of Time and Computation*. CSLI Publications, 1983.

[Haarslev *et al.*, 1998] Volker Haarslev, Carsten Lutz, and Ralf Möller. Foundation of spatiotemporal reasoning with description logics. In A.Cohn, L.Schubert, and S.C.Shapiro, editors, *Proc. of the 6 th International Conference on Principles of Knowledge Representation and Reasoning*, pages 112–123, Trento, Italy, June 1998. Morgan Kaufmann.

[Halpern and Moses, 1985] J. Y. Halpern and Y. Moses. A guide to the modal logic of knowledge and belief: Preliminary draft. In *Proc. of the 9th Int. Joint Conf. on Artificial Intelligence (IJCAI'85)*, pages 480–490, 1985.

[Halpern and Shoham, 1991] J. Y. Halpern and Y. Shoham. A propositional modal logic of time intervals. *J. of the ACM*, 38(4):935–962, 1991.

[Hollunder *et al.*, 1990] B. Hollunder, W. Nutt, and M. Schmidt-Schauß. Subsumption algorithms for concept description languages. In *Proc. of the 9 th ECAI*, pages 348–353, Stockholm, Sweden, 1990.

[Lutz, 1999a] Carsten Lutz. Complexity of terminological reasoning revisited. In Harald Ganzinger, David McAllester, and Andrei Voronkov, editors, *Proceedings of the 6th International Conference on Logic for Programming and Automated Reasoning (LPAR'99)*, number 1705 in Lecture Notes in Artificial Intelligence, pages 181–200. Springer-Verlag, September 1999.

[Lutz, 1999b] Carsten Lutz. Reasoning with concrete domains. In *Proc. of the 16 th IJCAI*, pages 90–95, Stockholm, Sweden, 1999.

[Morurovic *et al.*, 2000] M. Morurovic, F. Wolter, and M. Zakharyaschev. Modalized description logic - how much? Technical report, Computer Science Department, University of Leipzig, Germany, 2000. To appear.

[Schild, 1991] Klaus D. Schild. A correspondence theory for terminological logics: Preliminary report. In *Proc. of the 12 th IJCAI*, pages 466–471, Sidney, Australia, 1991.

[Schild, 1993] Klaus D. Schild. Combining terminological logics with tense logic. In *Proc. of the 6th Portuguese Conf. on Artificial Intelligence (EPIA'93)*, 1993.

[Sturm and Wolter, 2000] Holger Sturm and Frank Wolter. A tableau calculus for temporal description logics: the expanding domain case. *J. of Logic and Computation*, 2000.

[Tobies, 2000] Stephan Tobies. The complexity of reasoning with cardinality restrictions and nominals in expressive description logics. *J. of Artificial Intelligence Research*, 12:199–217, May 2000.

[Wolter and Zakharyaschev, 1998] F. Wolter and M. Zakharyaschev. Satisfiability problem in description logics with modal operators. In *Proc. of the 6th Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR'98)*, pages 512–523, 1998.

[Wolter and Zakharyaschev, 1999a] F. Wolter and M. Zakharyaschev. Modal description logics: Modalizing roles. *Fundamenta Informaticae*, 39:411–438, 1999.

[Wolter and Zakharyaschev, 1999b] F. Wolter and M. Zakharyaschev. Temporalizing description logics. In D. Gabbay and M. de Rijke, editors, *Frontiers of Combining Systems*. Studies Press-Wiley, 1999. Also in the Proceedings of *FroCoS'98*, Amsterdam, 1998.

[Wolter, 2000] F. Wolter. The product of CPDL and polymodal K. *J. of Logic and Computation*, 10(2):223–251, 2000.