Homogeneous Concepts in a Temporal Description Logic

Alessandro Artale[*] and **Enrico Franconi**
Knowledge Representation and Reasoning Lab.
IRST, 38050 Povo TN, Italy
{artale, franconi}@irst.it

Claudio Bettini
University of Milano
via Comelico 39, 20135 Milano, Italy
bettini@imiucca.csi.unimi.it

1 Introduction

In the temporal literature *homogeneity* is a property of predicates in relation to time. Homogeneity characterizes the temporal behavior of world states: when a state holds over an interval of time t, it also holds over subintervals of t. Thus, if a parking is free on Sunday, one can conclude that it is also free on Sunday morning.

On the other hand, actions are not necessarily homogeneous. In the linguistic literature a difference is made between *activity* and *performance* verbs, the distinction comes out in the fact that performance verbs do not, whereas activity verbs do, have subevents that are denoted by the same verbs. Generally, activity verbs represent ongoing events, for example *to eat* and *to run*, and can be described as homogeneous predicates; whereas performance verbs represent events with a well defined granularity in time, like *to prepare spaghetti*. Performance verbs are example of anti-homogeneous events: if they occur over an interval of time t, then they do not occur over a subinterval of t, as they would not yet be completed.

2 The Temporal Language

We describe a simple description logic to represent classes of individuals and their temporal relations – related work can be found in [Schmiedel,1990; Weida and Litman,1992; Bettini,1992]. This extends with the homogeneity operator the language presented in [Artale and Franconi,1994a; 1994b; 1993; Artale,1994]. For sake of simplicity, the language presented here does not include complex concept forming operators, like disjunction and temporal substitutive qualifier.

Basic types of the language are *concepts*, *individuals*, *temporal variables* and *intervals*. A concept is a description gathering the common properties among a collection of individuals. Concepts can describe entities of the world, states, events and processes. Temporal variables denote intervals bound by temporal constraints, by means of which abstract temporal patterns in the form of constraint networks are expressed. Concepts (resp. individuals) can be specified to hold at a certain interval variable (resp. value) defined by the constraint network.

[*]Current address: Ladseb-CNR, I-35020 Padova PD, Italy

C, D	$\rightarrow$	$A \mid$	(atomic concept)
		$\top \mid$	(top)
		$C \sqcap D \mid$	(conjunction)
		$p \downarrow q \mid$	(agreement)
		$p : C \mid$	(selection)
		$C@X \mid$	(temporal qualifier)
		$\Diamond(X^+)\ \mathcal{T}c^+.\ C \mid$	(temporal existential quantifier)
		∇C	(homogeneous concept)
p, q	$\rightarrow$	$f \mid$	(atomic feature)
		$\star g \mid$	(atomic parametric feature)
		$p \circ q$	(feature chain)
$\mathcal{T}c$	$\rightarrow$	$(X\ (R)\ Y)$	(temporal constraint)
R, S	$\rightarrow$	$R\ ,\ S \mid$	(disjunction of relations)
		$\mathsf{s} \mid \mathsf{mi} \mid \mathsf{f} \mid \ldots$	(Allen basic temporal relations)
X, Y	$\rightarrow$	$\sharp \mid \mathrm{x} \mid \mathrm{y} \mid \ldots$	(temporal variables)

Figure 1: Syntax rules for the temporal concept language.

In this way, *action types* (resp. *individual actions*) can be represented in a uniform way by temporally related concepts (resp. individuals).

Concept expressions (denoted by C, D) are syntactically built out of *atomic concepts* (denoted by A), *atomic features* (denoted by f), *atomic parametric features* (denoted by $\star g$) and constrained *interval variables* (denoted by X, Y) according to the abstract syntax rules of figure 1. For the basic interval relations we use the same notation as in [Allen,1991]: 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). Temporal variables are introduced by the temporal existential quantifier "$\Diamond$". Variables appearing in temporal constraints should be declared within the same temporal quantifier, with the exception of the special variable $\sharp$.

Concept expressions are interpreted in our logic over pairs of *temporal intervals* and *individuals* $\langle i, a\rangle$, meaning that the individual a is in the extension of the concept at the interval i. If a concept is intended to de-

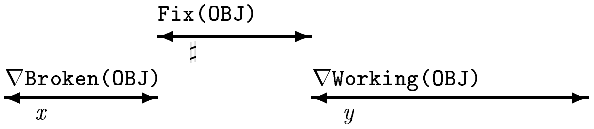

Figure 2: Temporal dependencies in the definition of the `Fix` action.

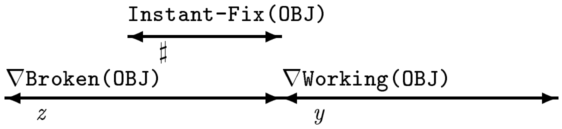

Figure 3: Temporal dependencies in the definition of the `Instant-Fix` action.

note an action, then its interpretation can be seen as the set of individual actions of that type occurring at some interval. Homogeneous concepts "∇C" are also temporal concepts. Their semantics will show that they are obtained by a special case of universal temporal quantifier "$\Box$" [Bettini,1993a]. For a formal definition of the semantics of the homogeneity-free language see [Artale and Franconi,1994a].

Within a concept, the special "$\sharp$" variable denotes the generic interval at which the concept itself *holds*; in the case of actions, it refers to the temporal interval at which the action itself *occurs*. A concept holds at an interval X if it is temporally qualified at X – written $C@X$; in this way, every occurrence of $\sharp$ embedded within the concept expression C is interpreted as the X variable. Since any concept is implicitly temporally qualified at the special $\sharp$ variable – $C \equiv C@\sharp$ – it is not necessary to explicitly qualify concepts at $\sharp$. The temporal existential quantifier introduces interval variables, related each other and possibly to the $\sharp$ variable in a way defined by the set of *temporal constraints*.

The semantics of homogeneous concepts is easily given in terms of the semantics of the universal temporal quantifier, the counterpart of the existential temporal quantifier operator available in this language. In fact, the following equivalence holds:

$$\nabla C \equiv \Box x \ (x \ (=, \mathsf{s}, \mathsf{d}, \mathsf{f}) \ \sharp).\ C@x$$

meaning that, C is an homogeneous concept (∇C) if and only if when it holds at an interval it remains true at each subinterval. In particular, $\Box x$ universally qualifies the temporal variable x, while the temporal constraint $(x \ (=, \mathsf{s}, \mathsf{d}, \mathsf{f}) \ \sharp)$ has to be read as the disjunction $(x = \sharp) \vee (x \ \mathsf{s} \ \sharp) \vee (x \ \mathsf{d} \ \sharp) \vee (x \ \mathsf{f} \ \sharp)$, imposing that x is a generic interval contained in $\sharp$. Moreover, it is always true that $\nabla C \sqsubseteq C$, i.e. ∇C is a more specific concept than C.

Let us consider as an example the definition of the `Fix` action:

$$\begin{aligned}&\texttt{Fix} \ \doteq\\ &\quad \Diamond(x \ y) \ (x \ \mathsf{m} \ \sharp)(\sharp \ \mathsf{m} \ y).\\ &\qquad (\star\texttt{OBJECT} : (\nabla\texttt{Broken}@x \ \sqcap \ \nabla\texttt{Working}@y))\end{aligned}$$

`Fix` denotes, according to the definition, any action occurring at some interval involving an $\star$`OBJECT` that was once `Broken` and then `Working`. The $\sharp$ interval could be understood as the occurring time of the action type being defined: referring to it within the definition is an explicit way to temporally relate states and actions occurring in the world with respect to the occurrence of the action itself. The variables x, y are existentially introduced by the operator $\Diamond$. The temporal constraints $(x \ \mathsf{m} \ \sharp)$ and $(\sharp \ \mathsf{m} \ y)$ state that the interval denoted by x should meet the interval denoted by $\sharp$ – the occurrence interval of the action type `Fix` – and that $\sharp$ should meet y. Figure 2 shows the temporal dependencies of the intervals in which the concept `Fix` holds. The parametric feature $\star$`OBJECT` plays the role of *formal* parameter of the action, mapping any individual action of type `Fix` to the object to be `Fix`ed, independently from time. Please note that, whereas the existence and identity of the $\star$`OBJECT` of the action is time invariant, it can be qualified differently in different intervals of time, e.g., the $\star$`OBJECT` is necessarily `Broken` only during the interval denoted by x and its subintervals.

The assertion $\texttt{Fix}(i, a)$ says that a is an individual action of type `Fix` occurred at the interval i. Moreover, the same assertion implies that a is related to a $\star$`OBJECT`, say b, which is of type `Broken` at some interval j – meeting i – and at all intervals included in j, and of type `Working` at another interval l – met by i – and at all intervals included in l:

$$\begin{aligned}&\texttt{Fix}(i, a) \Longrightarrow\\ &\quad \exists b.\ \star\texttt{OBJECT}(a, b) \ \wedge\\ &\qquad \exists\, j, l.\ \forall\, \hat{j}, \hat{l}.\ \mathsf{m}(j, i) \wedge \mathsf{m}(i, l) \wedge\\ &\qquad\qquad (=, \mathsf{s}, \mathsf{d}, \mathsf{f})(\hat{j}, j) \wedge (=, \mathsf{s}, \mathsf{d}, \mathsf{f})(\hat{l}, l) \wedge\\ &\qquad\qquad \texttt{Broken}(\hat{j}, b) \wedge \texttt{Working}(\hat{l}, b).\end{aligned}$$

It is interesting to notice that the `Fix` action subsumes the following `Instant-Fix` action, whose temporal dependencies are depicted in figure 3:

$$\begin{aligned}&\texttt{Instant-Fix} \ \doteq\\ &\quad \Diamond(z \ y) \ (\sharp \ \mathsf{f} \ z)(\sharp \ \mathsf{m} \ y).\\ &\qquad (\star\texttt{OBJECT} : (\nabla\texttt{Broken}@z \ \sqcap \ \nabla\texttt{Working}@y))\end{aligned}$$

Subsumption holds because the class of intervals – obtained by homogeneity of the state `Broken` in the definition of `Fix` – including x and all its subintervals, is a subset of the class of intervals over which the object is known to be broken, according to the definition of `Instant-Fix`; this class includes all the subintervals of z.

Please note that, if the `Instant-Fix` action had been defined without the ∇ operator, then it would not specialize any more the `Fix` action. In fact, according to the weaker definition of `Instant-Fix`, specifying that the object is broken at z does not imply that the object is broken at subintervals of z; in particular, we can not deduce any more that the object is broken at x and its subintervals, as specified in the definition of `Fix` action. For the same reason, the *weak* `Instant-Fix` action type would not specialize the *weak* `Fix` action type, too. Thus, homogeneity helps us to define states and actions in a more accurate way, such that important inferences are captured.

3 Calculus for Subsumption

In [Artale and Franconi,1994a] it is shown that subsumption between temporal concepts in this logic, without homogeneous concepts, is decidable. We remind that subsumption in a propositionally complete description logic with both existential and universal temporal quantification is undecidable and it is still not clear if it becomes decidable in absence of negation [Bettini,1993a]. The homogeneity operator is a restricted form of universal quantification and we are particularly interested in an even more restricted form, where the concept C in ∇C does not contain any other temporal operator (we call it *simple homogeneous concept*). The expressiveness of the resulting logic is enough, for example, to correctly represent the homogeneous nature of states. We have developed an algorithm to compute subsumption in this logic (see also [Bettini,1993c]). Even if a formal proof is still not available, there are good arguments to conjecture its completeness with respect to the usual definition of subsumption (see [Artale and Franconi,1994a] for its temporal formulation). This would also prove decidability of this logic and of corresponding modal logics.

We first compute a *normal form* for concepts: each temporal concept is normalized and transformed into an equivalent minimal temporal constraint network of the form:

$$\diamond(\overline{X})\ \overline{Tc}.\ (Q^1@X^1 \sqcap \ldots \sqcap Q^n@X^n)$$

where Q^i are conjunctions of non-temporal and simple homogeneous concepts ($Q^i = L^i \sqcap \nabla M^i$, with L^i, M^i non-temporal concepts), while arcs are labeled with temporal relations. Each normalized conceptual temporal constraint network is computed in such a way that it has some interesting properties: temporal constraints are always explicit, computing the transitive closure of the Allen temporal relations obtaining the minimal temporal network (see e.g. [van Beek and Cohen,1990]); the information in each node is independent from the information in the other nodes; the network is minimal wrt time-invariant information, maintained only in the $\sharp$ node of the graph.

In absence of the homogeneity operator, concept subsumption in the temporal language is reduced to concept subsumption between non-temporal concepts and to subsumption between temporal constraint networks. A similar general procedure was first presented in [Weida and Litman,1992], where the language is less expressive. Algorithms to compute subsumption between non-temporal concepts are well known, see e.g. [Hollunder *et al.*,1990]. An exhaustive search is done to find a mapping between each node in the potential subsumer and some node in the subsumee, such that subsumption between concepts labeling corresponding nodes holds, while temporal relations labeling arcs between corresponding nodes have to be less specific in the potential subsumer (with $R_1 \subseteq R_2$, we indicate that R_1 is a more restricted set of temporal constraints than R_2).

Definition (S-mapping)
A *s-mapping* from a conceptual temporal constraint network $\langle \overline{X}, \overline{Tc}_1, \overline{Q@X} \rangle$ to a conceptual temporal constraint network $\langle \overline{Y}, \overline{Tc}_2, \overline{P@Y} \rangle$ is a total function $\mathcal{S} : \overline{X} \mapsto \overline{Y}$ such that $\mathcal{S}(\sharp) = \sharp$, and the concept labeling each node x in $\overline{X}$ subsumes the concept labeling the corresponding node $\mathcal{S}(x)$, and for each temporal constraint $(X_1\ R_1\ X_2) \in \overline{Tc}_1$ there exists a temporal constraint $(Y_1\ R_2\ Y_2) \in \overline{Tc}_2$ which satisfies $\mathcal{S}(X_1) = Y_1$, $\mathcal{S}(X_2) = Y_2$ and $R_1 \supseteq R_2$. □

The following proposition introduces a procedure for checking subsumption between conceptual temporal constraint networks.

Proposition 1 *A conceptual temporal constraint network $C = \langle \overline{X}, \overline{Tc}_1, \overline{Q@X} \rangle$ subsumes a conceptual temporal constraint network $D = \langle \overline{Y}, \overline{Tc}_2, \overline{P@Y} \rangle$, written $C \sqsupseteq D$, if and only if there exists a s-mapping from C to D.*

Proof. See [Artale and Franconi,1994a]. □

In order to compute subsumption in presence of homogeneity, we must first define a simpler form of temporal subsumption, to test when a concept associated to a node subsumes a concept associated to another node. Since each Q^i, in the normal form, can be expressed by the conjunction $L^i \sqcap \nabla M^i$, we extend the non-temporal subsumption algorithm to test whether $(L \sqcap \nabla M) \sqsupseteq (N \sqcap \nabla O)$.

Proposition 2 *Let L, M, N, O be non-temporal concepts:*
$(L \sqcap \nabla M) \sqsupseteq (N \sqcap \nabla O)$ iff $L \sqsupseteq (N \sqcap O)$ and $M \sqsupseteq O$.

Proof. See [Bettini,1993c]. □

Proposition 2 shows how this simple form of temporal subsumption can be reduced to non-temporal subsumption, for which algorithms are known.

In order to take into account homogeneity, new variables in the subsumee D – representing intervals of time *contained* into intervals explicitly present in D – should be considered. These are actually new nodes in the conceptual temporal constraint network, and their associated concepts are inherited from the nodes that temporally contain them. Since we have a dense temporal domain we can have an infinite number of these nodes. However, many of them will have the same qualitative temporal relations with the intervals present in D and identical associated concepts. As a matter of fact, there is only a finite number of significative groups of these intervals. We can consider one new node for each group. Each new node must be an image of a node in the subsumer, C, for which does not exist an exact mapping. Its temporal constraints with other nodes in D, that are images of nodes in C, must be tighter or equal with respect to the corresponding constraints in C. Moreover, the node will be useful to detect subsumption only if the interval associated to it is contained in one or more intervals associated to nodes in D. In fact, only from these nodes it could inherit some homogeneous concept.

The algorithm, illustrated in figure 4, implements a procedure that find a *S-mapping*, modified as discussed above in order to take into account homogeneous concepts. The first time, it tries to find a mapping without introducing new nodes in the subsumee – it can be used for computing subsumption in the absence of homogeneity, too. The function FIND-MAP, after trying a *direct* mapping without success, calls TRY-HOM to check if the subsumption can be proved considering homogeneous properties of concepts. TRY-HOM is illustrated in figure 5; it considers the fact that each node in the subsumee, labeled with a homogeneous concept, can be expanded generating new nodes. While, in principle, their number is infinite, the number of *useful* ones is finite and a mapping is found if and only if the subsumption holds. The TRY-HOM function succeeds if:

- the added node does not modify the temporal constraints among the original variables of D – as computed in HOM-TEST;
- the new node inherits homogeneous concepts from other nodes, in such a way that the concept associated to the unmatched node of the subsumer network could subsume the concept associated with the new node.

The function HOM-TEST, used by TRY-HOM, takes as arguments, respectively, a subset of nodes, a network and the new node. It restricts each relation between the new node and nodes in the subset to include only relations in {s,d,f}. For example, {o,s,d} is restricted to {s,d}. This restriction is necessary to inherit homogeneous concept: the node must be necessarily *internal*. Finally the constraints are propagated. The test succeeds if the restricted relations are *compatible* with the original network, and the relations involving nodes in the original network are not restricted – otherwise we would be checking subsumption of a different network, not necessarily subsumed by the original one.

Particular attention must be paid to the temporal relations that we enforce on the nodes that we add to D. This fact is reflected in the TEMP-SUBS function. This function is called when a mapping between a node of the subsumer and an existing node of the subsumee is tried. This node can be an original node of D or an internal node added in a previous step of the algorithm. TEMP-SUBS returns a network. If the node is an original node of D, it must return a nonzero value iff temporal constraints in C subsume the corresponding constraints in D, under the partial mapping, *Map*, augmented with the new mapping – $(x_i \rightarrow y_j)$. If the node is an added node, we just check that the *intersection* of corresponding temporal constraints is not empty. If this intersection restricts the constraints in the subsumee network,

HOM-SUBS(C,D)
/*
* $C = \langle \overline{X}, \overline{Tc}_1, \overline{Q@X} \rangle$
* where $\overline{Q@X} = Q^0@\sharp \sqcap Q^1@x_1 \sqcap \ldots \sqcap Q^n@x_n$
* $D = \langle \overline{Y}, \overline{Tc}_2, \overline{P@Y} \rangle$
* where $\overline{P@Y} = P^0@\sharp \sqcap P^1@y_1 \sqcap \ldots \sqcap P^m@y_m$
*/

- IF $Q^0 \sqsupseteq P^0$
  THEN subs $\leftarrow$ FIND-MAP(($\sharp \rightarrow \sharp$),$D$,$x_1$)
  ELSE subs $\leftarrow$ False

RETURN(subs).

FIND-MAP(Map,Net2,x_i)

- success $\leftarrow$ False ; $j \leftarrow 1$
- WHILE (y_j in $\overline{Y}$) AND NOT(success) DO
  - Newnet2 $\leftarrow$ TEMP-SUBS(Map,Net2,x_i,y_j)
  - IF Newnet2 AND ($Q^i \sqsupseteq P^j$) THEN
    - NewMap $\leftarrow$ Map $\cup$ $\{x_i \rightarrow y_j\}$
    - IF $i = n$ THEN success $\leftarrow$ NewMap
      ELSE success $\leftarrow$
      FIND-MAP(NewMap,Newnet2,x_{i+1})
  - $j \leftarrow j + 1$
- IF NOT(success) THEN
  /* There is no direct mapping with variables in D
  * Try to exploit homogeneity properties */
  success $\leftarrow$ TRY-HOM(Map,Net2,x_i)

RETURN(success).

TEMP-SUBS(Map,Net2,x_i,y_j)

- Temp-net $\leftarrow$ Net2
- WHILE ((xRx_i) $\in C.\overline{Tc}_1$) AND (Temp-net) DO
  IF ($x \rightarrow y$) $\in$ Map THEN
  IF ($y \in D.\overline{Y}$ AND ($yR'y_j$) $\in D.\overline{Tc}_2$
  AND NOT ($R \supseteq R'$))
  /* y is one of the original nodes in D*/
  THEN Temp-net $\leftarrow$ 0 /* infeasible mapping */
  ELSE IF ($y \notin D.\overline{Y}$ AND ($yR'y_j$) $\in$ Net2.$\overline{TC}$)
  THEN /* y is one of the added nodes */
  – $R' \leftarrow R \cap R'$
  – IF ($R' = \emptyset$) THEN Temp-net $\leftarrow$ 0
  ELSE Temp-net $\leftarrow$ Propagate(Net2)

RETURN(Temp-net).

Figure 4: An algorithm to compute subsumption in the temporal language extended with the homogeneity operator.

TRY-HOM(Map,Net2,x_i)

- success $\leftarrow$ False
- Newnet2 $\leftarrow$ ADDNODE(Net2,$\overline{y}$,$\overline{x}_i$,Map)
- Newnet2 $\leftarrow$ Propagate(Newnet2)
- IF Newnet2 THEN
  - Let $Y = \{y \in D.\overline{Y} : \overline{y}Ry \in$ Newnet2.$\overline{TC}$ AND (s,d,f)$\cap R \neq \emptyset)\}$
  - IF $Y \neq \emptyset$ THEN FOR each subset SY of Y DO
    - ∗ Let $P@\overline{y} \in$ Newnet2.$\overline{P@Y}$
    - ∗ IF $Q^i \sqsupseteq P$ THEN
      - · Tempnet $\leftarrow$ HOM-TEST(SY,Newnet2,$\overline{y}$)
      - · IF Tempnet THEN
        - \+ NewMap $\leftarrow$ Map $\cup \{x_i \rightarrow \overline{y}\}$
        - \+ Newnet2 $\leftarrow$ Tempnet
        - \+ IF $i = n$ THEN success $\leftarrow$ NewMap
          ELSE success $\leftarrow$
          FIND-MAP(NewMap,Newnet2,x_{i+1})

RETURN(success).

ADDNODE(Net,$\overline{y}$,$\overline{x}_i$,Map)

- Newnet.$\overline{Y} \leftarrow$ Net.$\overline{Y} \cup \{\overline{y}\}$
- Newnet.$\overline{TC} \leftarrow$ Net.$\overline{TC} \cup \{(yR\overline{y}) : \exists x_k\ (x_k \rightarrow y) \in$ Map AND $(x_k R x_i) \in C.\overline{Tc}_1\}$
- Newnet.$\overline{P@Y} \leftarrow$ Net.$\overline{P@Y} \cup \{\top@\overline{y}\}$

RETURN(Newnet).

HOM-TEST(Z,Net,$\overline{y}$)

- FOR each $\overline{y}Rz_i \in$ Net.$\overline{TC}$ with $z_i \in Z$ DO
  $R \leftarrow R \cap \{\mathsf{s}, \mathsf{d}, \mathsf{f}\}$
- success $\leftarrow$ Propagate(Net)
- IF success THEN /**The network is consistent**/
  FOR each $y_i R' y_j \in D.\overline{Tc}_2$ with $y_i, y_j \in D.\overline{Y}$ DO
  IF $R \subset R'$ where $y_i R y_j \in$ Net.$\overline{TC}$ THEN
  success $\leftarrow$ Fail

RETURN(success).

Figure 5: A sketch of the TRY-HOM algorithm.

this restriction must be considered in further steps of the algorithm.

References

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

[Artale and Franconi, 1993] Artale, Alessandro and Franconi, Enrico 1993. A unified framework for representing time, actions and plans. In Anger, F. D.; Guesgen, H. W.; and van Benthem, J., editors 1993, *Workshop Notes of the IJCAI Workshop on Temporal and Spatial Reasoning*, Chambery, France. 193–217. Also in the Workshop Notes of the IJCAI-93 Workshop on Object-Based Representation System; a shorter version appears in the Workshop Notes of the Italian 1993 Workshop on Automatic Planning, Roma Italy, September 1993.

[Artale and Franconi, 1994a] Artale, Alessandro and Franconi, Enrico 1994a. A computational account for a description logic of time and action. In *Proc. of the 4 th International Conference on Principles of Knowledge Representation and Reasoning*, Bonn, Germany.

[Artale and Franconi, 1994b] Artale, Alessandro and Franconi, Enrico 1994b. Time, actions and plans representation in a description logic. *International Journal of Intelligent Systems.* To appear.

[Artale, 1994] Artale, Alessandro 1994. *Rappresentazione di Tempo ed Azioni e Ragionamento Tassonomico per Object-Oriented Databases nel Contesto dei Linguaggi Terminologici.* Ph.D. Dissertation, University of Florence, Italy. (in Italian).

[Bettini, 1992] Bettini, Claudio 1992. A formalization of interval-based temporal subsumption in first order logic. In *Workshop Notes of the ECAI Workshop on Theoretical Foundations of Knowledge Representation and Reasoning*, Vienna, Austria.

[Bettini, 1993a] Bettini, Claudio 1993a. *Temporal Extensions of Terminological Languages.* Ph.D. Dissertation, Computer Science Department, University of Milano, Italy. (in Italian).

[Bettini, 1993b] Bettini, Claudio 1993b. A family of temporal terminological logics. In *Proc. of the 3 rd Congress of the Italian Association for Artificial Intelligence, Torino, Italy*, Lecture Notes in Artificial Intelligence, N.728, pages 120–131. Springer Verlag, 1993.

[Bettini, 1993c] Bettini, Claudio 1993c. Computing subsumption between action descriptions. Technical Report 102-93, University of Milano, Computer Science Dept.,Italy.

[Hollunder *et al.*, 1990] Hollunder, B. and Nutt, W. and Schmidt-Schauß, M. 1990. Subsumption algorithms for concept description languages. In *Proc. of ECAI-90*, Stockholm, Sweden. 348-353.

[Schmiedel, 1990] Schmiedel, A. 1990. A temporal terminological logic. In *Proc. of AAAI-90*, Boston, MA. 640-645.

[van Beek and Cohen, 1990] van Beek, P. and Cohen, R. 1990. Exact and Approximate Reasoning about Temporal Relations. In *Computational Intelligence*, 6:132-144.

[Weida and Litman, 1992] Weida, Robert and Litman, Diane 1992. Terminological reasoning with constraint networks and an application to plan recognition. In *Proc. of the 3 rd International Conference on Principles of Knowledge Representation and Reasoning*, Cambridge, MA. 282–293.