F2), F2-F1 and F1∩F2. The Qualitative Relations Manager is simply our implementation of Allen's operations of inversion, intersection and composition of Interval Algebra relations. Finally, the third module implements our operations of inversion (not described in the paper), check-intersection and composition.

Given an I-Time definition, one could ask the temporal location of the instances of the I-Time in a given Frame-Time (e.g., one could want in output the list of all Mondays between 1-1-90 and 1-6-90). This operation is managed by the Unfolding Manager. Besides these queries, TeMP also deals with queries concerning the KB of periodic events specifications. For example, one can ask which are the temporal relations between two periodic events ev1* and ev2*, in a given Frame-Time F. In such a case, TeMP gives as output the list of temporal specifications of the form F'∩F ev1* EACH C R ev2* in the KB of periodic events where F' is a Frame-Time intersecting F.

TeMP has been implemented in Quintus Prolog and runs on Sun workstations, under Unix.

8 Conclusions

The temporal framework sketched in this paper constitutes an integration of part of the works developed by the two mainstreams of research about periodic events in AI and TDB, extending current approaches to deal with both user-defined I-Times and qualitative temporal relations concerning periodic events. Although we believe that our framework is significantly more powerful and expressive than the other approaches dealing with periodic events in the AI and TDB literature, many other aspects have to be taken into account in order to obtain a comprehensive approach to periodic events. In particular, we are currently investigating the possibility of extending our framework for dealing with quantifiers such as "only", "sometimes" etc. in [Morris et al., 93] and with partial relations between I-Times.

This work was partially supported by the Italian CNR, project "Ambienti e strumenti per la gestione di informazioni temporali".

References

[Allen, 83] J.F. Allen: "Maintaining Knowledge about Temporal Intervals", Comm ACM 26(11), 832-843 (1983).

[Allen, 91] J. Allen: "Time and Time again: the Many Ways to Represent Time", International Journal of Intelligent Systems 6(4), 341-355 (1991).

[Baudinet et al., 93] M. Baudinet, J. Chomicki, P. Wolper: "Temporal Deductive Databases", in A. Tansell, R. Snodgrass, J. Clifford, S. Gadia, A. Segev (eds): Temporal Databases: Theory, Design, and Implementation, Benjamin-Cummings (1993).

[Chandra and Segev, 93] R. Chandra, A. Segev: "Managing Temporal Financial Data in an Extensible Database", Proc. 19th International Conference on Very Large Databases (1993).

[Chomicki and Imielinsky, 88] J. Chomicki, T. Imielinsky: "Temporal Deductive Databases and Infinite Objects", Proc. ACM Symposium on Principles of Database Systems, 61-73 (1988).

[Kabanza et al., 90] F. Kabanza, J.-M. Stevenne, P. Wolper: "Handling Infinite Temporal Data", Proc. ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, 392-403 (1990).

[Ladkin, 86a] P. Ladkin: "Primitive and Units for Time Specification", Proc. AAAI'86, 354- 359 (1986).

[Ladkin, 86b] P. Ladkin: "Time Representation: A Taxonomy of Interval Relations", Proc. AAAI'86, 360-366 (1986).

[Leban et al., 86] B. Leban, D.D. McDonald, D.R. Forster: "A representation for collections of temporal intervals", Proc. AAAI'86, 367-371 (1986).

[Ligozat 91] G. Ligozat: "On Generalized Interval Calculi", Proc. AAAI'91, 234-240 (1991).

[Morris et al., 93] R.A. Morris, W.D. Shoaff, L. Khatib: "Path Consistency in a Network of Non-convex Intervals", Proc. IJCAI'93, 655-660 (1993).

[Niezette and Stevenne, 92] M. Niezette, J.-M. Stevenne: "An Efficient Symbolic Representation of Periodic Time", Proc. First International Conference on Information and Knowledge Management (1992).

[Poesio, 88] M. Poesio: "Toward a Hybrid Representation of Time", Proc. ECAI'88, 247-252 (1988).

[Soo and Snodgrass, 92] M. Soo, R. Snodgrass: "Mixed Calendar Query Language Support for Temporal Constraints", Tech. Rep. TempIS No.29, University of Arizona (1992).

[Soo, 93] M. Soo: "Multiple Calendar Support for Conventional Database Management Systems", Proc. Int. Workshop on an Infrastructure for Temporal Databases, (1993).

[Stonebraker, 90] M. R. Stonebraker: Chapter 7: Extensibility. Readings in Database Systems, M.R. Stonebraker (editor), Morgan Kaufman (1990).

[Terenziani, 95] P. Terenziani: "Integrating calendar-dates and qualitative temporal constraints in the treatment of periodic events", Tech. Report 12-95, Dipartimento di Informatica, Universita' di Torino (1995).

[Van Eynde, 87] F. Van Eynde: "Iteration, Habituality and Verb Form Semantics", Proc. 3rd Conf. of European Chapter of the Association for Computational Linguistics, 270-277 (1987).

Property 2. Our operation of check-intersection does not loose information (this is obvious for composition, which only adds new pieces of information).

Property 1 has been proved by showing that, for each case in the definitions of check-intersection and composition, the logical description of the "antecedent" part of the definition (including also the axioms formalising the relation between the I-Times being considered) implies the logical description of the "consequent" part. For example, we proved the correctness of our definition of composition in the case where the relation between the two I-Times being composed is $\in$ by proving that

ev1* EACH C1* R1 ev2* $\wedge$
ev1* EACH C2* R2 ev3* $\wedge$ C1* $\in$ C2* $\rightarrow$
ev2* EACH C1* R ev3*

where R is the composition of R1 and R2 in Allen's Interval Algebra and each one of the specifications has been replaced by the logical axioms describing its meaning.

Property 2 has been proved by showing that, for each case in the definition of check intersection, the "consequent" part of the definition, plus the axioms formalising the relation between the I-Times being considered imply the "antecedent" of each definition. For instance, in the case where the relation between the two I-Times being composed is temporal equality, we proved that

ev1* EACH C1* R ev2* $\wedge$ C1* $=^T$ C2* $\rightarrow$
ev1* EACH C1* R1 ev2* $\wedge$
ev1* EACH C2* R2 ev2*

(where R is the intersection of R1 and R2, in Allen's Interval Algebra).

It is important to notice that Properties 1 and 2 grant that our operations of check-intersection and composition can be regarded as a *compilation* of a set of logical inferences that could also be performed (in a less efficient way), e.g, by a theorem prover for the first order logic. Moreover, as a consequence of these properties, Corollary 1 holds.

Corollary 1. PCforPE is correct and does not loose information (in fact, PCforPE is a path-consistency algorithm repeatedly applying check-intersection and composition).

The temporal logic and the proofs are not reported in this paper for the sake of brevity, and are presented in [Terenziani, 95].

7 The TeMP system

The TeMP system (**Te**mporal **M**anager of **P**eriodic events) has been realised on the basis of the approach described in this paper. TeMP is a general purpose temporal manager dealing with user-defined calendric definitions (I-Times) and temporal specifications about periodic events. The architecture of TeMP is shown in figure 2: boxes denote modules and ovals represent data.

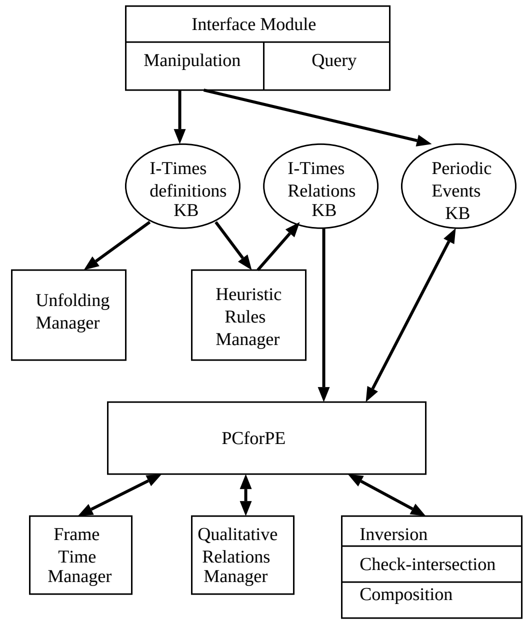

Figure 2: Architecture of the TeMP system

The Interface Module manages the interaction with the user, allowing the insertion/deletion of I-Times definitions and temporal specifications about periodic events (using the formalism described in section 2), as well as queries. Insertions and deletions build up the knowledge base (KB) of Periodic Events specifications and the KB of I-Times definitions.

The Heuristic Rules Manager is an auxiliary module which operates on the definitions of I-Times provided by the user and gives as output the relation holding between each pair of I-Times, on the basis of a set of heuristic rules (see section 4).

The basic reasoning module is PCforPE, which is based on the algorithm shown in section 5. PCforPE takes in input the KB of temporal specifications about periodic events and the KB of temporal relations between I-Times, and check the consistency of the temporal specifications and infers new specifications. In order to operate, the PCforPE module takes advantage of three different modules, which operate on the different components of our temporal specifications (i.e., <Frame>, <Qual-Rel> and <I-Time> in the syntax (S)). The Frame Time Manager takes in input two Frame-Times F1 and F2 and provides as output three (possibly empty) sets of Frame-Times: F1-F2 (i.e., the difference between F1 and

- C1 $\in$ C2 (the inverse holds if C2 $\in$ C1)

  The result of the composition is ev2* Q C1 R ev3*. Notice that C1 is the I-Time for the new specification between ev2* and ev3*. For example, the composition of (s11) and (s12) is (s13).

- C1 $\in^{inc}$ C2 (the inverse holds if C2 $\in^{inc}$ C1)

  No new specification in our formalism can be inferred (the motivation is analogous to that discussed when dealing with $\prec$). For example:

  a* EACH Christmas* (BEFORE,MEETS) b* @
  a* EACH Months* (AFTER) c* ->
  NO NEW TEMPORAL SPECIFICATION

- C1 $\subset$ C2 (the same holds if C2 $\subset$ C1)

  An inconsistency is reported, since it cannot be the case that ev1* happens exactly once both in C1 and in C2. For example, consider again the composition of (s12) and (s14).

- C1 $\sharp$ C2 (C1 and C2 are not comparable)

  No new qualitative relation can be inferred. For example:

  a* EACH Mondays* (BEFORE,MEETS) b* @
  a* EACH Tuesdays* (AFTER) c* ->
  NO NEW TEMPORAL SPECIFICATION

5.3 Reasoning Process

We developed an extension of Allen's path consistency algorithm [Allen, 83] in order to reason with a knowledge base of temporal specifications in our formalism. Since path-consistency is not complete for Allen's Interval Algebra, a-fortiori it is not complete for our temporal specifications (which include the Interval Algebra for specifying the qualitative constraints): as in many AI approaches dealing with the Interval Algebra (see, e.g., the survey in [Allen, 91]), we chose to loose the completeness of the reasoning process in order to retain tractability.

Differently from [Allen, 83], we must consider multiple temporal specifications relating the same pair of events. We denote as S_a(ev1*, ev2*) a temporal specification relating ev1* and ev2*; "S" is indexed in order to distinguish among different specifications relating ev1* and ev2*. Figure 1 sketches our path consistency algorithm for periodic events (PCforPE). In PCforPE, Paths(S_a(ev1*, ev2*)) contains, for each event evh* (ev1* $\neq$ evh* $\neq$ ev2*), all the specifications relating ev1* and evh* -i.e., S_i(ev1*, evh*), for all i and for all event evh*-). STACK is a stack containing all the new specifications. Before the execution of PCforPE, all the input specifications are pushed onto STACK. Both check-intersection and composition may generate new specifications, which are pushed onto the STACK, in order to propagate the new constraints they convey. The algorithm stops when the stack is empty or when an inconsistency is reported (by check-intersection or composition).

```
FORALL specification Sa(ev1*, ev2*) in STACK DO
   POP Sa(ev1*, ev2*) from STACK
   FORALL specification Si(ev1*, ev2*) DO
      X <- Sa(ev1*, ev2*) ⊓ Si(ev1*, ev2*)
      PUSH the new specifications in X
         (if any) onto STACK OD
   FORALL specification Si(ev1*, evh*)
   in Paths(Sa(ev1*, ev2*)) DO
      X <- Sa(ev1*, ev2*) @ Si(ev1*, evh*)
      PUSH the new specifications in X
         (if any) onto STACK OD OD
```

Figure 1: PCforPE algorithm

Check-intersection and composition do not generate either new bounds (different from the upper and lower bounds of the input Frame-Times) nor new I-Times. Let H and K be the number of bounds and I-Times introduced by the user in the input specifications. Thus, at most O($H * K$) specifications may hold between the same pair of periodic events. Thus, PCforPE considers considers O($H * K * N^2$) different temporal specifications (where N is the number of events in the knowledge base). Each of them can be pushed onto STACK at most 13 times (due to the fact that an ambiguous qualitative relation is at most a disjunction of 13 basic Interval relations, so that it can be reduced at most 13 times [Allen, 83]). Whenever a specification S is pushed onto the stack, PCforPE performs at most O($H * K$) check-intersections and O($H*K*N$) (the cardinality of Paths(S)) compositions. Thus, PCforPE operates in polynomial time, performing O($H^2 * K^2 * N^2$) check-intersections and O($H^2 * K^2 * N^3$) compositions.

6 Temporal logic and properties of the approach

We provided a logical formalization for the different components of our temporal approach and, besides the others, for

(i) the basic notions of correlation and association,
(ii) the temporal specifications in our formalism,
(iii) the relations between I-Times (e.g., $\prec$).

On the basis of our first order temporal logic for periodic events we proved the following properties:

Property 1. Our operations of check-intersection and composition are correct (where Q is the quantifier EACH, C1 and C2 are I-Times and R1 and R2 are qualitative relations in Allen's Interval Algebra) in the intersection [d1', d1"] ∩ [d2', d2"] of the Frame-Times is defined by cases (in the following, R indicates the intersection between the Interval Algebra relations R1 and R2):

- C1 $=^T$ C2
  If R is the null relation, an inconsistency is reported. Otherwise, the result of check-intersection is ev1* Q C1 R ev2*. For example:
  a* EACH Days* (BEFORE,MEETS) b* $\underline{\cap}$
  a* EACH Days* (BEFORE,OVERLAPS) b*
  –> a* EACH Days* (BEFORE) b*

- C1 $\prec$ C2 (the inverse holds if C2 $\prec$ C1)
  If R is empty, an inconsistency is reported. Otherwise, the result of check-intersection is ev1* Q C1 R ev2* and ev1* Q C2 R ev2* (the intersection of the qualitative temporal relations is selected in the output specification). E.g.,
  a* EACH Mondays* (BEFORE) b* $\underline{\cap}$
  a* EACH Weeks* (BEFORE,MEETS) b* –>
  a* EACH Mondays* (BEFORE) b*,
  a* EACH Weeks* (BEFORE) b*

  Notice that both specifications must be provided in output, since they conjunctively convey the information that a* and b* occur exactly once each Weeks*, and, more specifically, in the Mondays* part of Weeks*.

- C1 $\in$ C2 (the inverse holds if C2 $\in$ C1)
  If R is empty, an inconsistency is reported. Otherwise, the result of check-intersection is ev1* Q C1 R ev2* and ev1* Q C2 R2 ev2*. In fact, the qualitative relations holding in the restricted I-Time must be forced to be compatible those holding in general. For example, since Mondays* $\in$ Days*, the result of check-intersection on (s8) and (s7), in the overlapping part of the Frame-Times (i.e., in [1-6-91, 1-1-92]) is the pair of specifications (s10) and (s11) above.

- C1 $\in^{inc}$ C2 (the inverse holds if C1 $\in^{inc}$ C1)
  If R is empty, an inconsistency is reported. Otherwise, the result is ev1* Q C1 R ev2* and ev1* Q C2 R2 ev2*. For example:
  a* EACH Christmas* (AFTER,STARTS) b* $\underline{\cap}$
  a* EACH Months* (AFTER,MEETS) b* –>
  a* EACH Christmas* (AFTER) b*,
  a* EACH Months* (AFTER,MEETS) b*

- C1 $\subset$ C2 (the same holds if C2 $\subset$ C1)
  In such a case an inconsistency is reported. For example:
  a* EACH Days* (BEFORE,MEETS) b* $\underline{\cap}$
  a* EACH Months* (BEFORE) b*
  gives an inconsistency, since a* cannot happen exactly once a day and exactly once a month.

- C1 $\sharp$ C2
  The temporal specifications provide two different constraints between the same pair of periodic events, holding at "incomparable" I-times. In such a case no new qualitative constraint can be inferred, and the input constraints are left unchanged (with the implicit meaning that both of them must hold in the intersection -if any- of the instances of the two I-Times). For example:
  a* EACH Mondays* (MEETS) b* $\underline{\cap}$
  a* EACH Tuesdays* (AFTER) b* –>
  INPUT SPECIFICATIONS UNCHANGED

5.2 Composition

Composition (@) applies to two specifications of periodic events as in (s16)

(s16) [d1', d1"] ev1* Q C1 R1 ev2* @
[d2', d2"] ev1* Q C2 R2 ev3*

No new information can be inferred as regards the non-intersecting parts of the two Frame-Times. In the time interval [d1', d1"] ∩ [d2', d2"] (if any), the result of composition depends on which relation holds between the I-Times C1 and C2 (in the following, R represents the composition, in Allen's Interval Algebra, of the inverse of R1 with R2).

- C1 $=^T$ C2
  The composition of the two specifications is ev2* Q C1 R ev3*. For example:
  a* EACH Days* (BEFORE,MEETS) b* @
  a* EACH Days* (AFTER) c* –>
  b* EACH Days* (AFTER) c*

- C1 $\prec$ C2 (the inverse holds if C2 $\prec$ C1)
  In such a case, no new specification in our specification formalism can be inferred. For instance,
  a* EACH Mondays* (BEFORE, MEETS) b* @
  a* EACH Weeks* (AFTER) c*
  –> NO NEW SPECIFICATION
  Notice that it would not be correct to infer either (i) b* EACH Mondays* (AFTER) c* or (ii) b* EACH Weeks* (AFTER) c*. In fact, (i) corresponds to arbitrarily assume that c* happens each Mondays*, while (ii) corresponds to arbitrarily assume that b* happens once a week (while we only have that it happens once each Mondays*, so that it could happen also, e.g., on Tuesdays). Thus, neither (i) nor (ii) are implied by the input specifications.

Thus, we cannot propose a compact definition of intersection and composition such as in [Allen, 83], [Morris et al., 93]. On the other hand, we have to point out a set of basic relations between I-Times (see section 4) and to propose a definition by cases of check-intersection and composition, on the basis of the relation holding between the I-Times in the specifications.

4 Relations between I-Times

Given two I-Times C1* and C2*, since we use the quantifier EACH ("exactly once each") in the temporal specifications, we are interested in the cases where for each instance of C1* there is just a related instance of C2* and vice versa (bijective relation between instances of C1* and of C2*) or in the cases where for each instance of one of the two I-Times (say C1*) there is just a related instance of the other I-Time (say C2*) but not viceversa. The treatment of the partial relations which do not cover all instances of at least one of C1* and C2* requires an extension of our formalism which is discussed in [Terenziani, 95]. Moreover, the cases to be considered for temporal reasoning are those in which C1* and C2* allow one to refer to the same instances of a periodic event, i.e., those cases where there is a relation of temporal containment between the corresponding instances of C1* and C2* (C1* EQUAL C2* is a special case of temporal containment, which must be distinguished since it allows one to draw further inferences). The disjoint relations $=^T$, $\prec$, $\in$ and $\in^{inc}$ (plus inverses) cover these cases. Given two I-Times C1* and C2*,

- C1* $=^T$ C2* (read as: C1* and C2* are temporally equal) iff there is a bijection between instances of C1* and instances of C2*, and Allen's relation EQUAL holds between each pair of corresponding instances.
- C1* $\prec$ C2* (C1* is more specific than C2*) iff for each instance of C1* there is exactly one instance of C2* which properly contains it and, conversely, for each instance of C2* there is exactly one instance of C1* which is properly contained in it (bijection) (e.g., Mondays* $\prec$ Weeks*).
- C1* $\in$ C2* (C1* is a restriction of C2*) iff for each instance of C1* there is an instance of C2* which is temporally equal to it, but not vice versa (e.g., Mondays* $\in$ Days*).
- C1* $\in^{inc}$ C2* (C1* is an inclusion restriction of C2*) iff for each instance of C1* there is an instance of C2* which properly contains it, but not vice versa (e.g., Christmas* $\in^{inc}$ Months*).

Besides these relations, it is important to introduce two further relations.

- C1* $\subset$ C2* (C1* is more frequent than C2*) characterises the cases where two assertions such as (i) "$event_x$ happens exactly once each C1*" and (ii) "$event_x$ happens exactly once each C2*" are inconsistent in a given Frame-Time I. Roughly speaking, this happens when, in any way we choose a time interval in each instance of C1* in I, at least two of these time intervals intersect the same instance of C2* (e.g., Days* $\subset$ Weeks*; see [Terenziani, 95] for a formal definition of $\subset$ and of the relations $=^T$, $\prec$, $\in$ and $\in^{inc}$).
- C1* $\sharp$ C2* (C1* and C2* are temporally incomparable) iff none of the above relations (or their inverses) hold between C1* and C2* (e.g., Mondays* $\sharp$ Tuesdays*).

We devised a set of heuristic rules for determining automatically which one of the 6 relations above holds between two user-defined I-Times. For instance, rule (IT) states that a definition of the form

C1* $\equiv$ n / C2* :during: C3*

implies C2* $\subset$ C3*, C1* $\prec$ C3* and C1* $\in$ C2* (e.g., from the definition Aprils* $\equiv$ 4 / Months* :during: Years* we have that Months* $\subset$ Years* and Aprils* $\prec$ Years* and Aprils* $\in$ Months*). Our rules proved to be powerful enough to cover "non exceptional" cases. However, since the user is completely free in the use of the specification language for I-Times, they do not cover all possible cases. If no relation between a pair of I-Times is determined by the heuristic rules, the relation is asked to the user.

5 Check-Intersection, Composition and Reasoning Process

5.1 Check-intersection

Check-intersection ($\underline{\cap}$) operates on two temporal specifications involving the same pair of periodic events and works in two steps. First, the intersection of the two Frame-Times is computed. If it is empty, then no further operation must be devised, and the original specifications are left unchanged. Otherwise, (i) the original specifications are left unchanged as regards the non-intersecting parts of the Frame-Times and (ii) in the intersecting part of the Frame-Times, check-intersection forces the compatibility of the qualitative temporal relations, depending on the relation between the I-Times. More specifically, the value of the check-intersection operation

(s15) [d1', d1"] ev1* Q C1 R1 ev2* $\underline{\cap}$
[d2', d2"] ev1* Q C2 R2 ev2*

(S) <Frame> ev1* <Quant> <I-Time>
<Qual-Rel> ev2*

where <Frame> is specified as the range of time spanning between a starting point and an ending point (e.g. [1-1-90, 1-6-94]), <Quant> is the quantifier "EACH", which stands for "exactly once each", <I-Time> is specified as in [Leban et al., 86] and <Qual-Rel> is a (possibly ambiguous) relation in Allen's Interval Algebra. For instance, the temporal content of Ex.1 above can be represented by (s4) (given the definitions of I-Times in (s1-s3)):

(s4) [1-1-90, 1-6-94] Sam-visits-office-X01*
EACH First-Monday-of-Aprils*
(BEFORE) Sam-goes-to-his-office*

The meaning of a temporal specification of the form [d1, d2] ev1* EACH C R ev2* is the following: for each instance C' of the I-Time C in the Frame-Time [d1, d2], (i) there is one and only one instance ev1' of the periodic event ev1* and one and only one instance ev2' of ev2* associated with C' and (ii) ev1' and ev2' are correlated, and the temporal relation R holds between them.

Our approach also deals with temporal specifications in which the I-Time is omitted. For instance, Ex.2 can be specified in our formalism by (s5)

(s5) $(-\infty, +\infty)$ EACH John-works* (BEFORE)
Mary-works*

with the meaning that, in the Frame-Time $(-\infty, +\infty)$, there is a one-to-one correspondence between instances of John-works* and instances of Mary-works*, and the relation BEFORE holds between the temporal extent of each correlated pair of instances. We also deal with temporal specifications in which only the I-time of a periodic event is specified. For example Ex.5 can be represented by (s6),

Ex.5 "Between 1-1-90 and 1-1-91 John run each Monday"

(s6) [1-1-90, 1-1-91] John-runs* EACH Mondays*

with the meaning that, between 1-1-90 and 1-1-91, there is exactly one instance of John-runs* associated with each Monday.

For the sake of brevity, in this paper we only consider temporal specifications expressed according to the schema (S) above. The complete description of our formalism is proposed in [Terenziani, 95].

3 Intersection and Composition of temporal specifications

Since our temporal specifications consider also I-Times and Frame-Times, new problems have to be faced when defining intersection and composition. As regards intersection, for instance, in our approach it is no longer true that at most one specification may relate each pair of events. In fact, different temporal specifications may be involved at different Frame-Times, or even in equal or overlapping Frame-Times (consider, e.g., (s7) and (s8)).

(s7) [1-6-91, 1-1-92] mail* EACH Days*
(BEFORE, MEETS) visit*

(s8) [1-1-91, 1-1-92] mail* EACH Mondays*
(BEFORE, AFTER) visit*

However, the consistency of the temporal specifications on the "overlapping parts" of the Frame-Times and of the I-Times must be checked. For instance, given (s7), the AFTER relation between mail* and visit* asserted in (s8) is not possible on Mondays since 1-6-91 until 1-1-92, and must be ruled out. Thus, we have to introduce an operation of "check-intersection" (indicated as $\underline{\cap}$), which gives the intersection of two temporal specifications (or an inconsistency) just in case the temporal specifications overlap in a given "context" (e.g. since 1-6-91 until 1-1-92 on Mondays* in (s7) and (s8)). Check-intersection may give in output more than one temporal specification; e.g., the application of check-intersection to (s7) and (s8) gives as result (s9), (s10) and (s11):

(s9) [1-1-91, 1-6-91) mail* EACH Mondays*
(BEFORE, AFTER) visit*

(s10) [1-6-91, 1-1-92] mail* EACH Days*
(BEFORE, MEETS) visit*

(s11) [1-6-91, 1-1-92] mail* EACH Mondays*
(BEFORE) visit*

Of course, the results of check-intersection crucially depend on the relations between the I-Times of the input specifications. For instance, if we put the I-time Tuesdays* in (s7) instead of Days*, the AFTER relation in (s8) has no longer to be ruled out. Analogously, also composition depends on the relation between the I-Times of the input specifications. For instance, the composition of (s11) above and (s12) gives as result (s13):

(s12) [1-6-91, 1-1-92] mail* EACH Days*
(AFTER) meeting*

(s13) [1-6-91, 1-1-92] visit* EACH Mondays*
(AFTER) meeting*

On the other hand, the composition between (s12) and (s14) reports an inconsistency, since it is not possible that mail* happens exactly once each day and once each week.

(s14) [1-6-91, 1-1-92] mail* EACH Weeks* (AFTER)
meeting*

Ex.2 "John always works before Mary"
Ex.3 "Bill always works (only) during Mary's work"

provides the information that John always works before Bill (see, e.g., [Morris, 93]). However, the approaches in this mainstream deal only with "context-independent" temporal specifications, in which no Frame-Time and no I-Time is considered (see, e.g., Ex.2 and Ex.3). This limitation allow these approaches to propose compact definitions of composition and intersection (see, e.g., [Morris, 93]), but compromises their practical applicability.

Our goal is that of extending the approaches in the second mainstream for dealing also with the "context" (Frame-Time and I-Time) in which periodic events occur (see, e.g., Ex.1), in order to increase their expressiveness and their practical applicability to areas such as scheduling, process-control, financial trading, work flow and office automation.

In section 2, we introduce our formalism for dealing with I-Times and qualitative relations between periodic events. In section 3, we discuss some of the main problems in the definition of intersection and composition of temporal specifications expressed in our formalism. In section 4, we distinguish between six different types of relations between I-Times. These relations are then used in section 5 for defining intersection and composition. In section 5 we also introduce a path-consistency algorithm which uses intersection and composition for performing temporal reasoning, and discuss its complexity. In section 6, we sketch some of the properties (e.g., correctness) of our approach, and in section 7 we briefly describe the architecture of TeMP, a temporal manager of periodic events which is based on the approach described in this paper.

2 Temporal Representation of Periodic Events

We assume time to be a linear order on a domain consisting of points. A time interval I is a convex set of points between a starting and an ending point. As in [Leban et al., 86], we define a *collection of intervals* as an ordered set of non-overlapping time intervals. Time intervals are the temporal extents in which events take place. Collections of time intervals represent the collection of the temporal extents upon which the different instances (realisations) of a periodic event take place. The formalism we use for specifying I-Times is that in [Leban et al., 86], who introduced a notation for defining basic calendars and two types of operators on collections of intervals (dicing -e.g., ":during:" in (s1) - and slicing -e.g., "2 /" in (s1)-) for building new user-defined collections on the basis of the basic calendars. For example, given the basic definitions of Days*, Weeks* and Months*, the collection of the first Mondays of April can be incrementally defined as follows:

(s1) Mondays* $\equiv$ 2 / Days* :during: Weeks*
(s2) Aprils* $\equiv$ 4 / Months* :during: Years*
(s3) First-Monday-of-Aprils* $\equiv$
1 / Mondays* :during: Aprils*

In order to deal with the qualitative temporal relations between periodic events, we adopt the qualitative relations of Allen's Interval Algebra [Allen, 83]. However, since Allen's relations hold between pairs of time intervals, and periodic events happen over collections of time intervals, a way for relating pairs of time intervals belonging to different collections is needed. As in [Morris et al., 93], we introduce the equivalence relation of *correlation* between pairs of instances of periodic events, which holds as a result of some contingent relation in the world between them. For instance, in Ex.1, correlation holds between each corresponding pair of Sam visiting the branch office X01 and Sam going to his office, and the temporal relation "before" holds between (the temporal extents of) each pair of correlated instances. Since we deal with qualitative relations which hold in an I-Time, we consider also the relation of *association*, which relates each instance of a periodic event with the instance of the I-Time in which it occurs. In particular, an instance e of a periodic event which occurred in a time interval i is associated with an instance p of an I-Time if and only if i is contained in p.

Since the user may introduce more than one specification concerning the very same type of event, s/he may want to specify that these specifications concern the same periodic event (in this case, we say that the different specifications *co-designate* [Morris et al., 93] the same periodic event) or different periodic events of the same type. For example, Ex.4 introduces two different periodic events of the same type "John brushing his teeth". In other words, Ex.4 involves two different collections of time intervals in which John brushes his teeth.

Ex.4 "Each day, John brushes his teeth after breakfast and after lunch"

In our approach, different indexes are used to distinguish between different collections of the same type of event (e.g., $John-brush^*_1$ and $John-brush^*_2$). Indexes will be omitted in the rest of the paper, for the sake of clarity.

2.1 Temporal formalism

In our approach, complex specifications of periodic events can be provided, according to the syntax (S):

Reasoning about Periodic Events

P. Terenziani
Dipartimento di Informatica, Università di Torino,
Corso Svizzera 185, 10149 Torino, Italy
E-mail: terenz@di.unito.it

Abstract

The paper describes a temporal formalism which deals with both (i) quantitative information concerning the frame of time and the user-defined calendar-dates in which periodic events are located and (ii) the qualitative relations between periodic events. The paper defines the operations of intersection and composition of temporal specifications, and describes an algorithm which takes advantage of these operations for performing temporal reasoning. This is the kernel of TeMP, a temporal manager of periodic events.

1 Introduction

Periodic events are widely studied in many research areas, such as Artificial Intelligence (AI) and Temporal Databases (TDB). In particular, most AI and TDB approaches provided a high-level, powerful and user-friendly formalism for representing periodic events, and (especially in AI) some form of temporal reasoning operating on them. This work belongs to such a stream of research, and aims at providing a framework in which it is possible to deal also with very rich temporal specifications, such as

Ex.1 "Between 1-1-90 and 1-6-94, each first Monday of April Sam visited the branch office X01 before going to his office "

(following [VanEynde, 87], we call *Frame-Time* the interval which contains all the instances of the event -e.g., "From 1-1-90 to 1-6-94" in Ex.1-, *I-Time* the periodic time interval over which periodic events take place, which is usually expressed by some calendric expression -e.g., "first Monday of April" in Ex.1- and *e-Time* the time in which the actual instance of the periodic event occurred -e.g., "before going to his office"). Current AI and TDB approaches do not allow one to deal with such complex specifications. In particular, current approaches can be roughly divided into two mainstreams, depending on the types of temporal information they deal with.

In the first mainstream (carried on especially in the TDB community) most attention is devoted to the treatment of I-Times (see, for instance, [Leban et al., 86], [Chomicki and Imielinsky, 88], [Kabanza et al., 90], [Soo and Snodgras, 92], [Baudinet, 93], [Chandra and Segev, 93], [Soo, 93]). These approaches are based on the consideration that, in most cases, periodic events are "context-dependent", in the sense that they take place at specific periods of time (I-Times). Moreover, since different calendric systems are used for specifying I-Times (depending e.g., from cultural and social factors; see e.g., [Soo, 93]), most of these approaches stress the necessity of dealing with user-defined calendars. This is necessary, e.g., for dealing with temporal information in many areas, such as scheduling, manufacturing, process-control, financial trading, work flow and office automation (see, e.g., the discussions in [Stonebraker, 90], [Chandra and Segev, 93] as regards financial trading, and consider [Soo and Snodgrass, 92], [Soo, 93] for a more general discussion). However, the approaches in this mainstream do not consider the possibility of specifying the e-Time of a periodic event in term of its relative position with respect to other periodic events (as, e.g., the qualitative relation "before" in Ex.1). As a consequence, these approaches devised only limited forms of temporal reasoning.

On the other hand, the approaches in the second mainstream (carried on especially in the AI community) focus on the treatment of quantifiers and e-Time, stressing the importance of dealing with qualitative relations between periodic events (see, for instance, [Ladkin, 86a, 86b], [Poesio, 88], [Ligozat, 91], [Morris et al., 93]). This involves, among other things, the development of some form of temporal reasoning. Following [Allen, 83], also temporal reasoning about periodic event has been usually performed using the basic operations of intersection, for checking the consistency of temporal specifications, and composition, for inferring new specifications; for example, the composition of Ex.2 and Ex.3