Generating Scenarios from Specifications of Repeating Events

Robert A. Morris and **Lina Khatib**
Florida Institute of Technology
Computer Science Program
Melbourne, FL 32901
morris@cs.fit.edu

Gérard Ligozat
LIMSI, Université Paris-Sud,
B.P. 133 91403 Orsay, Cedex, France
ligozat@limsi.fr

Abstract

This paper addresses the problem of reasoning about events that recur over time, especially when knowledge about those events may be incomplete or qualitative. A constraint-based framework is proposed to solve the problem of mapping a specification of temporal information about recurring events to a *scenario*, a qualitative representation of an assignment of times to those events. The main result of this paper is to reduce the problem of generating recurrence scenarios to the problem of generating a set of scenarios for convex interval relations.

1 Introduction

Many temporal reasoning problem involve a concept of *recurrence*. For example, consider the following specification: *Office hours (OH) happen twice a week. Faculty meetings (FM) happen at least once a week. The CS-1 class (CM) meets twice a week. Preparation time (PT) only precedes or meets class meetings. Class meetings, in general, are always before, or finished by, lab research meetings (LR), and once a week the class meetings are finished by the lab meetings. Preparation time is always and only during office hours. Unless otherwise indicated, each occurrence is disjoint from every other.* This specification contains qualitative and quantitative knowledge about recurring events.

The statements in this specification can be classified into three kinds, viz., expressing: *cardinality constraints* (faculty meetings happen at least once); *recurring relationships*: (preparation time only precedes or meets classes); and *global constraints*: (every occurrence is disjoint from every other).

A *scenario* offers a finite, qualitative abstraction of an infinite set of *solutions*, assignments of times to events. A graphical representation of a scenario for the example specification is found in Figure 1. Each event occurrence is indicated by a horizontal line, and the temporal relationships by their relative positions. The meaning of the rectangles and arrows in the figure are explained below. The aim of this paper is to describe a complete framework for representing the temporal information found in specifications of the kind just illustrated, and to describe a reasoning mechanism which, applied to this knowledge, is capable of generating scenarios for recurring events. The paper extends the framework found in [5], [6].

2 Recurrence Relationships

Relational contexts involving plural noun phrases describing recurring events (e.g. *class meetings*) can be interpreted as a form of *collective predication* (i.e. predication over collections of elements from some domain). Thus, the domain of discourse for reasoning about recurrence can be viewed as collections of the standard time units, either intervals or points. For example, if intervals are viewed in the normal way as ordered pairs of minimal time units, then the set

$$I = \{\langle I_1^-, I_1^+\rangle, \langle I_2^-, I_2^+\rangle \ldots, \langle I_n^-, I_n^+\rangle\}$$

is a potential interpretation of a plural noun phrase describing a finitely recurring event[1].

For the sake of completeness, we review the proposed semantics for predicative contexts involving collections of intervals. Given a temporal relation such as *follows*, it is possible to express a recurrence relationship by pluralizing: thus, the sentence **games follow national anthems** expresses a recurring relationship between recurring events. Contextual knowledge about games and national anthems indicates a one-to-one mapping between occurrences. In the first-order representation, annotations can be added to quantifiers to express this type of mapping. For example,

$$(\forall i : games)(\exists_d! j : N.A.) follow(i, j)$$

represents a one-one functional mapping, where $\exists_d!$ is read "there exists a unique, distinct". Other contexts are compatible with different mappings; for example, the sentence **parallel sessions precede lunches** suggests a many-to-one mapping between sessions and lunches.

Applying temporal adverbs such as "always", "only" and "sometimes" to these contexts refines their meaning. Consider, for example **games only follow national anthems**. The truth condition of this sentence implies a notion of "corresponding pairs" of occurrences. Again following the previous analysis, corresponding components will be said to be *correlated*. Formally, correlation is an equivalence relation over its domain.

The sentence **Preparation time only precedes or meets classes** in the example specification can be expressed in first-order logic as the following conjunction:[2]

$$(\forall i \in PT)(\exists_d! j \in CM) COR(i, j) \wedge \{p, m\}(i, j)$$

$$\wedge[(\forall i \in PT)(\forall j \in CM) COR(i, j) \rightarrow \{p, m\}(i, j)]$$

Here, the variables range over interval components, and the expression "$COR(i, j)$" expresses the correlation of components. If a time frame is specified, relational expressions such as "$WEEK(i, j)$" (i and j happen during the same week) can be added to the representation. In this paper, we ignore the issue of reasoning with different time frames.

This class of first order formulas will be abbreviated as second-order relational expressions. The relations which provide the interpretations of these expressions will be called $Q\exists$-relations. This second-order representation will be expressed in the form $QR(R)$ where QR stands for a $Q\exists$-relation, and R stands for an interval temporal relation (e.g. an Allen relation between pairs of intervals). Rather than review the formal semantics of these contexts, illustrative examples are provided in Table 1. I and J stand for recurring events. The table translates an English context involving recurrence relations into a second-order formula ex-

[1] In previous accounts, such an I is called an n-interval.

[2] The thirteen atomic interval relations will in this paper be abbreviated as follows: precedes (p), preceded by (pi), meets (m), met by (mi), equals (eq), starts (s), started by (si), finished (f), finished by (fi), during (d), contains (di), overlaps (o), overlapped by (oi). Following custom, the set notation is used to abbreviate contexts involving disjunctions of these relations.

pressing a $\forall\exists$-relation, and its first-order equivalent.

Certain collections of recurrences relations form the domain elements of an *interval algebra* [7]. In [6], a *recurrence algebra* $\mathcal{RA}$ is defined based on a set of recurrence relations of the form discussed briefly above. This algebra is closed under the operators inverse, intersection and composition. Intersection of recurrence relations results in the formation of relations which are best expressed as a form of conjunction. For example, the sentence **meetings always follow or meet lunches; furthermore, they sometimes meet lunches** expresses a relation which can be viewed as the intersection of two recurrence relations: **always follow or meet** and **sometimes meet**. We call the *sometimes* operator in this context the *refinement* operator, and the relations that result the *refining relations*; the relation being refined will be called the *leading relation.* Following the earlier notation [5], the symbol $\oplus$ will stand for *furthermore.*

Furthermore can be generalized to arbitrary finite conjunctions of simple relations. We define a *normal form* for recurrence relations in $\mathcal{RA}$ as

$$QR(R) \oplus \exists\exists_d!(S_1) \oplus \ldots \oplus \exists\exists_d!(S_n)$$

where

1. R is one of the 2^{13} atomic interval relations [1];
2. QR is either $\forall\exists_d^o!$, $\forall\exists_d!^{oi}$, $\forall\exists_d!$, or $\forall\exists_d!^{o\sqcap oi}$ (see Table 1);
3. $S_i \subset R, i = 1 \ldots n$; and
4. each S_i is *atomic*, i.e., consisting of a single temporal relation.

Remark 1 *Let RA be the set of recurrence relations in normal form. For pairs $R, S \in RA$, let inverse R^{-1}, intersection $R \sqcap S$, and composition $R \otimes S$ be defined as in [6]. The set RA is closed under these operations.*

The proof appeared in [6].

3 Reasoning About Recurring Relations

First, we extend the notion of temporal constraint network for the purpose of storing information about recurring events:

Definition 1 *($\mathcal{RA}$ Networks) An $\mathcal{RA}$* network *is a network of binary relations where*

1. *variables represent collections of convex intervals;*
2. *for each variable I, D_I, the domain of I is the set of all sets of non-overlapping convex intervals; and*
3. *the binary relations between variables are elements of RA.*

The restriction of elements of the domain to sets of *non-overlapping* intervals means that the elements within the set can be totally ordered. This restriction is not necessary, but is typical for the sorts of applications under consideration.

The next definition generalizes the corresponding notion within the interval calculus:

Definition 2 *(Instantiation) An* instantiation *of m variables in a set $\mathcal{I}$ representing collections of intervals is an m-tuple of sets of intervals representing an assignment of elements of D_I to each $I \in \mathcal{I}$. Given a set of binary temporal relations from RA between elements of $\mathcal{I}$, a* consistent instantiation *is an instantiation that satisfies all the relations in the set. An RA-network is* consistent *if such an instantiation exists; otherwise, the network is inconsistent.*

The next concept formalizes the various patterns of recurrence corresponding to a given recurrence relation:

Definition 3 *(Concretization of a Recurrence Relation) A recurrence relation*

$$\mathcal{R} = QR(R) \oplus \exists\exists(S_1) \oplus \ldots \oplus \exists\exists(S_n)$$

defines a set of concretizations $\mathcal{R}^c$ *taken from a set of elements* $\mathcal{A}$*, called the* alphabet *for* $\mathcal{R}$*.* $\mathcal{A}$ *contains* R*, as well as possibly other elements, depending on the value of* QR*, as follows:*

1. *If* $QR = \forall\exists_d!^o$ *then* $\mathcal{A}$ *contains a symbol representing occurrences of an element in range*$(\mathcal{R})$*, and each concretization in* $\mathcal{R}^c$ *contains zero or more occurrences of this symbol;*
2. *If* $QR = \forall\exists_d!^{oi}$ *then* $\mathcal{A}$ *contains a symbol representing occurrences of an element in dom*$(\mathcal{R})$*, and each concretization in* $\mathcal{R}^c$ *contains zero or more occurrences of this symbol;*
3. *Finally,* $\mathcal{A}$ *contains 0, and each concretization in* $\mathcal{R}^c$ *contains zero or more occurrence of this symbol.*

Terminology: if $\mathcal{R}$ *is a recurrence relation between* I *and* J*, then* $\mathbf{i}$ *and* $\mathbf{j}$ *will be the the symbols representing arbitrary elements of* I *and* J*, respectively. Each element in* $\mathcal{R}^c$ *is called a* concretization*. Each concretization* r *must contain an occurrence of each* $S_i, i = 1 \ldots n$*. A* k*-concretization is a finite concretization of length* k*.*

Example 1 *(Concretization) The sequence* $\langle p, m, p, m, o, o, \ldots \rangle$ *represents an infinite recurrence of temporal relations between pairs of collections of intervals. It is a concretization of many recurrence relations; among them*

$$\forall\exists_d!^{o \sqcap oi}\{p, m, o\}.$$

To properly explain the notion of concretization, and its corresponding set $\mathcal{A}$, out of which concretizations are built, it is necessary to revisit the notion of *correlated* occurrences. A recurrence scenario can be viewed as a *partition* of a set of occurrences based on the relation COR. Formally:

Definition 4 *(C-partition). Given a set* $\mathcal{I}$ *of finite sets of intervals and* $I, J \in \mathcal{I}$*, a* c-partition *of* $\mathcal{I}$ *is the structure* $\langle \{[X_i] : i = 1 \ldots k\}, COR \rangle$*, for some* k*, such that* $\{[X_i]\}$ *is a partition of* $\bigcup_{I \in \mathcal{I}} [\bigcup_{I_j \in I} I_j]$ *determined by an equivalence relation* COR*, where*

$$[X_i] = [X_j] \textit{ if and only if } COR(X_i, X_j)$$

Each $[X_i]$ *will be called a* stage *in the c-partition. A c-partition will be said to be* well-behaved *if there exists a total ordering* $<_{COR}$ *of the elements of* $\{[X_i]\}$ *defined by*

$$[X_i] <_{COR} [X_j] \textit{ if and only if}$$

$$(\forall I_j \in [X_i])(\forall J_p \in [X_j]) I_j < J_p;$$

For example, Figure 1 represents a well-behaved c-partition of a set of repeating events. Each pair of intervals i, j in the c-partition (represented by the rectangles) are correlated. Examples of correlated intervals are represented by the arrows. Any permutation of this order is also a c-partition. Not all c-partitions are well-behaved, as a later example will illustrate. It may not, in fact, be possible to generate a well-behaved c-partition, given a specification of recurrence information.

An ordered c-partition corresponds to a concretization r as follows. Let $r[i]$ be the ith element of a concretization r of a recurrence relation between I and J. This element will typically denote a temporal relation between some $I_m \in I$ and some $J_n \in J$ in the ith stage of the ordered c-partition of the set containing I and J. Notice however that, in general, it is not required for a stage to contain an element from each recurring event. For example, in the second stage of the c-partition in Figure 1, there is no occurrence of either OH or PT, hence no temporal relation between occurrences. In addition, there is an occurrence of FM but not of CM in the same partition. We will say that if $r[m] =\mathbf{i}$ ($\mathbf{j}$), then an occurrence of some element of I (J) appears in the mth stage of a c-partition of a set containing I and J. If $r[i] = 0$, then neither I nor J has an element in the ith stage of the c-partition that corresponds to r.

As a final preliminary to a formalization of the notion of recurrence scenario, we associate instantiations of a set of interval collections to concretizations. For simplicity, and without loss of generality, we assume that a set of concretizations have a fixed length k:

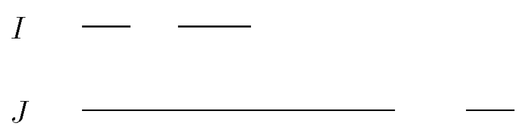
Definition 5 *(C-satisfaction) Let $\mathcal{I}$ be a set of variables standing for recurring events. An instantiation of every element of this set* c-satisfies *a set of k-concretizations $\{r_{I,J} : I, J \in \mathcal{I}\}$ if there exists a c-partition $\{[X_i]\}$ of $\bigcup_{I \in \mathcal{I}}[\bigcup_{I_i \in I}]$ into k disjoint, non-overlapping sets such that, for each $m = 1 \ldots k$ exactly one of the following holds for each $[X_i], i = 1 \ldots k$ and each concretization $r_{I,J}$:*

1. *if $r_{I,J}[i]$ is an atomic interval temporal relation, there exists a pair $[I_q], [J_p]$ such that $[X_i] = [I_q] = [J_p]$ and $\langle I_q, J_p \rangle$ satisfies the relation $r_{I,J}[m]$;*
2. *if $r_{I,J}[m]$ is $\mathbf{i}$, there exists an $[I_q]$ such that $[X_i] = [I_q]$ and there exists no $[J_p]$ such that $[I_q] = [J_p]$;*
3. *if $r_{I,J}[m]$ is $\mathbf{j}$ there exists a $[J_p]$ such that $[X_i] = [J_p]$ and there exists no $[I_q]$ such that $[J_p] = [I_q]$;*
4. *if $r_{I,J}[m]$ is 0, then for no $[I_q], [J_p]$ is it the case that $[X_i] = [J_p]$ or $[X_i] = [I_q]$.*

Example 2 *Consider the instantiation:*

$$I = \{\langle 0, 1\rangle, \langle 2, 4\rangle\}, J = \{\langle 0, 10\rangle, \langle 12, 13\rangle\}.$$

This corresponds to the following scenario:



This instantiation c-satisfies the concretization $\langle s, p\rangle$, as well as $\langle \mathbf{i}, d, \mathbf{j}\rangle$.

Notice that no well-behaved c-partition exists for this instantiation. There are weaker, "natural" c-partitions which are not well-behaved. Intuitively, natural c-partitions are any that partition on the basis of temporal proximity. They are natural because they correspond to the manner in which recurrence patterns are organized in thought and communicated.

3.1 Generating a Recurrence Scenario

A scenario for a set of n repeating events $\mathcal{I}$ can be viewed as a set $\mathcal{C} = r_1, \ldots, r_{n(n-1)}$ of concretizations of recurrence relations, one for each pair of elements of $\mathcal{I}$, that satisfy certain properties. Intuitively, for each $i = 1 \ldots n(n-1)$, the set $\{r[i]_1, \ldots r[i]_{n(n-1)}\}$ contains consistent temporal information. The computational problem of interest, then, is that of generating a consistent set of concretizations from a specification.

The notion of a concretization network aids in solving this problem:

Definition 6 *(Concretization Network) Given a $\mathcal{RA}$-network, M, M^c is a concretization of M if each edge $M^c_{I,J}$ of M^c is labeled with a concretization $r_{I,J}$ of the recurrence relation $R_{I,J}$ which labels the edge $M_{I,J}$ of M.*

This leads, finally, to the generalization of the notion of a *consistent scenario*:

Definition 7 *(k-Scenario) A concretization network M^c of $\mathcal{RA}$-relations is a* k-scenario *of a network M if every edge in M^c is labeled by a concretization of length k of the corresponding relation in M. M^c is a consistent k-scenario if there is an instantiation of all the variables in M^c that c-satisfies all the labels on the edges of M^c.*

A simple method transforms a concretization network into a set of interval relation networks.

Definition 8 *(I-transformation of a finite concretization network). Given a finite concretization network M^c, and index $p = 1 \ldots k$, where k is the maximum length of any concretization in M^c, the pth I-transformation of M^c is a network $M[p]$ defined as follows:*

1. *Each vertex of $M[p]$ represents an element of some I in M for which the p^{th} element of the concretization $r_{I,J}$ (denoted by $r[p]_{I,J}$) on the arc $M_{I,J}$ (equivalently: on the arc $M_{J,I}$) consists of either an atomic interval relation or* **i***;*

2. *For each pair of vertices I, J, the arc $M[p]_{I,J}$ is labeled as follows:*
   - *$M[p]_{I,J} = r[p]_{I,J}$, if $r[p]_{I,J}$ is an atomic interval relation; otherwise*
   - *$M[p]_{I,J}$ =?? where ?? denotes the universal interval relation (i.e., the relation that all pairs of intervals satisfy).*

M is an I-transformation of a concretization network M^c if, for some p, M is $M[p]$, the p^{th} I-transformation of M^c.

Remark 2 An I-transformation is a convex interval relation network.

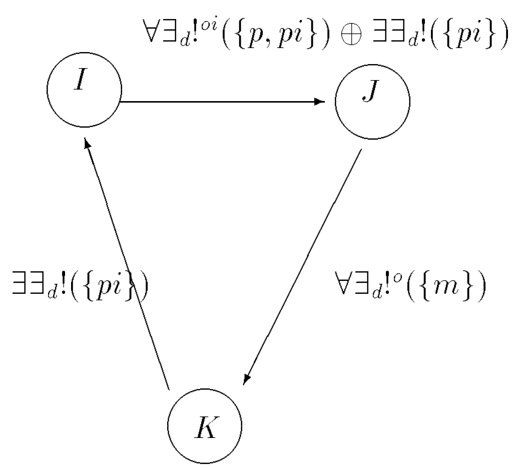
It is now possible to reduce the problem of generating a consistent scenario from a concretization network to the problem of generating a set of solutions to interval networks. This reduction is based on the following result, whose proof follows immediately from the preceding definitions:

Theorem 1 *A concretization network M^c is a (finite) m-scenario of an $\mathcal{RA}$-network M if:*

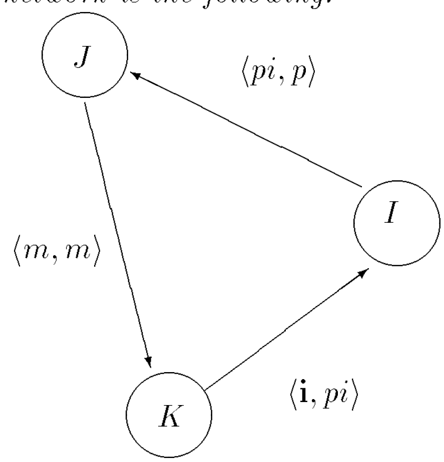
1. *Each edge $M^c_{I,J}$ of M^c is labeled by a member of the set $R^c_{I,J}$ of concretizations of the relation $R_{I,J}$ on arc $M_{I,J}$; and*

2. *For each $k = 1 \ldots m$, the I-transformation $M[k]$ of M^c has a consistent scenario.*

Example 3 *Consider the following $\mathcal{RA}$-network:*



A concretization network for this $\mathcal{RA}$-network is the following:



This network is a 2-scenario of the previous network.

The theorem suggests that the problem of generating k-scenarios from specifications of recurrence relations stored as a $\mathcal{RA}$-network M can be reduced to the problem of taking a set $\mathcal{I} = I_1, \ldots, I_n$ of repeating events and their $n(n-1)$ relations and generating an $n(n-1) \times k$ matrix such that

- The set of values in each column of the matrix define relations which can be I-transformed into a scenario for a set of intervals; and
- Each row of the matrix defines a concretization for the recurrence relation for some pair of repeating events.

An algorithm based on this reduction has been designed; it will be discussed in future work. The algorithm is similar to, and in fact utilizes, a technique similar to that employed by Ladkin [3] for generating a scenario of interval temporal relations.

4 Summary

The objective here has been the development of a framework for reasoning about the recurrence of temporal relations. To meet this objective, we focus on specifications of recurrence whose counterparts in natural language involve the application of an adverbial modifier to a prepositional phrase describing a temporal order. These contexts are mapped into second-order unary relational expressions interpreted over pairs of collections of intervals. The class $\mathcal{RA}$ of recurrence relation is a subset of the $Q\exists$-class of relation, which were found to make up the logical structure of relation recurrence. Knowledge expressible as an $\mathcal{RA}$-relation can be stored and manipulated within a constraint-based framework. In particular, the main result of this paper was showing how recurrence scenarios can be generated from specifications of recurring events.

References

[1] Allen, J. (1983) Maintaining Knowledge About Temporal Intervals. In Brachman, R., and Levesque, H., (eds.) *Readings in Knowledge Representation*, (San Mateo:Morgan Kaufman), 510- 521.

[2] Ladkin, P.B. and Maddux, R. (1988) The Algebra of Constraint Satisfaction and Temporal Reasoning. Technical Report, Kestrel Institute, Palo Alto, CA.

[3] Ladkin, P.B., On Binary Constraint Problems. *Journal Of the ACM* Vol. 41, Number 3, 435-469.

[4] Ligozat, G., On Generalized Interval Calculi. In *Proceedings of the Ninth National Conference on Artificial Intelligence* (AAAI-91), 234-240.

[5] Morris, R., Shoaff, W., Khatib, L., Path Consistency in a Network of Non-Convex Intervals.*Proceedings of the 13th International Joint Conference on Artificial Intelligence (IJCAI-93)*, Chambery, France (1993), pp. 650-655.

[6] Morris, R., Shoaff, W., and Al-Khatib, L., (1994) Reasoning about Recurrence. Forthcoming in *The Journal of Computational Intelligence.*

[7] Van Beek, P. (1990). *Exact and Approximate Reasoning about Qualitative Temporal Relations.* Technical Report TR 90-29, University of Alberta, Edmonton, Alberta, Canada.

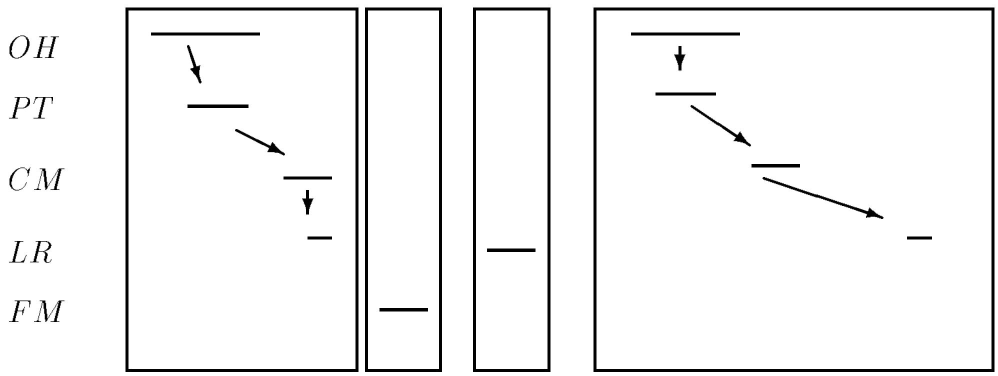


Figure 1: A scenario for the academic scheduling example

English Context	**Second-order Form**	**First-order Equivalent**
I only before J	$I\ \forall\exists_d!^{o}(\{p\})\ J$	$(\forall i \in I)(\exists_d! j \in J) COR(i,j) \wedge \{p\}(i,j) \wedge [(\forall i \in I)(\forall j \in J) COR(i,j) \rightarrow \{p\}(i,j)]$
I always before J	$I\ \forall\exists_d!^{oi}(\{p\})\ J$	$(\forall j \in J)(\exists_d! i \in I)\ COR(i,j) \wedge \{p\}(i,j) \wedge [(\forall i \in I)(\forall j \in J) COR(i,j) \rightarrow \{p\}(i,j)]$
I always and only before J	$I\ \forall\exists_d!^{o \sqcap oi}(\{p\})\ J$	$(\forall j \in J)(\exists_d! i \in I)\ COR(i,j) \wedge \{p\}(i,j) \wedge (\forall j \in J)(\exists i \in I)\ COR(i,j) \wedge \{p\}(i,j) \wedge [(\forall i \in I)(\forall j \in J) COR(i,j) \rightarrow \{p\}(i,j)]$
I before J	$I\ \forall\exists_d!(\{p\})\ J$	$(\exists i \in I)(\exists_d! j \in J)\ COR(i,j) \wedge \{p\}(i,j) \wedge [(\forall i \in I)(\forall j \in J) COR(i,j) \rightarrow \{p\}(i,j)]$
I sometimes before J	$I\ \exists\exists_d!(\{p\})\ J$	$(\exists i \in I)(\exists_d! j \in J)\ COR(i,j) \wedge \{p\}(i,j)$

Table 1: Interpreting Recurrence