A Partitioning Algorithm for Detecting Eventuality Coincidence in Temporal Double recurrence

B.O. Akinkunmi
(Department of Computer Science, University of Ibadan, Ibadan, Nigeria
ope34648 [at] yahoo [dot] com
bo [dot] akinkunmi [at] mail [dot] ui [dot] edu [dot] ng)
A Partitioning Algorithm for Detecting Eventuality Coincidence in Temporal Double Recurrence

Abstract
A logical theory of regular double (or multiple) recurrence of eventualities (which are regular patterns of occurrences that are repeated) in time, has been developed within the context of temporal reasoning that enabled reasoning about the problem of coincidence. i.e. if two complex eventualities (or eventuality sequences) consisting respectively of component eventualities \(x_0, x_1, \ldots, x_r\) and \(y_0, y_1, \ldots, y_s\) both recur over an interval \(k\) and all eventualities are of fixed durations, is there a subinterval of \(k\) over which the occurrence \(x_p\) and \(y_q\) for \(1 \leq p \leq r\) and \(1 \leq q \leq s\) coincide.

We present the ideas behind a new algorithm for detecting the coincidence of eventualities \(x_p\) and \(y_q\) within a cycle of the double recurrence of \(x\) and \(y\). The algorithm is based on the novel concept of \(gcd\)-partitions that requires the partitioning of each of the incidences of both \(x\) and \(y\) into eventuality sequences each of which components have a duration that is equal to the greatest common divisor of the durations of \(x\) and \(y\). The main idea behind the algorithm is the fact that there exists a subinterval of the cycle of double recurrence over which any pair of component eventualities from the \(gcd\)-partitions of \(x\) and \(y\), are fully incident. Thus we only need determine whether or not a coincidence happens during one of those intervals when such a pair from the \(gcd\)-partition, with which \(x_p\) and \(y_q\) are non-disjoint for each of the incidences of \(x\) and \(y\), fully coincide. The \(gcd\)-partitioning of a double recurrence significantly reduces the search space that one needs to explore while looking for the coincidence of \(x_p\) and \(y_q\) from a cycle of multiple recurrence to two periods of \(x\) and \(y\).

The worst-case running time of the partitioning algorithm is linear in the maximum of the durations of \(x\) and \(y\), while the worst case running time of an algorithm exploring a complete cycle is quadratic in the durations of \(x\) and \(y\). Hence the partitioning algorithm works faster than the cyclical exploration in the worst case.

Keywords: temporal reasoning, double recurrence, regular patterns, partitioning.

1. Introduction

The problem of temporal recurrence is known in knowledge representation \([1, 8, 9]\). The concept entails having recurring incidences of an eventuality (such as state, event, action)[6] or an eventuality sequence over an interval. An example of an eventuality sequence recurring calendar time state sequence recurring over time is the days of the week sequence: \(<\text{Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday}>\). When two different sequences recur over an interval, it is sometimes the case in the process of planning or scheduling that we either want certain pairs of states in either of the sequences to coincide either for, (borrowing the language of distributed systems), liveness reasons (i.e. in order to make a plan to succeed) or not coincide for safety reasons(i.e. in order to prevent an unwanted situation).

The problem of coincidence in double and regular temporal recurrence was introduced in \([2]\). The computational solution proposed in that paper requires temporal projection over a cycle of double recurrence or in other words, one needs to explore only a cycle of double recurrence in order to determine coincidence. This paper presents a solution to the problem of coincidence in...
double regular recurrence that is based on the notion of partitions. The solution presented here requires projection over two periods of the occurrence of the eventuality sequences which are shorter intervals than the cycles of recurrence.

Our interest in this problem has been motivated by the need to enable a planner/scheduler cope with reasoning about the possible occurrence of the preconditions of an action in the context of double recurrence as opposed to a single recurrence addressed in [8]. Given that two sequences of eventualities $x_0, x_1, ..., x_r$ and $y_0, y_1, ..., y_s$ (such that any pair of eventualities from the $x$ sequence or the $y$ sequence are mutually exclusive) both recur over some interval, and a planner requires the coincidence of an eventuality pair $x_p$ and $y_q$ from the sequences, in order to carry out a certain action. Can a planner infer the existence (or otherwise) of such an interval which is a subinterval of two intervals over which $x_p$ and $y_q$ are both true. In other words can a planner anticipate the occurrence or otherwise of such a coincidence?

Sometimes in addition to knowing that a coincidence which is a pre-requisite for an action will happen, it is also important, that the planner/scheduler must be able to know within which time limit the coincidence will happen. This is particularly useful where the planner is dealing with deadlines and may have other alternatives to that course of action. In that regard, the main result of [2] is that if a coincidence for those eventualities would exist at all, one must happen within every cycle of double recurrence. In the light of that knowledge, one should have to computationally explore a whole cycle of double recurrence in order to determine coincidence.

A simple example of the coincidence problem from a factory domain arises when a piece of factory machinery which must work continuously for five days which must be followed by a maintenance process that must last three continuous days, but the maintenance engineer will not be available on a Wednesday. In this case a scheduler must reason about the avoidance of a situation in which the maintenance period includes a Wednesday. Here there are two repeated regular patterns. One involves the weekday sequence: Monday, Tuesday, Wednesday, Thursday, Friday, Saturday and Sunday while the other involves the factory’s repeated switching from a state of working (5 days) to a state of maintenance (3 days).

In this example a scheduler must hope that he can avoid a situation in which one of the three consecutive maintenance days is a Wednesday. If the factory starts working on a Monday, Then the first maintenance days will be a Saturday to Monday. Then the next one will be a Sunday to Tuesday, while the third one will be a Monday to Wednesday. The next three maintenance sessions will all include a Wednesday. Thus our planner should know that it cannot possibly avoid a situation in which Wednesday is a maintenance day.

The conclusion in the last paragraph was arrived at following an explicit exploration into any arbitrary cycle, using our knowledge of the eventualities that make up the two recurring sequences, and their relative durations. In this case there are two sequences. The first one is the sequence of weekdays with eventualities <Monday, Tuesday, Wednesday...Sunday>, each with a duration of 1 day. The second sequence consists of two eventualities <Working_period, Maintenance_period>, where the first eventuality has a duration of 5 days, while the other one has a duration of three days. The duration of the first sequence is 7 days long while the other sequence is 8 days long. A solution based on such temporal projection will be computationally costly. It will take temporal projection over 56 days (which is a cycle of the double recurrence of the two eventuality sequences) in the worst case in order to conclude whether or not a coincidence will happen.
The problems described above are examples of the coincidence problem which arises in contexts described by an extension of Koomen’s parlance [8, 9] as “double recurrence”. In this paper, even though, our ultimate goal is to present an algorithmic solution to the problem of eventuality coincidence which arises when one is required to carry out planning or scheduling within such a context, but in the process, we accompany every definition with a logical formalization which enables us to prove basic properties about temporal relations of eventualities within two sequences. All theorems in this paper can be proved logically except Theorem 5.1 whose proof is based on the properties of pseudo-random number generators. However we believe that the paper should be readable to those who are not interested in logical formalization and proofs.

In a sense, we can view this problem as a (temporal reasoning) search problem. We are searching an interval over which two different eventualities recur for a coincidence of a pair of components from the two recurring eventualities. We can divide an interval over which temporal double recurrence happens into cycles. Each cycle is exactly like others in terms of the temporal relations between any pair of components from the two eventualities. As such it has been shown that any search for coincidence within an interval over which double recurrence take place can be narrowed down to any arbitrary cycle[2]. This paper presents the key ideas behind an algorithm that uses the concept of gcd-partitions to narrow down the search space further.

A partition for an eventuality sequence is any other eventuality sequence whose duration is the same as it. A gcd-partition for a recurring pair of sequence eventualities x, y is another pair of sequence eventualities w, z such that:

- w is a partition of x and z is a partition of y.
- the duration of each of the component eventualities in w and z is equal to the greatest common divisor of the durations of x and y.

A basic property of gcd-partitions is that any pair of component eventualities from each of the sequences w and z would both be maximally (fully) incident over some subinterval of any cycle of the double recurrence of x and y. Therefore we can narrow down our search to those intervals, over which the pairs of components of w and z that each share a common interval with x_p and y_q respectively, are fully coincident.

Thus eventualities x_p and y_q coincide within any cycle of the double recurrence of x and y if and only if:

There is an eventuality pair w_r and z_s from the gcd partition w and z such that:
- any incidence of x_p is non-disjoint with any incidence of w_r within any incidence of x
- any incidence of y_q is non-disjoint with any incidence of z_s within any incidence of y
- the incidences of w_r and z_s being the same implies the corresponding incidences of x_p and y_q that are non-disjoint with those of w_r and z_s respectively are non-disjoint.

As such there are three major parts to the algorithm needed to implement this solution. The first part should find the greatest common divisor of the durations of x and y. The second part should create a gcd partition w, z for the pair of eventualities x and y storing the exact (qualitative and quantitative) relationships between incidences of x_p (y_q) and those of components of the partition w (z respectively) with which it is non-disjoint. The third part algorithm will then determine whether if any pair of such components from w and z happen over the same time interval would imply that the incidences of x_p and y_q are non-disjoint.
The worst case running time of the part of the second part of the algorithm is linear in the maximum duration of the incidence of the eventualities of x and y. That is asymptotically more significant than the running time of an algorithm that finds the greatest common divisor which is the square the logarithm of the maximum of the durations of x and y. The worst case running time of the third part of the algorithm however is constant. This is because if incidences of \( x_p \) (\( y_q \)) is non-disjoint with incidences of more than two components of w (respectively z) then it is guaranteed that there is a pair of non-disjoint incidences of \( x_p \) and \( y_q \) within a cycle of the double recurrence of x and y. Thus we will show that the running time of the algorithm is linear in the maximum of the durations. This result is better than the running time of a cyclical exploration in the worst case.

This paper uses the same language of the reified first order logic used in [2] which notation is briefly presented in section 2. The problem of coincidence within the context of double recurrence is formally presented in section 3. Our solution to the problem of coincidence is presented in section 5 makes use of the concept of gcd partitions, which is introduced in section 4. Section 4 presents the key properties of partitions and gcd-partitions that lead to the key theorem about gcd-partitions (i.e. Theorem 5.1) which form the basis of the algorithm in section 6.

The paper ends with an algorithmic analysis of the two main algorithms involved in the proposed solution and compares their performance with the simple temporal projection algorithm proposed in [2].

2. Background

2.1 Notation

The logical language used in this paper is standard many sorted first-order logic with equality. It is exactly the same language used in [2]. Using a logical language here enables us to prove some key results that lead to the algorithm. However, readers who are not interested in the proofs should be able to read the paper while avoiding the logical formalization which is always preceded by a prosaic interpretation. The domains include Eventualities, (interval relations) Intervals, Time Maps and (interval relations) Relations, which will include all of Allen’s relations and others introduced earlier in this paper and the domains of ordinals and natural numbers. A time interval is time unit over which a proposition may be regarded as being either true or false[3, 4]. It connotes the idea of an interval of time. In some literature a time interval is regarded as a pair of time points[5]. A time map is a sequence of contiguous time intervals. Each individual time interval in a time map can be accessed by applying a variety of functions introduced later.

An Eventuality is either an event or a process (simple eventuality) or a sequence of other eventualities (complex eventuality) described by a proposition. An example of a simple eventuality is ‘it is Monday’ or ‘Green light is on’. An example of this is ‘Weekdays’ which is a sequence of ‘Monday’, ‘Tuesday’ etc. Another example is ‘Traffic light’ which is a sequence of Red, Amber, and Green lights. An eventuality is reified in the language. [7]. Each eventuality has an assigned length, which designates the number of component eventualities. There is an ordering among these components. Intuitively, a time-map should be understood as a sequence of contiguous time intervals, so that it is possible identify the order of the intervals in the time map. That is done in this paper by using indices, so that \( tm[k] \) represents the \( k^{th} \) interval in the
time- map. The notation \( \text{tm}[k] \) should be a syntactic denotation for a function taking a time-map and the integer \( k \), and returning a time interval.

We will use the standard logical operators: conjunction \( \land \), disjunction \( \lor \), equivalence \( \leftrightarrow \), implication \( \Rightarrow \) and (unary) negation \( \neg \). The scope of a quantified variable is the rest of the sentence. Whenever necessary, we delimit the scope of a quantified variable by the explicit use of parentheses. The precedence of the logical operators is as assumed for first order predicate logic, i.e. precedence of \( \neg \) is higher than those of \( \land \) and \( \lor \) which is higher than those of \( \leftrightarrow \) and \( \Rightarrow \).

The predicates used in our theory include the standard predicates for associating eventualities with time intervals otherwise known as truth predicates. We use the predicates here as used in [8] i.e. \( \text{TT} \) (true throughout), \( \text{MT} \) (maximal truth) and \( \text{RT} \) (recurrently true). We define the partition relationship, \( \text{Part} \) between an eventuality and a sequence of eventualities. We shall introduce the notion of partitions later. We define interval relation predicate \( \text{Intrel} \), as a relation between an interval, a temporal relation and another interval. This enables us to make interval relations terms, so that we can reason about the similarity of interval relations between two pairs of intervals. The double recurrence predicate \( \text{MRT} \) defines the double recurrence of two eventualities over some interval. The signatures for the predicates are given below:

- \( \text{TT, MT, RT: Eventuality } \times \text{Interval } \rightarrow \text{Boolean} \)
- \( \text{Intrel: Interval } \times \text{Relation } \times \text{Interval } \rightarrow \text{Boolean} \)
- \( \text{MRT: Eventuality } \times \text{Eventuality } \times \text{Interval } \rightarrow \text{Boolean} \)
- \( \text{Aux: Interval } \times \text{Interval } \times \text{Interval } \rightarrow \text{Boolean} \)

Of the truth predicates described above, true over(\( \text{TT} \)) is the most primitive, capturing the notion of an eventuality being True over an interval. The other truth predicates will be defined in terms of \( \text{TT} \) subsequently.

In the rest of the paper, we use the letters \( x, y, w \) and \( z \) without subscripts to represent eventualities. When \( x, y, w, \) and \( z \) are made bold, they represent lists of eventualities of finite length. The symbols \( i, j, k \) and \( l \) with or without suffixes, are used for temporal intervals, while we use \( p, q, r, s, t, u \) as variable ordinals for identifying components of an eventuality. We write \( w_p \) as a shorthand for the application of the component function (just ahead) to obtain the \( p+1 \)th component of the eventualities \( w \). Time maps are denoted by \( \text{tm} \) with or without subscripts.

The list of functions used here and their description is given below.

- \( \text{dur: Interval } \rightarrow \mathbb{N} \): This function returns a natural number signifying the duration of given interval. Even when a natural unit for duration measures the unit of an interval as a real number, as long as all the measured intervals in the domain are rational numbers, the interval durations can be measured in a smaller unit (for example, \( 10^{-6} \)th of a second) that allows durational values to be natural numbers.

- \( \text{cover: Interval } \times \text{Interval } \rightarrow \text{Interval} \)

The cover function (defined in section 2.2) of two meeting intervals is the interval that covers the two intervals. The function is a partial function defined only for any pair of meeting intervals.
cover*: \( Time-Map \rightarrow Interval \)

The cover* function (defined in section 2.2) of a time map is a single interval covering the entire time map.

\( \Pi : Eventuality \rightarrow Interval-Set \) (\( \Pi \) returns a period set)

Usually, we denote a period of occurrence by \( \pi \) with or without suffixes. As such when we write:

\[ \pi \in \Pi(x). \]

We are saying that \( \pi \) is a period of the occurrence of \( x \).

\( \Omega : Eventuality \times Eventuality \rightarrow Interval-set \) (\( \Omega \) returns a cycle set for the double recurrence of the two eventualities)

So that when we write \( \omega \in \Omega(x, y) \), we are saying that \( \omega \) is a cycle of the double recurrence of \( x \) and \( y \).

\( + : Eventuality \times Eventuality \rightarrow Eventuality \)

The function \( + \) returns some eventuality, that corresponds to the two eventualities being true simultaneously.

\( \eta: Eventuality \times Interval \rightarrow Time-Map \)

The function \( \eta \) returns for an eventuality \( x \), the contiguous sequence of intervals which are its periods within the interval, if \( x \) recurs over that interval. It returns an empty list if \( x \) is not recurrently true over the given interval.

Thus \( \eta(x, j) \) returns a time-map of \( x \)'s occurrence within the time interval \( j \).

Index: \( Time-Map \times N \rightarrow Interval \)

The index function returns the p\(^{th}\) interval of a time map, where \( p \) is the given ordinal. We will write \( \text{tm}[p] \) as a short hand for \( \text{index(tm, p)} \).

As a notation \( \eta(x, j)[p] \) refers to the p\(^{th}\) interval over which \( x \) is maximally true with interval \( j \).

\( \text{dim: Time-Map} \rightarrow N \)

This function returns for a given time-map, the maximum ordinal for which index will return an interval.

\( \phi : Eventuality \times Interval \rightarrow Interval \)

The function returns the first sub interval of the given interval over which the eventuality is maximally true i.e. its interval of maximal occurrence within the given interval. If there are more than one such sub intervals, \( \phi \) is undefined.

If \( x \) is a sequence of eventualities \( x_1, x_2, \ldots, x_n \), then \( \phi(x_r, \eta(x, j)[p]) \) refers to the sub interval within the p\(^{th}\) interval of maximum occurrence of \( x \) within \( j \) over which \( x_r \) is maximally true.
len: \textit{Eventuality} \rightarrow \mathbb{N}
The function returns the number of components that make up the eventuality. Intuitively we think of the components of an eventuality being in a sequence.

head: \textit{Time-Map} \rightarrow \textit{Interval}
The function returns the first interval in the time map.

tail: \textit{Time-Map} \rightarrow \textit{Time-Map}
The function returns the list excluding the first element. It returns the value ‘nil’ if the time-map consist of only one interval.

comp: \textit{Eventuality} \times \mathbb{N} \rightarrow \textit{Eventuality}
This partial function returns the component of the given eventuality, whose position is indicated by the ordinal. As we have indicated earlier, we will write \( w_p \) as shorthand for \( \text{comp}(w, p) \).

Thus we proceed to formally state the problem of coincidence within the context of double recurrence.

2. 2 Interval Calculus
Allen [3] formulated the interval calculus, as a means of representing time, along with a computationally efficient reasoning algorithm based on constraint propagation. Allen had argued that time points proposed earlier by McDermott [11], was inadequate.

Allen defined a linear logic of time based on the concept of intervals. In this logic, time intervals are treated as individuals. A number of qualitative binary relations are defined on intervals. The basic ones are:

\[
\begin{align*}
\text{Meets} & \quad \text{Ends} \\
\text{After} & \quad \text{Starts} \\
\text{Contains} & \quad \text{Overlaps} \\
\text{Equals}^1 &
\end{align*}
\]

The names assigned to these relations fit the relationship the reader is likely to assign to them intuitively. For example, an interval \( i \) \textit{Meets} another interval \( j \) if \( i \) starts before \( j \) and ends just when \( j \) is starting so that they have no interval between them, and no subinterval in common.

Based on these interval relations, we define (like Koomen) four additional temporal relations between time intervals: disjoint, within, subinterval and the same-beginning relations.

\textbf{Definition 2.2.1} (Disjoint relation)
\textit{The time intervals} \( j \) \textit{and} \( k \) \textit{are disjoint if and only if either} \( j \) \textit{is after} \( k \), \textit{or} \( k \) \textit{is after} \( j \) \textit{or} \( j \) \textit{meets} \( k \) \textit{or} \( k \) \textit{meets} \( j \).

\( \forall j, k. \text{Intrel}(j \text{ disjoint } k) \iff \text{Intrel}(j, \text{ after }, k) \lor \text{Intrel}(k, \text{ after }, j) \lor \text{Intrel}(k \text{ meets } j) \lor \text{Intrel}(j \text{ meets } k) \)

\( ^1 \) We do not differentiate between the Allen’s Equal relation and the equality predicate (=) in first order predicate calculus.
As usual we will write \( j \) disjoint \( k \) or \( k \) disjoint \( j \). Note that disjoint is a symmetric relation.

**Definition 2.2.2 (Within relation)**

The interval \( j \) is within \( k \) if and only if either \( j \) starts \( k \) or \( j \) ends \( k \) or \( j \) is during \( k \) or \( j \) equals \( k \).

\[
\forall j, k. \text{Intrel}(j \text{ within } k) \Leftrightarrow \text{Intrel}(j, \text{starts}, k) \lor \text{Intrel}(j, \text{ends}, k) \lor \text{Intrel}(k, \text{contains}, j)
\]

Sometimes we find it necessary to refer to the subinterval relation. This relation will be a disjunction of the within relation and equality.

**Definition 2.2.3 (The Subinterval relation)**

An interval is a subinterval of another, if it is either within that interval or equal to it.

\[
\forall j, k. \text{Intrel}(j, \text{sub}, k) \Leftrightarrow \text{Intrel}(j, \text{within}, k) \lor k = j
\]

**Definition 2.2.4 (Samebegin relation)**

The time intervals \( j \) and \( k \) have the same beginning if and only if either \( j \) equals \( k \), or \( k \) starts \( j \) or \( j \) starts \( k \).

\[
\forall j, k. \text{Intrel}(j, \text{samebegin}, k) \Leftrightarrow \text{Intrel}(j, \text{starts}, k) \lor \text{Intrel}(k, \text{starts}, j) \lor k = j
\]

Allen’s interval relations implicitly assume the existence of some other interval apart from the related pair, whose existence is an aspect of the definition of the given relation. For example, the fact that interval \( j \) is before interval \( k \), indicates the existence of some interval meeting \( k \) and met by \( j \). Similarly, the fact that \( j \) starts \( k \) implies the existence of some interval met by \( j \) and ending \( k \). If \( j \) overlaps \( k \) however, there are two auxiliary intervals: One starting \( j \) and meeting \( k \), and another ending \( k \) and met by \( j \). It is the existence of such interval that enabled the definition of all of Allen’s interval relations in terms of the relation meets.

It is also important to note that auxiliary intervals defined for any pair of non-disjoint intervals, exclude any interval shared by both intervals. Those intervals are defined as common intervals as Definition 2.2.5. Thus, here we define formally the concept of an auxiliary interval for two non-disjoint intervals.

**Definition 2.2.5a (Auxiliary Intervals)**

Given any two distinct non-disjoint intervals, an auxiliary interval is any interval, which starts or ends one of the two intervals, and meets or is met by the other.

\[
\forall j, k, m. \neg \text{Intrel}(k, \text{disjoint}, j) \land \text{Aux}(k, j, m) \Leftrightarrow \neg (k = j) \land \\
(\text{Intrel}(m, \text{starts}, j) \lor \text{Intrel}(m, \text{ends}, j) \lor \text{Intrel}(j, \text{starts } m) \lor \text{Intrel}(j, \text{ends } m)) \land \\
(\text{Intrel}(m, \text{meets }, k) \lor \text{Intrel}(k, \text{meets }, m))
\]

**Definition 2.2.5b**

Given any two disjoint non-meeting intervals an auxiliary interval meets one and is met by the other.

\[
\forall j, k, m. \text{Intrel}(k, \text{disjoint}, j) \land \text{Aux}(k, j, m) \Leftrightarrow \\
\neg \text{Intrel}(k, \text{meets }, j) \land \neg \text{Intrel}(j, \text{meets }, k) \land \\
(\text{Intrel}(k, \text{meets } m) \lor \text{Intrel}(m, \text{meets } k)) \land (\text{Intrel}(m, \text{meets }, j) \lor \text{Intrel}(j, \text{meets } m))
\]
From our definition it is intuitively clear that two equal or meeting intervals have no auxiliary intervals. As such

$$\forall j, k. \neg\exists m. (\text{Intrel}(j, \text{meets } k) \lor j = k) \land \text{Aux}(k, j, m)$$

**Definition 2.2.6 (Common Intervals)**

The function common returns for any given non-disjoint interval pair their maximal common sub interval.

$$\forall j, k, m. \neg\text{Intrel}(j, \text{disjoint } k) \Rightarrow$$

$$\text{common}(k, j) = m \Leftrightarrow \text{Intrel}(m, \text{sub, } j) \land \text{Intrel}(m, \text{sub, } k)$$

$$\land (\forall m_1. \text{Intrel}(m, \text{sub, } m_1) \Rightarrow \neg\text{Intrel}(m_1, \text{sub, } j) \lor \neg\text{Intrel}(m_1, \text{sub, } k))$$

Now we define the idea of a time interval that covers two meeting intervals. This concept is referred to in this paper as in [8], as the cover of the two intervals.

**Definition 2.2.7 (The cover of two intervals)**

The cover of two meeting intervals is the interval that is started by the earlier interval and ended by the latter interval.

$$\forall j, k, m. \text{cover}(j, k) = m \Leftrightarrow \text{Intrel}(j, \text{meets, } k) \land \text{Intrel}(j, \text{starts, } m) \land \text{Intrel}(k, \text{ends, } m)$$

The next axiom describes the basic property of time maps.

**Axiom 2.2.8**

Each interval in a time map meets with the next interval in the map.

$$\forall tm, p. 1 \leq p < \text{dim}(tm) \Rightarrow \text{Intrel(tm}[p\text{]} \text{ meets } tm[p+1])$$

Finally, we extend the concept of cover to a time map by defining the unary function cover* for time-maps.

**Definition 2.2.9 (cover * of a time map)**

The partial function cover* (defined for a time map) returns a single interval which covers the entire time map.

$$\forall tm, m. \text{cover*}(tm) = m \Leftrightarrow (\text{dim}(tm) = 1 \land m = tm[1]) \lor$$

$$(\text{dim}(tm) > 1 \land \text{Intrel(tm}[1\text{]} \text{ starts } m) \land \text{Intrel(tm}[\text{dim}(tm)] \text{ ends } m))$$

This definition clearly spells out what the temporal relations between individual eventualities in the time-map tm[p] and its cover, the interval m will be.

### 3. Maximal Truths, Sequences and Recurrences

**Axiom 3.1a (Maximal truth of eventualities of length 1)**

A simple eventuality x is maximally true over an interval j if x is true over j and x is not true over any interval which is a super interval of j, or overlaps or is overlapped by j.
\( \forall x, k. \ len(x) = 1 \Rightarrow \\
(\text{MT}(x, k) \iff \text{TT}(x, k) \land (\forall j. \text{Intrel}(j, \text{overlaps}, k) \lor \text{Intrel}(k, \text{Overlaps} j) \lor \text{Intrel}(k, \text{within}, j) \Rightarrow \neg \text{TT}(x, j))) \)

**Axiom 3.1b** (The truth of eventualities of length greater than 1)

*If an eventuality x of length greater than 1 is true over an interval j, then its first component \( x_0 \) is maximally true over some interval starting subinterval of j, and provided the length is at least p, the p-1th component is maximally true over a subinterval of j over which the kth component is maximally true.*

\( \forall x, k. \ len(x) > 1 \Rightarrow \\
(\text{TT}(x, k) \iff \exists k_1. \text{MT}(x_0, k_1) \land \text{Intrel}(k_1, \text{starts} k) \land \\
(\forall p. 0 < p \leq \text{len}(x) - 1 \Rightarrow \exists k_2, k_3. \text{Intrel}(k_2, \text{within}, k) \land \text{Intrel}(k_3, \text{within}, k) \land \\
\text{MT}(x_p - 1, k_2) \land \text{MT}(x_p, k_3) \land \text{Intrel}(k_2 \text{meets} k_3)) \)

**Axiom 3.1c** (The maximality of truth of eventualities of length greater than 1)

*If an eventuality is of length greater than 1, then it is maximally true over an interval when it is true over the same interval.*

\( \forall x, k. \ len(x) > 1 \Rightarrow \\
\text{MT}(x, k) \iff \text{TT}(x, k) \)

We are treating eventuality sequences as a solid eventuality and as such they are only true over intervals that they are maximally true over.

The next theorem establishes the relationship between truth and maximal truth for all eventualities. It follows from definitions 3.1a and 3.1c.

**Theorem 3.2**

*If an eventuality x is maximally true over an interval j, it is also true over the same interval.*

\( \forall x. \text{MT}(x, j) \Rightarrow \text{TT}(x, j) \)

A basic domain assumption is the fact that a proposition defining an eventuality cannot be true forever. The next Axiom formalizes this domain assumption.

**Axiom 3.3** (Nothing is true forever)

*If an eventuality x is true over an interval j, then j is a part of or equal to some interval k over which x is maximally true.*

\( \forall x, j. \text{TT}(x, j) \Rightarrow \exists k. \text{MT}(x, k) \land (j = k \lor \text{Intrel}(j, \text{within} k)) \)

The next axiom introduces a slight restriction needed to achieve the representation of regular recurrence. While this may slightly limit the applicability of the domain, as we have seen before in the examples in section 1, there are domains where this restriction does not limit.

**Axiom 3.4** (Fixed duration eventualities)

*If an eventuality x is maximally true over two different intervals j and k, then the durations of j and k are equal.*

\( \forall x, j, k. \text{MT}(x, j) \land \text{MT}(x, k) \Rightarrow \text{dur}(j) = \text{dur}(k) \)

Such an assumption will be valid when we are dealing with eventualities with fixed duration of maximal incidence, as opposed to variable durations. All of the results of this paper are based on
this assumption. As such, the results discussed here in section 3 and 4 will be invalid in a domain where there is some fuzziness about how long a state or event may take. An example of fuzziness is when an event takes between 2 and 3 hours to complete. Interestingly, there are applications for which we do not have to contemplate the problem of fuzziness. However we return to the problem of fuzzy durations later.

We now define the truth of an eventuality $x+y$ which is defined as the coincidence of $x$ and $y$. As introduced earlier + when applied to a pair of eventualities returns an eventuality that corresponds to the incidences of the two eventualities sharing a common subinterval. However + can only be formally defined in terms of the truth of the product of its application. That is presented here:

**Definition 3.5 (+ operators for eventualities)**

a. We say that the eventuality $x + y$ is maximally true over an interval if and only if that interval is a common subinterval of two non-disjoint intervals such that $x$ is maximally true over one of the intervals and $y$ is maximally true over the other.

$$\forall x, y, k. \text{MT}(x+y, k) \iff \exists j, i. \text{MT}(x, j) \land \text{MT}(y, i) \land \neg \text{Intrel}(i, \text{disjoint} j) \land k = \text{common}(j, i)$$

b. Eventuality $x + y$ is true over an interval $k$ if and only if $k$ is a subinterval of some other interval over which $x + y$ is maximally true.

$$\forall x, y, k. \text{TT}(x+y, k) \iff \exists j. \text{MT}(x+y, j) \land \text{Intrel}(k, \text{subs}, j)$$

The following Theorem contains some properties of the + operator. All of the properties follow from Definition 3.5.

**Theorem 3.6 (Properties of +)**

a. The operator + is commutative.

$$\forall x, y, k. \text{MT}(x+y, k) \iff \text{MT}(y+x, k)$$

b. For all $x$ and $y$, $x + y$ is downward hereditary irrespective of the Shoham type of $x$ and $y$.

$$\forall x, y, k. \text{MT}(x+y, k) \Rightarrow \forall j. \text{Intrel}(j \text{ within } k) \Rightarrow \text{TT}(x+y, j)$$

c. If $x+y+z$ is maximally true over any interval then $x+y$, $y+z$ and $x+z$ are all true over that interval.

$$\forall x, y, z, j. \text{MT}(x+y+z, j) \Rightarrow \text{MT}(x+y, j) \land \text{MT}(x+z, j) \land \text{MT}(z+y, k)$$

It is appropriate at this point to introduce the $\phi$ (phi) function which returns a unique interval of incidence, within an interval if indeed there is such an interval, and is not defined if there is no such interval or there is more than one such.

**Definition 3.7 (The phi ($\phi$) function)**

Given an eventuality $x$ and interval $k$, the $\phi$ function returns the only (possibly improper) subinterval of $k$, over which $x$ is maximally true. It is undefined either if there is no such interval or there are more than one such interval.
\[ \forall x, k, j. \phi(x, k) = j \Leftrightarrow \exists i. \text{MT}(x, i) \land \text{Intrel}(i, \text{sub}, k) \land j = i \]

Maximal truth also forms the basis for defining recurrence. Although Koomen offered no formal definition of recurrence but had an axiom that states that: if \( x \) recurs over an interval then \( \text{either} \) it is maximally true over the interval \( \text{or} \) it is maximally true over some starting subinterval of the interval, and it recurs over the remaining subinterval of the given interval. However we strengthen that axiom and offer a definition for recurrence thus:

**Definition 3.8 (Recurrence)**

An eventuality is said to recur over an interval if and only if either it is maximally true over the interval or it is maximally true over some starting subinterval of the interval, and it recurs over the remaining subinterval of the given interval.

\[ \forall x, k. \text{RT}(x, k) \iff \text{MT}(x, k) \lor (\exists j, i. \text{Starts}(j, k) \land \text{Ends}(i, k) \land \text{Meets}(j, i) \land \text{MT}(x, j) \land \text{RT}(x, i)) \]

The effect of this strengthening is to make maximal truth a special case of recurrence.

Now we introduce the \( \eta \) function into our formalism. We use the function to refer to the list of intervals over which \( x \) is maximally true, within an interval over which \( x \) recurs.

**Axiom 3.9 (The eta \( \eta \) function)**

Given that \( x \) recurs over an interval \( j \), then, the function \( \eta \) returns a time map of the interval whose cover* is the original interval, and \( x \) is maximally true over each time interval returned by the index function.

\[ \forall x, j. \text{RT}(x, j) \Rightarrow \exists \text{tm}. \eta(x, j) = \text{tm} \land j = \text{cover}^*(\text{tm}) \land (\forall p. 1 \leq p \leq \text{dim}(\text{tm}) \Rightarrow \text{MT}(x, \text{tm}[p])) \]

The eta function \( \eta \) is a partial function that takes an eventuality and an interval over which the eventuality is recurrently true, and returns a time map which contains time intervals over which the eventuality is true throughout. To illustrate the usefulness of eta function in our formalization of recurrence, we can express the fact that if \( x \) is recurrently true over \( k \), then any subinterval \( j \) of \( k \) over which \( x \) is maximally true, is a member of the list returned by \( \eta(x, k) \) by writing:

\[ \forall x, k. \text{RT}(x, k) \Rightarrow (\forall j. \text{MT}(x, j) \land \text{Intrel}(j \text{sub} k) \Rightarrow \exists n_1. \eta(x, k)[n_1] = j) \]

**Definition 3.10 (Double Recurrence)**

We say that \( x \) and \( y \) doubly recur over an interval \( k \) if and only if both \( x \) and \( y \) each recur over \( k \)

\[ \forall x, y, k. \text{MRT}(x, y, k) \Leftrightarrow \text{RT}(x, k) \land \text{RT}(y, k) \]

So for example, to say \( x \) recurs over an interval \( j \) in Koomen’s theory, one writes as a logical formula \( \text{RT}(x, j) \). However, there is often the need to represent the recurrence of two different eventualities over the same interval. For example, to state that eventualities \( x \) and \( z \) recur over the interval \( j \) we will write the formula: \( \text{RT}(x, j) \) and. \( \text{RT}(z, j) \). We will refer to this as a double recurrence of \( x \) and \( z \) over \( j \).
The following is basically true of components of an eventuality because any two of them cannot be maximally incident on two non-disjoint intervals.

**Axiom 3.11 (Mutual exclusion)**

*Any two different components of an eventuality are mutually exclusive in time, i.e. any pair of intervals over which they respectively true, must be disjoint.*

\[ \forall \, x, \, p, \, q, \, j, \, k. \, p \neq q \land \text{TT}(x_p, \, j) \land \text{TT}(x_q, \, k) \Rightarrow \text{Intrel}(k, \, \text{disjoint}, \, j) \]

We proceed to define some properties of recurrence and double recurrence, and some basic truths about them.

### 3.1 Properties of Recurrence and Double recurrence

This section presents basic inferences about the properties of recurrence and double recurrence. The notions of periods of an eventuality and the cycles of a double recurrence are introduced. Their properties are also proved.

**Definition 3.13 (Period of Eventuality)**

A period of an eventuality is an interval over which it is maximally true. We assume that the function \( \Pi \) returns the set of all such intervals for any given eventuality.

\[ \forall x, \, \pi, \, k. \, \pi \in \Pi(x) \Leftrightarrow \text{MT}(x, \, \pi) \]

The following theorem follows from the definitions of recurrence (3.8) and double recurrence (3.9). The proof is trivial and is thus omitted.

**Theorem 3.14**

*If there is a double recurrence of \( x \) and \( y \) over an interval \( k \), then Either there exists two starting sub intervals of \( k \) such that \( x \) is maximally true over one and \( y \) is maximally true over the other and two ending subintervals of \( k \) such that \( x \) is maximally true over one and \( y \) is maximally true over the other Or either \( x \) or \( y \) are maximally true over \( k \).*

\[ \forall k, \, x, \, y. \, \text{MRT}(x, \, y, \, k) \Rightarrow \exists \, j_1, \, j_2, \, j_3, \, j_4. \, \text{Intrel}(j_1, \, \text{starts} \, k) \land \text{Intrel}(j_2, \, \text{starts} \, k) \land \text{MT}(x, \, j_1) \land \text{MT}(y, \, j_2) \land \text{MT}(x, \, j_3) \land \text{MT}(y, \, j_4) \land \text{Intrel}(j_3, \, \text{ends} \, k) \land \text{Intrel}(j_4, \, \text{ends} \, k) \lor (\text{MT}(x, \, k) \lor \text{MT}(y, \, k)) \]

Now we define the cycle of a double recurrence.

**Definition 3.15 (Cycle of a double recurrence \( \Omega \))**

A cycle \( \omega \) of a double recurrence of two eventualities \( x \) and \( y \), is a minimal interval over which \( x \) and \( y \) are doubly recurrent.

\[ \forall x, \, y, \, \omega. \, \omega \in \Omega(x, \, y) \Leftrightarrow \text{MRT}(x, \, y, \, \omega) \land \]
\[ \forall j. \text{Intrel}(j, \text{within}, \omega) \Rightarrow j \notin \Omega(x, y) \]

The following Theorem 3.18 is a direct consequence of our basic assumption i.e. Axiom 3.4 and the Definition 3.13 of periods.

**Theorem 3.16**

*All periods of the same eventualities have the same durations.*

\[ \forall x, \pi_1, \pi_2. \pi_1, \pi_2 \in \Pi(x) \Rightarrow \text{dur}(\pi_1) = \text{dur}(\pi_2) \]

The following theorem is a direct consequence of Axiom 3.4 and the Definition 3.15 of a cycle. Thus every period of x have the same duration. In a similar way all cycles of a recurrence of any pair of eventualities have the duration as presented below

**Theorem 3.17**

*All cycles of double recurrence for the same pairs of eventualities are of the same duration.*

\[ \forall x, y, \omega_1, \omega_2, \omega_1, \omega_2 \in \Omega(x, y) \Rightarrow \text{dur}(\omega_1) = \text{dur}(\omega_2) \]

The following theorem follows from axiom 3.4, a bit of arithmetic reasoning about the durations of x and y. It states that the duration of a cycle is the longest common multiple of the duration of x and y.

**Theorem 3.19**

*The duration of a cycle of double recurrence equals the least common multiple of the periods of the two recurrences.*

\[ \forall x, y, \omega, j, k, n_1, n_2.
\omega \in \Omega(x, y) \land (\text{MT}(x, j) \Rightarrow \text{dur}(k) = n_1) \land (\text{MT}(y, k) \Rightarrow \text{dur}(k) = n_2) \Rightarrow
\text{dur}(\omega) = \text{lcm}(n_1, n_2) \]

The following theorem from the literature[1], presents an interesting result about the temporal relations between each incidences of x\_p and y\_q within each cycle. It states that in each cycle, the relation between the r\textsuperscript{th} incidence of x\_p and the s\textsuperscript{th} incidence of y\_q is always the same in every cycle.

**Theorem 3.20** (Each cycle is exactly like every other)

*For the double recurrence of two sequences x and y, whatever relation holds between interval of incidence of any two members of the sequences x\_p and y\_q, within the r\textsuperscript{th} period of the x-sequence and the s\textsuperscript{th} period of the y-sequence respectively of any cycle, also holds in all other cycles.*

\[ \forall \omega_1, \omega_2, x, y, \omega_1, \omega_2 \in \Omega(x, y) \Rightarrow
\forall p, q, r, s, \text{rel.} 0 \leq p \leq \text{len}(x) - 1 \land 0 \leq q \leq \text{len}(y) - 1 \wedge
\text{Intrel}(\phi(x_p, \eta(x, \omega_1)[r]) \text{rel} \phi(y_q, \eta(y, \omega_1)[s])) \iff
\text{Intrel}(\phi(x_p, \eta(x, \omega_2)[r]) \text{rel} \phi(y_q, \eta(y, \omega_2)[s])) \]
The implication of this theorem is that the existence of a coincidence can be determined by exploring a cycle of a double recurrence because by setting up a projection of a cycle of double recurrence. Our objective in this paper is to present a method of partitions by which coincidence of \( x_p \) and \( y_q \) can be determined by the temporal projection of one period each of \( x \) and \( y \). Those periods are shorter than a cycle of the double recurrence of \( x \) and \( y \). This represents a significant reduction in the search space for the solution. The subject of partitions is addressed in section 4, while section 5 and 6 presents the algorithms.

3.2 Formal statement of the “coincidence” problem and an initial solution
We are now better equipped to define the coincidence problem more formally. The problem statement is presented thus:

Given two sequences of eventualities \( seq(x_0...x_{s-1}) \) and \( seq(y_0...y_{t-1}) \) recurring over some interval \( j \), which is longer than a cycle. For some \( p \) and \( q \) within the bounds \([1..s]\) and \([1..t]\) respectively, can we find some subinterval of \( j \) over which both \( x_p \) and \( y_q \) are true.

Using the logical language we have developed our problem is to infer the truth or falsity of the following assertion for some particular sequences \( x \) and \( y \), ordinals \( p \) and \( q \) within the limits of the ordinals in the sequences and interval \( k \):

\[
\text{Is it the case that } \text{MRT}(x, y, k) \Rightarrow \exists j. \text{Intrel}(j, \text{within}, k) \land \text{TT}(x_p + y_q, j) \? 
\]

The solution to this problem based on the notion of gcd partitions is presented in section 5. Our immediate attention now turns to the task of introducing partitions and gcd partitions.

4. Partitions and GCD- partitions
We begin section by formally introducing the concept of partition for an eventuality, which is central to results presented in this section.

Definition 4.1 (Partition of a sequence eventuality)
For any eventuality \( x \), we define its partition, as any other eventuality, \( y \) which is maximally true over any interval on which \( x \) is maximally true, and such that the duration of any interval over which \( y \) is true equals the duration of any interval over which \( x \) is true.

\[
\forall x, y. \text{Part}(x, y) \iff \\
(\forall k. \text{MT}(x, k) \iff \text{MT}(y, k))
\]

From the above definition, the following theorem follows.

Theorem 4.2
If we define an eventuality \( w \) as a partition of another eventuality \( x \), and \( x \) is true over some interval \( j \), then for any component eventuality of \( x \), there is a component eventuality of \( w \) with which it shares some subinterval of \( j \).

\[
\forall x, y, j, p. \text{Part}(x, w) \land \text{MT}(x, j) \land 0 \leq p \leq \text{len}(x) \Rightarrow \\
\exists k_1, k_2, r. \text{MT}(x_p, k_1) \land \text{MT}(w_r, k_2) \land \text{MT}(x_p + w_r, k) \land \\
\text{Intrel}(k_1 \text{within } j) \land \text{Intrel}(k_2, \text{within}, j) \land k = \text{common}(k_1, k_2)
\]
The following theorem makes clear the fact that quantitative aspects of the temporal relationships within each period of partitioned eventuality are the same.

**Theorem 4.3** (Partitions preserve qualitative temporal relationships)

For any eventuality, x and its partition eventuality z and for any pair of intervals i and j over which x is maximally true, for all appropriate values of p and q, and time interval relation r, if there exists two intervals k and l within i over which x_p and z_q are respectively, maximally true and the temporal relationship between k and l is r, then there exists similar intervals within j, m and n say, over which x_p and z_q are maximally true and the relationship between m and n is r.

\[
\forall x, \pi_1, \pi_2, \pi_1 \in \Pi(x) \land \pi_2 \in \Pi(x) \land \text{Part}(x, z) \\
\Rightarrow \forall p, q, \text{rel.} \\
0 \leq p \leq \text{len}(x)-1 \land 0 \leq q \leq \text{len}(w)-1 \Rightarrow \\
(\text{Intrel}(\phi(x_p, \pi_1) \text{ rel } \phi(z_q, \pi_1)) \Leftrightarrow \text{Intrel}(\phi(x_p \pi_2) \text{ rel } \phi(z_q, \pi_2)) )
\]

Theorem 4.3 follows from Lemma 4.3a when we realize that each period of x is a cycle of the double recurrence of x and z. That is presented as Lemma 4.3b below.

**Lemma 4.3a**

For the double recurrence of two sequences x and y, the relation between interval of incidence of any two members of the sequences x_p and y_q say, within the i^{th} period of the x-sequence and the s^{th} period of the y-sequence respectively, are the same in all cycles.

\[
\forall \omega_1, \omega_2, x, y, \omega_1, \omega_2 \in \Omega(x, y) \\
\Rightarrow \\
\forall p, q, n_1, n_2 \text{ rel.} \ 0 \leq p \leq \text{len}(x)-1 \land 0 \leq q \leq \text{len}(y) -1 \Rightarrow \\
(\text{Intrel}(\phi(x_p, \eta(x, \omega_1)[n_1]) \text{ rel } \phi(y_q, \eta(y, \omega_1)[n_2])) \Leftrightarrow \\
\text{Intrel}(\phi(x_p, \eta(x, \omega_2)[n_1]) \text{ rel } \phi(y_q, \eta(y, \omega_2)[n_2]))
\]

In what follows, we present a theorem that states that what is preserved in the relationships of eventualities and their partition across periods is more than qualitative.

**Theorem 4.4** (Partitions preserve quantitative temporal relationships in periods).

If the eventuality z is a partition of the eventuality x, then for any two pair of non-disjoint intervals of maximal incidences of any component eventualities x_p and z_r happening during two different periods \(\pi_1\) and \(\pi_2\) then it is the case that:

1. For any auxiliary interval for \(\phi(x_p, \pi_1)\) and \(\phi(z_q, \pi_1)\), a similar auxiliary interval exists for \((\phi(x_p, \pi_2)\) and \(\phi(z_q, \pi_2)\) with same duration and similar relations to incidences of \(x_p\) and \(z_r\).
2. The common intervals of the interval pair \((\phi(x_p, \pi_1)\) and \(\phi(z_q, \pi_1)\) ) and the interval pair \((\phi(x_p, \pi_1)\) and \(\phi(z_q, \pi_1)\), have the same duration.
∀x, π₁, π₂, p, q, π₁, π₂ ∈ Π(x) ∧ Part(x, z) ∧ 0 ≤ p ≤ len(x)-1 ∧
0 ≤ q ≤ len(w)-1 ∧ ¬Intrel(x_p disjoint z_q) ⇒
(∀k, rel₁, rel₂. Aux(ϕ(x_p, π₁), ϕ(z_q, π₁), k) ∧
Intrel(k, rel₁, ϕ(z_q, π₁) ∧ Intrel(k, rel₂, ϕ(x_p, π₁)) ⇔
(∃j. Aux(ϕ(x_p, π₂), ϕ(z_q, π₂), j) ∧ Intrel(j, rel₁, ϕ(z_q, π₂)) ∧ Intrel(j, rel₂, ϕ(x_p, π₂))
∧ dur(k) = dur(j) ) ) ∧
dur(common(ϕ(x_p, π₁), ϕ(z_q, π₁))) = dur(common(ϕ(x_p, π₂), ϕ(z_q, π₂)))

The proof of Theorem 4.4 is contained in the appendix.

Given two recurrent eventualities x (of length t) and y (of length u), then the GCD partitions of the sequences is obtained by computing the greatest common divisor (gcd) of the durations of the two eventualities. And thereby creating two partition eventualities: z (of length r) and w (of length s), where the duration of each z_p and w_q equals to the gcd of the duration of the two eventualities, and the duration of the eventuality z is the duration of the eventuality x (z being a partition for x) while the duration of the eventuality w, equals the duration of y(w being a partition for y). Thus r = duration (x)/gcd and s = duration(y)/gcd. We define gcd partitions formally below:

**Definition 4.5 (gcd-partition)**
Given two eventualities x and y, we define the eventualities z and w as the gcd partition of the double sequence, if and only if:

1. The eventualities z of length r and w of length s are partitions respectively of the eventualities x and y.

2. The durations of maximal occurrence of all z_p which are components of the eventuality z and that of all w_q which are components of the eventuality w are all equal to the greatest common divisor (gcd) of durations of maximal occurrence of both the z-sequence and the w-sequence.

∀x, y, w, z. Gpart(x, y, w, z) ⇔ MRT(x, y) ∧ Part(x, w) ∧ Part(y, z) ∧
(∀ p, q. 0 ≤ p ≤ len(w) ∧ 0 ≤ q ≤ len(z) ∧ dur(w_p) = gcd(dur(x), dur(y)) ∧ dur(z_q) = dur(w_p))

Because a gcd-partition contains two individual partitions, each of those partitions shares the properties of Theorems 4.3 and 4.4 for all partitions. The implication of these is that every partition is exactly like every other in terms of the qualitative and quantitative relationships between occurrences of the same eventualities. As such the any interval of incidence of x_p for any given p, and w_q for any q is the same in any period of x. The case is similar for any component of y and its partition z (i.e. y_q and z_s, for any q nd s).

**Theorem 4.6 (Properties of gcd partitions)**

If an eventuality pair (w, z) is a gcd-partition of another eventuality pair (x, y) then every cycle of the double recurrence of x and y, is also a cycle of the double recurrence of w and z.

∀x, y, w, z. Gpart(x, y, w, z) ⇒ (∀ω. ω ∈ Ω(x, y)) ⇔ ω ∈ Ω(w, z)

Note that 4.6b follows from Definitions 4.5 and 4.1.

**Theorem 4.7**
If a pair of eventualities \( z \) (of length \( r \)) and \( w \) (of length \( s \)) is a gcd partition of some other eventuality pair \( x \) and \( y \), then it is the case that there are exactly \( s \) periods of the eventuality \( z \) and \( r \) periods of the eventuality \( w \), within any cycle of their double recurrence.

\[
\forall x, y, w, z. \ Gpart(x, y, w, z) \land \text{len}(w) = s \land \text{len}(z) = r \Rightarrow
\forall \omega, \omega \in \Omega(w, z) \Rightarrow \dim(\eta(z, \omega)) = s \land \dim(\eta(w, \omega)) = r \land \text{gcd}(r, s) = 1
\]

This follows from basic arithmetic and Theorem 3.19. By Theorem 3.19, the duration of a cycle is the lowest common multiple \( \text{lcm} \) of the durations of both the \( z \)-sequence and \( w \)-sequence. Note that the \( \text{lcm} \) of \( \text{dur}(z) \) and \( \text{dur}(w) \) is given by:

\[
\text{lcm} = r.s.g
\]

where \( g \) is the greatest common divisor of the durations of \( z \) and \( w \). Since \( z \) is of length \( r \) thus \( \text{dur}(z) = r.g \) and as such the number of contiguous \( z \) incidences within the cycle is \( \text{lcm}/r.g \) which is \( s \). Thus \( \dim(\eta(z, \omega)) = s \). Similarly, \( \dim(\eta(w, \omega)) = r \).

Thus \( r \) and \( s \) are relatively prime in the sense that their greatest common divisor is 1. Supposing \( r \) and \( s \) have a common divisor that is more than 1, then \( g \) cannot be the greatest common divisor of \( \text{dur}(w) \) and \( \text{dur}(z) \).

5. A Solution based on gcd partitions

In this section we present a solution that is based on temporal projection over two periods of the two eventualities involved in a double recurrence. First we present a fundamental result about gcd partitions that makes this possible. Theorem 5.1 states that for a gcd partition \( w, z \) of a double recurrence \( x, y \) then an interval exists within any cycle in which any pair of components from \( w \) and \( z \) are both maximally true.

**Theorem 5.1**

For any gcd partition \( z \) and \( w \) of some pair of doubly recurrring eventualities \( x \) and \( y \), then within any cycle of their recurrence, and for any pair \( p \) and \( q \), there exists two natural numbers \( n_1 \) and \( n_2 \), such that the interval of occurrence of \( w_p \) within the \( n_1 \)\(^{th} \) period of eventuality \( w \) and that of \( z_q \) within the \( n_2 \)\(^{th} \) period of the eventuality \( z \) are the same.

\[
\forall x, y, w, z. \ Gpart(x, y, w, z)
\Rightarrow \forall \omega, \omega \in \Omega(w, z) \Rightarrow
\exists n_1, n_2. \phi(w_p, \eta(w, \omega)[n_1]) = \phi(z_q, \eta(z, \omega)[n_2])
\]

(The Proof of Theorem 5.1 is in the Appendix)

The implication of Theorem 5.1 is that if we know all the partition elements \( w_r \) (for all values of \( r \)) with which incidence the incidence of \( x_p \) is non-disjoint in any period of \( x \), and we know all the elements of \( z_s \) (for all values of \( s \)) with which incidence the incidence of \( y_q \) is non-disjoint, then all we need do is find any pair of values of \( r \) and \( s \) if they do exist such that the incidences of \( w \) and \( z \) holding over the same interval implies that one would be able to determine through propagation, the fact that \( x_p \) and \( y_q \) will eventually happen over two non-disjoint intervals sometime within the cycle of \( x \) and \( y \).
This is the kind of reasoning a temporal reasoning system is capable of carrying out through constraint propagation as demonstrated by Algorithm 5.2 below. From Theorems 4.3 and 4.4, one only needs to consider one period of each of x and y.

Algorithm 5.2

Function Coincide(x, y: Eventuality, p, q: Integer) : Boolean
begin
/* Relations between occurrences of x and w, in every period of x are the same \forall p, r */
/* Same for y and z. According to Theorems 4.3 and 4.4 */
/* Thus, it suffices to consider any arbitrary period of x and y */
/* $\pi_x$ and $\pi_y$ represent any arbitrary periods of x and y respectively */
if p and q are not with limits of lengths of x and y respectively then
    halt
else;
    create interval network for a period $\pi_x$ of x and w gcd partition w
    create interval network for a period $\pi_y$ of y and gcd partition z
    for all r such that $\phi(w_r, \pi_x)$ is non-disjoint with $\phi(x_p, \pi_x)$
        for all s that $\phi(z_s, \pi_y)$ is non-disjoint with $\phi(y_q, \pi_y)$
            begin
                isolate a network of $\phi(x_p, \pi_x)$, $\phi(w_r, \pi_x)$ and their auxiliary and common intervals
                isolate a network of $\phi(y_q, \pi_y)$, $\phi(z_s, \pi_y)$ and their auxiliary and common intervals
                propagate constraint: $\phi(w_r, \pi_x) = \phi(z_s, \pi_y)$ through isolated interval networks
                if $\phi(x_p, \pi_x)$ and $\phi(y_q, \pi_y)$ are non-disjoint after propagation then
                    return true
                end
            end
    end
return false
end

Later in section 6, we shall present theorems that illustrate how an algorithm can be constructed for the coincidence problem directly from the isolated network without the use of constraint propagation.

However, in section 5.2 below, we discuss real life double recurrence examples for which the partitioning approach is used.

5.2 Real-Life Applications

Example 1

In order to illustrate the usefulness of our results we return to the factory management example we glossed over in the introduction:

A factory uses some machinery, which should be rested adequately on a periodic basis. The machines should be rested for at least three days, within which some routine maintenance should
take place. However, the maintenance engineers will not be available throughout a particular day of the week, say, Wednesday. Is it possible to avoid a situation in which the engineers will not be available on a rest day?

We now present the solution to this problem based on gcd partitions.

Eventualities: There are two major recurring eventualities: machine-status and weekday. These eventualities may be regarded as state eventualities. The weekday eventuality which has the following component eventualities: Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, Sunday. Each of these should be regarded as states. (e.g. the state of being Monday, etc.). The duration of the interval of maximal occurrence for each state is exactly one day or 24 hours.

The machine-status eventuality consists of two component eventualities. The machine works for five days, and then goes into a rest state for three days. This sequence recurs over time. Thus the component eventualities are: Working and Rest. The durations of occurrence for the eventualities are 5 and 3 days respectively. It is important to note here that all the eventualities mentioned here are downward hereditary.

GCD-Partition: The greatest common divisor of the total durations of states in the two sequences (7 and 8 days respectively) is one. The gcd-partition consists of two partitions respectively for each of the eventualities weekday and machine status. The appropriate partition for the first eventuality is an eventuality w having the following component eventualities: w₀, w₁, ..., w₆ each with a duration of occurrence of 1 day, while the second eventuality is z with components z₀, z₁, ..., z₇ also with each eventuality having a duration of occurrence of one day.

The eventuality Wednesday would always be maximally true over an interval that is equal to the interval of maximal truth of the eventuality w₂. The occurrence of Wednesday will always preclude the truth of every other component eventuality of w. Similarly, the intervals of maximal occurrence of z₅, z₆, z₇ would be within the interval of maximal occurrence of the rest state, within any interval of maximal occurrence of the working-rest sequence. The truth of the Rest eventuality would preclude the truth of every other component eventuality of z.

Conclusion: Wednesday is maximally true exactly over any interval on which w₂ is maximally true. Similarly the eventuality rest is maximally true over an interval which also covers intervals of maximal occurrence of the eventualities z₅, z₆, z₇ in that order.

Thus, whenever w₂ and z₅ (or z₆ and z₇) are both maximally true over some subinterval of any cycle of double recurrence of the eventualities Weekday and Machine-status, then the eventuality Wednesday + Rest would be true over some subinterval of that interval.

Since by Theorem 5.1, there must exist some subinterval of any cycle of Weekday and Machine-status over which w₂ and z₅ (or z₆ or z₇) are both maximally true, it follows that within any such cycle, there exists an interval over which Wednesday + Rest is true.

Comments: We observe here that a rest day within a cycle of recurrence may be any day of the week. One may also wish to tinker with a slightly amended version of the problem in which the machinery works for five days and rests for two days.
If the machines started operations on a Monday, then the rest days will always be Saturday and Sunday.

Example 2

In a synchronized factory, there are two production lines running in parallel. The first production line consist of a sequence of three processes $P_1, P_2, P_3$ while the second production line $P_4$ and $P_5$. The Processes $P_1$ and $P_5$ are process that must not coincide, presumably because some factory safety standard will be compromised. (It may be the case that $P_5$ involves some heating process and $P_1$ involves gasoline flushing.) $P_1$ and $P_2$ take 3 minutes each while $P_3$, $P_4$ and $P_5$ take 2 minutes each. Is it possible for $P_1$ and $P_5$ coincide?

Eventualities: There are two major recurring eventualities. We denote them as PL-1 and PL-2. PL-1 consists of the sequence P1, P2, P3, while PL-2 consists of the sequence P4, P5. The total duration of PL-1 is 8 minutes while the duration of PL-2 is 4 minutes.

GCD Partition: The gcd of the durations of PL-1 and PL-2 is 2 minutes. Thus let QL-1 and QL-2 be the gcd partition of PL-1 and PL-2. QL-1 consists of w1, w2, w3 and w4, while QL-2 consists of z1 and z2, each lasting 2 seconds.

Within each period of PL-1 the relationship between the intervals of maximum occurrence of the major eventualities are given thus:

- w1 starts P1
- P1 overlaps w2
- w2 overlaps P2
- w3 ends P2
- w4 = P3

Similarly, within each period of PL-2:

- P4 = z1
- P5 = z2.

By Theorem 5.1 there exists an interval within the cycle of the double recurrence of PL-1 and PL-2, in which $w2$ and $z2$ are both maximally true.

Conclusion: From the relations outlined above, we can conclude that during any interval in which $w2$ and $z2$ coincide, it will be the case that $P5$ overlaps $P2$. As such within any cycle of double recurrence of PL-1 and PL-2, $P2 + P5$ will be true over some subinterval.

6. The Algorithm

Instead of understanding this algorithm strictly in terms of constraint propagation as done in section 5.1 above, one can make the assumption that every pair of incidences of $w_r$ and $z_s$, for each $r$ and $s$ from the gcd-partition, will both be maximally true over some subinterval of any cycle of double recurrences. Consequently, in order to know whether or not $x_p$ and $y_q$ ever coincide within any cycle, we can treat the relations between incidences of $x_p$ and any $w_r$ which are non-disjoint within any period of $x$ and that between incidences of $y_q$ and any $z_s$ which are non-disjoint within any period of $y$, as relations between $x_p$ and the common interval over which
w_t and z_s are both maximally true. If a coincidence happens under these conditions then it will happen in any cycle. Otherwise if we try all possible incidences of w_t and z_s which are non-disjoint with incidences of x_p and y_q within any periods of x and y respectively, and assuming w_t and z_s both be maximally true on the same interval does not lead to the inference of coincidence, then x_p and w_t will never coincide.

As it turns out, it is possible to exhaust all the conditions under which there is a coincidence between x_p and y_q and therefore a decision algorithm can be constructed by exploring 15 different conditions (or thereabout) under which the coincidence of x_p and w_t will happen. Twelve of those possibilities are described in the first six theorems presented in this section and their corollary derived from reversing the pair x_p, w_t and y_q, z_s, in the antecedents of the theorems. There are three other theorems. Those Theorems follow from Theorem 5.1 and Allen’s interval logic.

Theorem 6.1

If within any cycle ω of the double recurrence of x and y with a gcd partition w, z, any occurrence of x_p starts or overlaps some occurrence of w_t and any incidence y_q starts or overlaps z_s, then there exists a subinterval of ω over which an incidence of x_p+ y_q is true.

\[ \exists \omega \forall x, y, p, q, w, z. \]
\[ \text{MRT}(x, y, \omega) \land \omega \in \Omega(x, y) \land \text{Gpart}(x, y, w, z) \land \]
\[ (\exists r, \pi_x. \text{Intrel}(\pi_x, \text{within}, \omega) \land (\text{Intrel}(\phi(w_t, \pi_x) \text{ overlaps } \phi(x_p, \pi_x)) \lor \text{Intrel}(\phi(w_t, \pi_x)) \land \]
\[ (\exists s, \pi_y. \text{Intrel}(\pi_y, \text{within}, \omega) \land (\text{Intrel}(\phi(y_q, \pi_y) \text{ starts } \phi(z_s, \pi_y)) \lor \text{Intrel}(\phi(y_q, \pi_y) \text{ finishes } \phi(z_s, \pi_y)) \lor \text{Intrel}(\phi(y_q, \pi_y) \text{ overlaps } \phi(z_s, \pi_y)) \lor \text{Intrel}(\phi(y_q, \pi_y) \text{ overlapped-by } \phi(z_s, \pi_y))) \Rightarrow \]
\[ \exists k. \text{Intrel}(k, \text{subs}, \omega) \land \text{MT}(x_p + y_q, k) \]

Theorem 6.2

If within any cycle ω of the double recurrence of x and y with a gcd partition w, z, any occurrence of x_p finishes some occurrence of w_t and any incidence of y_q starts z_s and the sum of the durations of the incidences of x_p and y_q is greater than the gcd of incidences of x and y, then there exists a subinterval of ω over which an occurrence of x_p+ y_q is true.

\[ \exists \omega \forall x, y, p, q, w, z. \]
\[ \text{MRT}(x, y, \omega) \land \omega \in \Omega(x, y) \land \text{Intrel}(\omega \text{ subs } j) \land \text{Gpart}(x, y, w, z) \land \]
\[ (\exists r, \pi_x. \text{Intrel}(\pi_x, \text{within}, \omega) \land (\text{Intrel}(\phi(x_p, \pi_x) \text{ finishes } \phi(w_t, \pi_x)) \land \]
\[ (\exists s, \pi_y. \text{Intrel}(\pi_y, \text{within}, \omega) \land (\text{Intrel}(\phi(y_q, \pi_y) \text{ starts } \phi(z_s, \pi_y)) \land \]
\[ \text{dur}(\phi(x_p, \pi_x)) + \text{dur}(\phi(y_q, \pi_y)) > \text{gcd}(\text{dur}(\pi_x), \text{dur}(\pi_y)) \Rightarrow \]
\[ \exists k. \text{Intrel}(k, \text{subs}, \omega) \land \text{MT}(x_p + y_q, k) \]

Theorem 6.3

If within any cycle ω of the double recurrence of x and y which happens within an interval j with a gcd partition w, z, if an incidence of x_p starts an incidence of w_t and an incidence of y_q is contained in an incidence of z_s and the incidences of y_q and z_s have an auxiliary interval
that meets the incidence of $y_q$ that is shorter than the incidence of $x_p$ then there is a coincidence between $x_p$ and $y_q$ within the cycle $\alpha$.

$$\exists \omega \forall x, y, p, q, w, z. \quad \text{MRT}(x, y, \omega) \land \omega \in \Omega(x, y) \land \text{Gpart}(x, y, w, z) \land$$
$$\exists r, \pi_x. \text{Intrel}(\pi_x, \text{within}, \omega) \land \text{Intrel}(\phi(x_p, \pi_x), \text{starts}, \phi(w_r, \pi_x)) \land$$
$$\exists s, \pi_y. \text{Intrel}(\pi_y, \text{within}, \omega) \land \text{Intrel}(\phi(z, \pi_y), \text{contains}, \phi(y_q, \pi_y)) \land$$
$$\exists j. \text{Aux}(\phi(z, \pi_y), \phi(y_q, \pi_y), j) \land \text{Intrel}(j, \text{meets}, \phi(y_q, \pi_y)) \land$$
$$\text{dur}(j) < \text{dur}(\phi(x_p, \pi_x)) \implies$$
$$\exists k. \text{Intrel}(k, \text{subs}, \omega) \land \text{MT}(x_p + y_q, k)$$

**Theorem 6.4**

*If within any cycle $\omega$ of the double recurrence of $x$ and $y$ with a gcd partition $w, z$, if an incidence of $x_p$ finishes an incidence of $w$, and an incidence of $y_q$ is contained in an incidence of $z_s$ and the incidences of $y_q$ and $z_s$ have an auxiliary interval that is met by the incidence of $y_q$ that is shorter than the incidence of $x_p$ then there is a coincidence between $x_p$ and $y_q$ within the cycle.*

$$\exists \omega \forall x, y, p, q, w, z. \quad \text{MRT}(x, y, \omega) \land \omega \in \Omega(x, y) \land \text{Gpart}(x, y, w, z) \land$$
$$\exists r, \pi_x. \text{Intrel}(\pi_x, \text{within}, \omega) \land \text{Intrel}(\phi(x_p, \pi_x), \text{finishes}, \phi(w_r, \pi_x)) \land$$
$$\exists s, \pi_y. \text{Intrel}(\pi_y, \text{within}, \omega) \land \text{Intrel}(\phi(z, \pi_y), \text{contains}, \phi(y_q, \pi_y)) \land$$
$$\exists j. \text{Aux}(\phi(z, \pi_y), \phi(y_q, \pi_y), j) \land \text{Intrel}(\phi(y_q, \pi_y), \text{meets}, j) \land$$
$$\text{dur}(j) < \text{dur}(\phi(x_p, \pi_x)) \implies$$
$$\exists k. \text{Intrel}(k, \text{subs}, \omega) \land \text{MT}(x_p + y_q, k)$$

**Theorem 6.5**

*If within any cycle $\omega$ of the double recurrence of $x$ and $y$ with a gcd partition $w, z$, if an incidence of $x_p$ overlaps an incidence of $w$, and an incidence of $y_q$ is contained in an incidence of $z_s$ and the incidences of $y_q$ and $z_s$ have an auxiliary interval that meets the incidence of $y_q$ that is shorter than the common interval of the incidences of $x_p$ and $w_r$, then there is a coincidence between $x_p$ and $y_q$ within the cycle.*

$$\exists \omega \forall x, y, p, q, w, z. \quad \text{MRT}(x, y, \omega) \land \omega \in \Omega(x, y) \land \text{Gpart}(x, y, w, z) \land$$
$$\exists r, \pi_x. \text{Intrel}(\pi_x, \text{within}, \omega) \land \text{Intrel}(\phi(x_p, \pi_x), \text{overlaps}, \phi(w_r, \pi_x)) \land$$
$$\exists s, \pi_y. \text{Intrel}(\pi_y, \text{within}, \omega) \land \text{Intrel}(\phi(z, \pi_y), \text{contains}, \phi(y_q, \pi_y)) \land$$
$$\exists j. \text{Aux}(\phi(z, \pi_y), \phi(y_q, \pi_y), j) \land \text{Intrel}(j, \text{meets}, \phi(y_q, \pi_y)) \land$$
$$\text{dur}(j) < \text{dur}(\text{common}(\phi(x_p, \pi_x), \phi(w_r, \pi_x))) \implies$$
$$\exists k. \text{Intrel}(k, \text{subs}, j) \land \text{MT}(x_p + y_q, k)$$

**Theorem 6.6**

*If within any cycle $\omega$ of the double recurrence of $x$ and $y$, with a gcd partition $w, z$, if an incidence of $w$ overlaps an incidence of $x_p$ and an incidence of $y_q$ is contained in an incidence of $z_s$ and the incidences of $y_q$ and $z_s$ have an auxiliary interval that the incidence of $y_q$ meets*
and is shorter than the common interval of the incidences of \( x_p \) and \( w_r \), then there is a coincidence between \( x_p \) and \( y_q \) within the cycle.

\[
\exists \omega \, \forall x, y, p, q, w, z.
\begin{align*}
MRT(x, y, \omega) & \land \omega \in \Omega(x, y) \land \text{Intrel}(\omega \text{ subs } j) \land \text{Gpart}(x, y, w, z) \land \\
\exists r, \pi_x. \text{Intrel}(\pi_x, \text{within}, \omega) & \land \text{Intrel}(\phi(w_r, \pi_x), \text{overlaps}, \phi(x_p, \pi_x)) \land \\
\exists s, \pi_y. \text{Intrel}(\pi_y, \text{within}, \omega) & \land ( \text{Intrel}(\phi(z_s, \pi_y), \text{contains}, \phi(y_q, \pi_y)) \land \\
\exists j. \text{Aux}(\phi(z_s, \pi_y), \phi(y_q, \pi_y), j) & \land \text{Intrel}(\phi(y_q, \pi_y), \text{meets}, j) \land \\
dur(j) & < \text{dur(\text{common}(\phi(x_p, \pi_x), \phi(w_r, \pi_x)))} \\
\Rightarrow \quad & \exists k. \text{Intrel}(k, \text{subs}, \omega) \land \text{MT}(x_p + y_q, k)
\end{align*}
\]

It is important to note that for Theorems 6.1 to 6.6 above, we can create corollaries in which the pair \( x_p \) and \( y_q \), the pair \( w_r \) and \( z_s \), as well as the pair \( \pi_x \) and \( \pi_y \) are all exchanged in the antecedent part of the implication operator, \( \Rightarrow \). In every single case, the consequent part remains the same. This suggests that all these theorems have some sort of commutative properties.

On the other hand, for all the following Theorems 6.7 to 6.9, the preconditions were simple enough to enable the inclusion of commutative conditions in the antecedents.

**Theorem 6.7**

If within any cycle \( \omega \) of the double recurrence of \( x \) and \( y \) with a \( \text{gcd} \) partition \( w, z \), if an incidence of \( x_p \) starts an incidence of \( w_r \) and an incidence of \( y_q \) starts an incidence of \( z_s \), OR if an incidence of \( x_p \) finishes an incidence of \( w_r \) and an incidence of \( y_q \) finishes an incidence of \( z_s \), then there is a coincidence between \( x_p \) and \( y_q \) within the cycle.

\[
\exists \omega \, \forall x, y, p, q, w, z.
\begin{align*}
MRT(x, y, \omega) & \land \omega \in \Omega(x, y) \land \text{Gpart}(x, y, w, z) \land \\
\exists r, s, \pi_x, \pi_y. \text{Intrel}(\pi_x, \text{within}, \omega) & \land \text{Intrel}(\pi_y, \text{within}, \omega) \land \\
\text{(Intrel}(\phi(x_p, \pi_x), \text{starts}, \phi(w_r, \pi_x)) & \land \\
\text{Intrel}(\phi(y_q, \pi_y), \text{starts}, \phi(z_s, \pi_y)) \lor \\
\text{(Intrel}(\phi(x_p, \pi_x), \text{finishes}, \phi(w_r, \pi_x)) & \land \\
\text{Intrel}(\phi(y_q, \pi_y), \text{finishes}, \phi(z_s, \pi_y)) \land \\
\Rightarrow \quad & \exists k. \text{Intrel}(k, \text{subs}, \omega) \land \text{MT}(x_p + y_q, k)
\end{align*}
\]

**Theorem 6.8**

If within any cycle \( \omega \) of the double recurrence of \( x \) and \( y \) with a \( \text{gcd} \) partition \( w, z \), if an incidence of \( x_p \) is contained an incidence of \( w_r \) and an incidence of \( y_q \) is contained in an incidence of \( z_s \), and the incidences of \( x_p \) and \( w_r \), have an auxiliary interval that meets the incidence of \( x_p \) and similarly, and the incidences of \( y_q \) and \( z_s \) have an auxiliary interval that meets the incidence of \( y_q \). Let \( j \) and \( k \) be auxiliary intervals such that \( j \) is an auxiliary of the incidences of \( x_p \) and \( w_r \) that meets the incidence of \( x_p \) and \( k \) is an auxiliary interval of the incidences of \( y_q \) and \( z_s \) which meets the incidence of \( y_q \). Then the only condition left for the incidences of \( x_p \) and \( y_q \) to coincide under these circumstances is if either duration of \( j \) is between that of \( k \) and the sum of the durations of \( k \) and that of \( y_q \), or the duration of \( k \) is between that of \( j \) and the sum of the durations of \( j \) and that of \( x_p \).
\exists \omega \forall x, y, p, q, w, z.
MRT(x, y, \omega) \land \omega \in \Omega(x, y) \land Gpart(x, y, w, z) \land
\exists r, \pi_x. \text{Intrel}(\pi_x, \text{within}, \omega) \land \text{Intrel}(\phi(w_r, \pi_x) \text{ contains } \phi(x_p, \pi_x)) \land
\exists s, \pi_y. \text{Intrel}(\pi_y, \text{within}, \omega) \land \text{Intrel}(\phi(z_s, \pi_y), \text{contains}, \phi(y_q, \pi_y)) \land
\exists j, k. \text{Aux}(\phi(w_r, \pi_x), \phi(x_p, \pi_x), j) \land \text{Intrel}(j, \text{meets}, \phi(x_p, \pi_x)) \land
\text{Aux}(\phi(z_s, \pi_y), \phi(y_q, \pi_y), k) \land \text{Intrel}(k, \text{meets}, \phi(y_q, \pi_y)) \land
\text{(dur}(k) \leq \text{dur}(j) < \text{dur}(k) + \text{dur}(\phi(y_q, \pi_y)) \lor
\text{dur}(j) \leq \text{dur}(k) < \text{dur}(j) + \text{dur}(\phi(x_p, \pi_x)) \Rightarrow
\exists m. \text{Intrel}(m, \text{subs}, \omega) \land \text{MT}(x_p + y_q, m)
\)

**Theorem 6.9**

*If within any cycle \( \omega \) of the double recurrence of \( x \) and \( y \) which happens within an interval \( j \) with a gcd partition \( w, z \), any incidence of \( x_p \) is a super-interval of some incidence of \( w \), and any incidence of \( y_q \) is a super-interval by \( z_s \), then there exists a subinterval of \( j \) over which an incidence of \( x_p + y_q \) is true.*

\exists \omega \forall x, y, p, q, w, z.
MRT(x, y, \omega) \land \omega \in \Omega(x, y) \land Gpart(x, y, w, z) \land
\exists r, \pi_x, \pi_y. \text{Intrel}(\pi_x, \text{within}, \omega) \land \text{Intrel}(\pi_y, \text{within}, \omega) \land
\text{(Intrel}(\phi(w_r, \pi_x), \text{subs}, \phi(x_p, \pi_x)) \lor
\exists s_1, \exists s_2. \text{Intrel}(\phi(z_s, \pi_y), \text{subs}, \phi(y_q, \pi_y)) \land
\Rightarrow
\exists k. \text{Intrel}(k, \text{subs}, \omega) \land \text{MT}(x_p + y_q, k)

From the antecedents of all these theorems we now construct a different rendition of the algorithm that detects coincidence which carries out an optimization. In the algorithm, \( A_{6.1-6.6} \) denotes the disjunction of the antecedents of Theorems 6.1 to 6.6, while Commutative\( (A_{6.1-6.6}) \) denotes the disjunction of the commutative versions of Theorem 6-1 to 6.6. A similar interpretation should be given to \( A_{6.7-6.8} \).

The optimized version of the Coincide function presented earlier as Algorithm 5.2 is now presented as Algorithm 6.1 below.

**Algorithm 6.10**

```
begin
  \text{gcd} = \text{greatest_common_division}(\text{dur}(x), \text{dur}(y))
  \text{if } p \text{ or } q \text{ are not with limits of lengths of } x \text{ and } y \text{ respectively then halt} 
  /* create interval network for an incidence of } y_q \text{ and incidences of elements of the gcd-partition } w \text{ within a period } \pi, \text{ that are non-disjoint, returning true if some } z_s \text{ is found to be a subinterval of } y_q */
  \text{reset the coincidence_flag}
```

/*coincidence_flag is a global variable*/
Create_network(y, z, q)
if coincidence_flag then return true
/* create interval network for an incidence of x_p and incidences of elements of the
gcd-partition z within a period x that are non-disjoint, returning true if some w_r
is found to be a subinterval of x_p */
reset the coincidence_flag
Create_network(x, w, p)
If coincidence_flag then return true
for all r such that \( \phi(w_r, \pi_x) \) is non-disjoint with \( \phi(x_p, \pi_x) \)
for all s that \( \phi(z_s, \pi_x) \) is non-disjoint with \( \phi(y_q, \pi_y) \)
if A_{6.1-6.6} or Commutative(A_{6.1-6.6}) or A_{6.7-6.8} then
return true
end
return false
end

This algorithm starts by creating an interval network around an incidence of y_q in any arbitrary
period of y, and the incidences of components of its gcd partition z with which it is non-disjoint.
In addition, this network creation procedure looks out to know if the incidence of y_q is a super-
interval of a component of z, and if it is sets a coincidence flag which helps conclude
the occurrence of a coincidence of the incidences of x_p and y_q in the main procedure (according to
Theorem 6.9). A similar network is built for x_p. The algorithm for the network creation task is in
Appendix B.

The second part of the algorithm focuses on applying Theorems 6.1 to 6.8. In the double for
loop, each of the incidences of components of w and z in the networks of x_p and y_q respectively
are examined in the light of Theorems 6.1 to 6.8, in order to determine coincidence. Once any of
the antecedents of any of these theorems is found to be true, then coincidence is determined. By
checking Theorem looking out for the antecedent of 6.9 while creating the network, we simplify
the time complexity of the embedded loops. Thus the number of iterations for each loop is not
more than 2, because we have first taken care of the case in which x_p’s incidence shares common
intervals with more than two intervals of the gcd-partition entity’s intervals by first addressing a
situation in which the interval of occurrence of x_p or y_q is a super-interval of a gcd-partition’s
entity. Thus the running time of the loops has a constant upper bound.

6.1 Algorithm Analysis

We start with a result on the asymptotic running time of Algorithm 6.1 presented as Lemma
6.1.1.

Lemma 6.1.1
The worst case running time of Algorithm 6.1 runs in a time that is linear in the maximum of the
durations of x and y i.e. \( O(\max(\text{dur}(\pi_x), \text{dur}(\pi_y))) \).

There are three basic tasks involved in the algorithm. The first task is finding the greatest
common divisor of the durations of x and y. The running time of typical gcd algorithms is
\( O((\lg(\max(\text{dur}(\pi_x), \text{dur}(\pi_y))))^2) \).

The second task is that of setting up the temporal networks for incidences of x_p and y_q within
respective cycles of x and y as done by the create-network algorithm in Appendix B. The two
calls to create networks run in times $O(len(x) + dur(\pi_x))$. Therefore the two calls to create-network have a worst case running time of:

$$O(max(len(x), len(y)) + max(dur(\pi_x), dur(\pi_y)))$$

Realizing that the length of any eventuality is usually smaller than its duration, (i.e. $len(x) \leq dur(x)$) then it is clear that the running time is $O(max(dur(\pi_x), dur(\pi_y)))$. That time is more asymptotically significant than the time it takes to find the gcd of the durations of $x$ and $y$.

Finally the task of determining coincidence from the networks set up for $x_p$ and $y_q$ is constant as argued by section 5. Hence the running time of this algorithm is linear in the maximum of the duration of $x$ and $y$.

On the other hand an algorithm based on temporal projection over a cycle of double recurrence has a worst case running time that is quadratic as stated in this lemma.

**Lemma 6.1.2**

A coincidence algorithm based on a temporal projection over a cycle of double recurrence has a worst case running time that is quadratic in the durations of $x$ and $y$.

In the worst case, the length of every eventuality is equal to its duration, and the gcd of the durations of $x$ and $y$ is 1. In that case the running time of an algorithm that does such a projection will be $O(dur(\pi_x)^n \cdot dur(\pi_y))$, because the duration of a cycle of $x$ and $y$ is the product of the duration of the periods of $x$ and $y$.

**7. Summary and Conclusions**

We have presented, a logical formalization for reasoning about double recurrence, and thereby presented a solution to the coincidence problem. The importance of the coincidence problem has been discussed. An algorithm that detects coincidence based on the property of gcd-partitions of double recurrences presented as Theorem 5.1 has been presented as algorithm 6.1. Theorem 5.1 states that every pair of components from the gcd-partition ($w$ and $z$ respectively), will maximally coincide within any cycle. The algorithm 6.1 then determines coincidence of $x_p$ and $y_q$ within a cycle, by finding a pair of the components of the gcd partition of $x$ and $y$ whose maximal coincidence imply that the incidences of $x_p$ and $y_q$ will coincide. The solution presented here is shown to be more efficient than the one presented in [2] in the worst case.

It is important to re-emphasize here that the eventualities considered here have definite as opposed to variable or fuzzy durations. A fuzzy domain will lead to a problem that is non-deterministic. Thus an average solution will be a reasonable approximation to the solution. In that case, the average durations of all eventualities are taken to be the definite durations.

Finally, this paper has addressed the coincidence problem within double recurrences for the case where the number of recurring eventualities is just two. A natural extension to this problem is when there are more than two sequences where one needs to find out the coincidence of more than two eventualities. This is an interesting direction for future work. It only remains to be seen how often such double recurrence problems may arise in practice.
References

1. B.O. Akinkunmi, A.O. Osofisan, Temporal properties of repetitive entities, *Artificial Intelligence Review* **22**(4), 297–338, December 2004

2. B.O. Akinkunmi The Problem of Coincidence in a Theory of Temporal Double recurrence. *Journal of Applied Logic*, **15**:46-68, May 2016.

3. J. F. Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*. **26**(11): 832-843, 1983

4. J. F. Allen. Towards a general theory of action and time. *Artificial Intelligence*, **23**(2) : 123-154, 1984.

5. T.L. Dean and D.V. McDermott. Temporal Database Management. *Artificial Intelligence* **32**(2):1-55, 1987

6. A. Galton Eventualities In M. Fisher, D. Gabbay and L Vila (eds) Handbook of Temporal Reasoning in Artificial Intelligence, pp 25-58, Elsevier (2005)

7. A. Galton. Operators vs Arguments: The Ins and Outs of Reification. *Synthese*, **150**(3): 415-441, 2006.

8. Johannes A.G.M..Koomen. *Reasoning about Recurrence*. Ph.D. thesis, Department of Computer Science, University of Rochester, 1989.

9. Johannes A. G. M. Koomen, Reasoning about recurrence, *International Journal of Intelligent Systems* **6**(5): 461-496 (1991)

10. D.H. Lehmer. Mathematical methods in Large-scale Computing Units. *Harvard University Computation Laboratory Annals*, **26**:141-146, 1951.

11. D. McDermott, A temporal logic for reasoning about processes and plans. *Cognitive Science*, **6** pp. 101-155, 1982.

Appendix A (Proofs)

Proof of Theorem 4.4

The Proof of Theorem 4.4 depends on two lemma 4.4a and lemma 4.4b below.

Lemma 4.4a

For any complex eventuality sequence x, the durations of incidence of any incidence of its sub-sequence within any two different periods of incidence of x are equal.
Lemma 4.4b

If there are intervals \( i, j, k \) and \( m \) such that \( i \) and \( j \) have the same beginning and \( i \) meets \( k \) while \( j \) meets \( m \). Assume there are corresponding intervals \( i_1, j_1, k_1, m_1 \) such that \( i_1 \) and \( j_1 \) have the same beginning \( i_1 \) meets \( k_1 \) while \( j_1 \) meets \( m_1 \). If the durations of each pair of corresponding intervals are equal then the following are true:

1. For any auxiliary intervals \( l \) for the interval pair \( (k, m) \), there is a similar auxiliary interval \( l_1 \) for the pair \( (k_1, m_1) \) whose relations to \( k_1 \) and \( m_1 \) are exactly the same as those of \( l \) to \( k \) and \( m \) respectively and the durations of \( l \) and \( l_1 \) are the same.
2. If there is a common interval between \( k \) and \( m \), then there is a common interval between \( k_1 \) and \( m_1 \) such that both have the same duration.

Proof of Theorem 4.4

Let \( \pi_1, \pi_2 \in \Pi(x) \) and \( \text{Part}(x, z) \) and for some \( p \) and \( q \) let \( 0 \leq p \leq \text{len}(x) - 1 \) and \( 0 \leq q \leq \text{len}(w) - 1 \) and \( \text{Intrel}(x_q \text{ disjoint } z_q) \)

1. \[ \text{dur}((\text{cover}^*([\phi(x_u, \pi_1)]_{u=1}^{u=p-1})) = \text{dur}(\text{cover}^*([\phi(x_u, \pi_2)]_{u=1}^{u=p-1})) \] Lemma 4.4a
2. \[ \text{dur}(\text{cover}^*([\phi(z_u, \pi_1)]_{u=1}^{u=q-1})) = \text{dur}(\text{cover}^*([\phi(z_u, \pi_2)]_{u=1}^{u=q-1})) \] Lemma 4.4a
3. \( \text{Samebegin}(\phi(x_1, \pi_1), \phi(z_1, \pi_1)) \) and \( \text{Samebegin}(\phi(x_1, \pi_2), \phi(z_1, \pi_2)) \) (Definition of Partition and MT)
4. \[ \text{Intrel}(\phi(x_1, \pi_1), \text{starts} , \phi(x_1, \pi_2)) \] and \( \text{cover}^*([\phi(x_u, \pi_1)]_{u=1}^{u=p-1}) \) \( \text{Intrel}(\phi(z_1, \pi_2), \text{starts} , \phi(z_1, \pi_2)) \) (Definition of MT)
5. \[ \text{Samebegin}(\text{cover}^*([\phi(x_u, \pi_1)]_{u=1}^{u=p-1}), \text{cover}^*([\phi(x_u, \pi_2)]_{u=1}^{u=p-1})) \] (4 and 5)
6. \[ \text{cover}^*([\phi(z_u, \pi_1)]_{u=1}^{u=p-1}), \text{cover}^*([\phi(z_u, \pi_2)]_{u=1}^{u=p-1})) \] (4 and 5)
7. \[ \text{Intrel}(\text{cover}^*([\phi(x_u, \pi_1)]_{u=1}^{u=p-1}), \text{meets} , \phi(x_p, \pi_1)) \]
In other words \( f(n^1) \) is the case that \( z \) and \( w \), i.e.

\[
\phi(x, o)(I) = \phi(z, o)(I),
\]

\[
\phi(w, o)(I) = \phi(z, o)(I),
\]

\[
\phi(w, o)(I) = \phi(z, o)(I),
\]

\[
\phi(w, o)(I) = \phi(z, o)(I),
\]

From that pattern above one concludes for any interval \( k \) and integer \( n \) within the bounds \([1..r]\) that if \( k = \phi(w, o)(n) \) then there exists \( n \) within the bounds \([1..s]\) such that \( \phi(z, o)(n) = k \), where \( f(n, p) \) is given by:

\[
f(n, p) = p \text{ if } n = 1
\]

\[
f(n, p) = f(n-1, p) + r \mod s \text{ otherwise}
\]

These two equations refer to a Linear Congruential Generator, belonging to a class of random number generators, which is a generalization of Lehmer’s class of generators [10]. Thus, by the following criteria:

1. The \( \gcd \) of \( r \) and \( s \) is 1 (by Theorem 4.7)
2. Let \( a \) be the coefficient \( f(n, p) \) then \( a-1 = 0 \) and it is divisible by all integers.

It is the case that \( y_p \) has the maximum period of \( s \). In other words \( f(n, p) \) generates different numbers within the bounds \([1..s]\) given the sequence of \( n \) values \([1..r]\).

Because there are \( r \) periods of the eventualty \( z \) within the cycle \( o \), then for any \( p \) there exists an interval over which \( x_p \) is true and \( z_q \) is also true for all possible \( q \).

**Appendix B (The Create Algorithms)**

Create_network(x, w, p)
begin
k = 1
j = 1
cumdurx = 0
cumdury = gcd
/*This loop moves forward to \( x_p \), the eventuality of interest */
while j < p do
    if cumdurx + x.dur[j] = k*gcd then begin
        inc j, inc k,
cumdurx = cumdurx + x.dur[j]
    end
    else if cumdurx + x.dur[j] > k*gcd then inc k
    else begin
        inc j
        cumdurx = cumdurx + x.dur[j],
    end
/* Setting a network of non-disjoint intervals around \( \phi(x_p, \pi_x) \) */
while j = p and not coincidence-flag do
    if (k-1)*gcd < cumdurx then
        if k * gcd < cumdurx + x.dur[p] then begin
            assert Intrel(\( \phi(w_k, \pi_x) \), overlaps,\( \phi(x_p, \pi_x) \))
        end
        else if k * gcd = cumdurx + x.dur[p] then begin
            assert Intrel(\( \phi(x_p, \pi_x) \), finishes, \( \phi(w_k, \pi_x) \))
        end
    else begin
        assert Intrel(\( \phi(w_k, \pi_x) \), contains,\( \phi(x_p, \pi_x) \))
    end
/* Storing information about the only auxiliary intervals for these two intervals above */
new = make label
assert Intrel(new, starts, \( \phi(w_k, \pi_x) \)) \& Intrel(new, meets, \( \phi(x_p, \pi_x) \))
\& dur(new) = comdurx - k*gcd
new = make label
assert intrel(new, ends, \( \phi(x_p, \pi_x) \)) \& Intrel(new, meets \( \phi(w_k, \pi_x) \), meets new) \&
dur(new) = cumdurx + x.dur[p] - (k+1)*gcd \&
dur(common(\( \phi(x_p, \pi_x) \), \( \phi(w_k, \pi_x) \)) ) = (k+1)*gcd - comdurx
inc k end
else if k * gcd = cumdurx + x.dur[p] then begin
    assert Intrel(\( \phi(x_p, \pi_x) \), finishes, \( \phi(w_k, \pi_x) \))
/* Storing information about the two auxiliary intervals for these two intervals above */
new = make label
assert Intrel(new, starts, \( \phi(w_k, \pi_x) \)) \& Intrel(new, meets, \( \phi(x_p, \pi_x) \))
\& dur(new) = comdurx - k*gcd,
inc j end
else begin
    assert Intrel(\( \phi(w_k, \pi_x) \), contains, \( \phi(x_p, \pi_x) \))
/* Storing information about the two auxiliary intervals for the two intervals above */
new = make label
assert Intrel(new, starts \( \phi(w_k, \pi_x) \)) \&
Intrel(new, meets, \( \phi(x_p,\pi_x) \)) \&
dur(new) = comdurx - k*gcd
new = make label
assert Intrel(\( \phi(x_p, \pi_x) \), meets new) \&
Intrel(new finishes \( \phi(w_k,\pi_x) \)) \&
dur(new) (k+1)*gcd

32
inc j end
else /* if (k-1)*gcd = cumdurx then */
    if k*gcd = cumdurx + x.dur[p] then begin
        Inrel(\(\phi(w_k, \pi_k)\), equals, \(\phi(x_p, \pi_x)\)).
/* There are no auxiliary intervals here. More importantly, we do not need them */
        set coincidence_flag
        inc j end
else if k*gcd < cumdurx + x.dur[p] then begin
    assert Inrel(\(\phi(w_k, \pi_k)\), starts, \(\phi(x_p, \pi_x)\))
/* There is an auxiliary interval here, but we do not need it. There is coincidence */
    set coincidence_flag
    inc k end
else begin assert Inrel(\(\phi(x_p, \pi_x)\), starts, \(\phi(w_k, \pi_k)\))
/* Storing information about the one auxiliary interval for these two intervals above */
    new = make label
    assert Inrel(new, finishes, \(\phi(w_k, \pi_k)\)) \&
    Inrel(\(\phi(x_p, \pi_x)\), meets new) \&
    dur(new) = (k+1)*gcd - cumdurx
    inc j end

return
end