

**SOFTWARE VERIFICATION RESEARCH CENTRE
SCHOOL OF INFORMATION TECHNOLOGY
THE UNIVERSITY OF QUEENSLAND**

**Queensland 4072
Australia**

TECHNICAL REPORT

No. 99-29

**Real-time Specification and Reasoning
Using Maximal Intervals**

**C.J. Fidge I.J. Hayes
B. P. Mahony A. K. Wabenhorst**

September 1999

Phone: +61 7 3365 1003

Fax: +61 7 3365 1533

<http://svrc.it.uq.edu.au>

Published in *6th Annual Australasian Conference on Parallel and Real-Time Systems (PART'99)*, Springer-Verlag, 1999.
© Copyright 1999 Springer-Verlag.

Note: Most SVRC technical reports are available via anonymous ftp, from `svrc.it.uq.edu.au` in the directory `/pub/techreports`. Abstracts and compressed postscript files are available via `http://svrc.it.uq.edu.au`

Real-time Specification and Reasoning Using Maximal Intervals

C.J. Fidge I.J. Hayes B. P. Mahony* A. K. Wabenhorst

Abstract.

Interval-based formalisms for real-time systems describe behaviour via the time intervals during which predicates on the system state hold. However, these formalisms are clumsy for relating the occurrences of state changes, or events. Here we overcome this by introducing definitions and laws for maximal intervals. These define the longest time intervals over which a predicate holds, and their endpoints thus mark significant state changes.

1 Introduction

A number of specification and reasoning formalisms for real-time systems use the time intervals during which state predicates hold as their fundamental modelling concept. These formalisms are ideally suited to a state-oriented style of specification in which system properties are defined over all time, or over certain intervals of time. However, they are awkward for expressing event-oriented behaviour in which state changes marking occurrences of significant events must be related to one another.

Here we overcome this by introducing definitions for *maximal* intervals. Rather than defining all intervals during which a predicate holds, maximal operators define the *longest* such intervals. The endpoints of each maximal interval thus mark times at which significant state changes occur, so that other events can be related to these points.

2 Motivation

Time intervals are the basic modelling concept in a number of real-time specification and reasoning formalisms. The Duration Calculus [14, 3] and its many derivatives, such as Temporal Algebra [12], are founded in interval temporal logic [8] and integral calculus. The Timed Interval Calculus [6, 2] is founded in

*Information Technology Division, Defence Science and Technology Organisation, Salisbury, South Australia 5108, Australia.

set theory, and is thus compatible with the Z specification language [11]. However, despite their different bases, all of these interval calculi offer the programmer similar operators and capabilities. Below we use Timed Interval Calculus notation.

Recall that an interval from point a , exclusive, to point b , inclusive, is traditionally written as $(a, b]$. Similarly for other combinations of endpoint inclusion or exclusion. The Timed Interval Calculus extends this notion to allow sets of time intervals to be defined using special brackets surrounding predicates on the system state. For a predicate P , expression $\{P\}$ denotes the set of all left-open, right-closed time intervals during which P holds at every point in the interval. Combinations of these intervals are possible. For instance, $\{P\}$ defines the set of left-open intervals in which P is true—both right-open and right-closed intervals may be included in the set. The case $\llbracket P \rrbracket$ defines all intervals in which P holds, regardless of endpoint closure. Within predicate P , three special values can appear: α denotes the starting time of the interval, ω denotes its finishing time, and δ denotes the interval’s duration.

For example, consider the well-known ‘gas burner’ problem [14]. Let there be two variables representing the state of a gas-fired furnace: real-valued variable gas is the current rate of gas flow in litres per second, and boolean variable $flame$ is true only if the gas is currently lit. The following predicate then expresses the property that there can be no flame without gas flow.

$$\llbracket flame \rrbracket \subseteq \llbracket gas > 0 \rrbracket \tag{1}$$

On the left are all intervals in which the flame is burning, and on the right are all intervals in which gas is flowing. Thus, any interval in which there is a flame is also one in which gas is available.

The converse is not necessarily true, however. Let states characterised by the following predicate represent a potentially hazardous gas leak due to there being a flow of unburnt gas.

$$Leak == gas > 0 \wedge \neg flame$$

Then we can express the property that gas must not leak continually for more than 4 seconds as follows.

$$\llbracket Leak \rrbracket \subseteq \llbracket \delta \leq 4 \rrbracket \tag{2}$$

On the left is the set of all intervals over which predicate $Leak$ holds. On the right is the set of all intervals whose duration δ does not exceed 4 seconds. Property (2) thus states that every interval in which gas is leaking has a duration no greater than 4 seconds.

To describe sequences of states, interval calculi must also provide a *concatenation* operator to allow subintervals to be joined end-to-end [8]. Given two sets of intervals S_1 and S_2 , then their concatenation ‘ $S_1 ; S_2$ ’ is the set of all intervals composed of an interval from S_1 concatenated with an interval from S_2 . The right-hand endpoint of the interval from S_1 must equal the left-hand endpoint

of the interval from S_2 and the intervals may not overlap [2]. (See Section 3.2.) Using this operator it is possible to express properties concerning state changes, or *events*. For example, a requirement that the gas must be switched off no more than three seconds after the flame goes out can be expressed as follows.

$$\begin{aligned} & \llbracket \text{flame} \rrbracket ; \llbracket \neg \text{flame} \wedge \text{gas} > 0 \rrbracket ; \llbracket \text{gas} = 0 \rrbracket \\ \subseteq & \llbracket \text{flame} \rrbracket ; \llbracket \delta \leq 3 \rrbracket ; \llbracket \text{gas} = 0 \rrbracket \end{aligned} \quad (3)$$

On the left we have intervals consisting of an initial subinterval in which the flame is lit, followed by a subinterval in which the flame is out but the gas is still flowing, followed by a final subinterval in which the gas is out. The concatenation points thus mark the events of interest, i.e., the time at which the flame goes out, and the time at which the gas is switched off. On the right, we have intervals consisting of a subinterval where the flame is lit, followed by a subinterval no longer than three seconds (during which the gas may still be flowing), followed by a subinterval where there is no gas flow. Therefore, the gas is off no more than 3 seconds after the flame was burning. Note that this property assumes the occurrence of the two events, i.e., the flame going out and the gas being switched off, and constrains only the duration between them. The property does not consider other sequences of events, such as the gas being switched on and the flame being subsequently lit, which may have a different timing requirement. Nor does it consider the case where the gas stays on indefinitely.

Properties concerning events are awkward to express because capturing a change to some predicate P means it must appear on both sides of a ‘;’ operator in a $\llbracket P \rrbracket ; \llbracket \neg P \rrbracket$ pattern. This becomes even worse when attempting to expressing minimum durations. For example, consider the requirement that once a gas leak has been stopped then no further leak may occur for at least 30 seconds. Using a concatenation operator, this must be written as follows [13].

$$\llbracket \text{Leak} \rrbracket ; \llbracket \neg \text{Leak} \rrbracket ; \llbracket \text{Leak} \rrbracket \subseteq \llbracket \delta \geq 30 \rrbracket \quad (4)$$

This achieves the desired effect because the *Leak* intervals at each end can be arbitrarily short. To make property (4) true, therefore, the predicate $\neg \text{Leak}$ in the middle must hold for at least 30 seconds. Stating this requirement, however, forced us to repeat predicate ‘*Leak*’ three times, in order to capture the two state changes of interest.

Such properties would be easier to express if the interval calculi had brackets for describing not all intervals during which a predicate holds, but rather the longest possible intervals. The endpoints of these intervals would then occur at the times where the state changed. To do this, Mahony and Hayes defined ‘cover’ intervals as the set of maximal disjoint intervals that encompass all times at which some predicate of interest was true [5, 4, 7]. This notion was further refined through a series of case studies [7, 6]. Here we update and simplify these definitions, and define laws for reasoning about specifications that use maximal intervals.

3 Maximal Intervals

In this section we present definitions of, and laws for, maximal intervals in a Z -based notation.

3.1 Intervals and Sets of Intervals

In previous work, we defined the time domain to be the set of real numbers, and then defined an interval calculus in terms of finite intervals [2]. However, defining maximal intervals introduces the need to consider properties that remain true indefinitely, and thus forces us to introduce infinite intervals to the calculus.

Let \mathbb{R} be the set of real numbers. In order to provide explicit values for left and right endpoints and durations of infinite intervals we introduce the extreme values $-\infty$ and ∞ . Let \mathbb{R}_∞ be the the real numbers plus two special values, $-\infty$ and ∞ [10, §2.3]. An interval is defined as a set of real numbers (not containing $\pm\infty$) in the usual way.

Definition 1 (Intervals)

$$\begin{aligned} (a, b) &== \{x : \mathbb{R} \mid a < x < b\}, & \text{where } a, b : \mathbb{R}_\infty \text{ and } a < b \\ (a, b] &== \{x : \mathbb{R} \mid a < x \leq b\}, & \text{where } a : \mathbb{R}_\infty, b : \mathbb{R} \text{ and } a < b \\ [a, b) &== \{x : \mathbb{R} \mid a \leq x < b\}, & \text{where } a : \mathbb{R}, b : \mathbb{R}_\infty \text{ and } a < b \\ [a, b] &== \{x : \mathbb{R} \mid a \leq x \leq b\}, & \text{where } a, b : \mathbb{R} \text{ and } a \leq b \end{aligned}$$

Sets of these intervals are then defined straightforwardly. Below, \mathcal{OO} will denote the set of left-open, right-closed intervals, \mathcal{CO} the set of left-closed, right-open intervals, and so on.

Definition 2 (Sets of Intervals)

$$\begin{aligned} \mathcal{OO} &== \{a, b : \mathbb{R}_\infty \mid a < b \bullet (a, b)\} \\ \mathcal{OC} &== \{a : \mathbb{R}_\infty, b : \mathbb{R} \mid a < b \bullet (a, b]\} \cup \{a : \mathbb{R} \cup \{-\infty\} \bullet (a, \infty)\} \\ \mathcal{CO} &== \{a : \mathbb{R}, b : \mathbb{R}_\infty \mid a < b \bullet [a, b)\} \cup \{b : \mathbb{R} \cup \{\infty\} \bullet (-\infty, b)\} \\ \mathcal{CC} &== \{a, b : \mathbb{R} \mid a \leq b \bullet [a, b]\} \cup \{a : \mathbb{R} \bullet [a, \infty)\} \cup \\ &\quad \{b : \mathbb{R} \bullet (-\infty, b]\} \cup \{(-\infty, \infty)\} \end{aligned}$$

Thus, intervals may extend to $\pm\infty$. In particular, intervals $(-\infty, b]$, $[a, \infty)$ and $(-\infty, \infty)$ have been defined to be in set \mathcal{CC} because they are closed in the standard topology of the reals [10].

3.2 Definitions for Maximal Intervals

When modelling real-time systems, which may interact with continuous hardware devices, or asynchronous software processes, we let the time domain \mathbb{T} be the real numbers.

Definition 3 (Time)

$$\mathbb{T} == \mathbb{R}$$

In interval calculi, system variables are modelled as functions from the time domain to their value at that time. For example, the two variables in the gas burner example are declared here as total functions. Let \mathbb{R}^+ be the set of non-negative real numbers, and \mathbb{B} be the boolean type.

$$\begin{aligned} gas &: \mathbb{T} \rightarrow \mathbb{R}^+ \\ flame &: \mathbb{T} \rightarrow \mathbb{B} \end{aligned}$$

For brevity, we allow such variables to appear in predicates as if they are simple variables of their range type, provided that the entire predicate is appropriately indexed with a time value. For instance, given a time t , let $Leak(t)$ denote the predicate ' $gas(t) > 0 \wedge \neg flame(t)$ '.

Sets of intervals during which a predicate P holds are then defined as the set of all intervals Φ such that predicate P is true at every time τ in Φ .

Definition 4 (Interval Predicates) *Let P be a predicate on the system state.*

$$\begin{aligned} \langle P \rangle &== \{ \Phi : \mathcal{OO} \mid (\forall \tau : \Phi \bullet P(\tau)) \} \\ \langle P \rangle &== \{ \Phi : \mathcal{OC} \mid (\forall \tau : \Phi \bullet P(\tau)) \} \\ [P] &== \{ \Phi : \mathcal{CO} \mid (\forall \tau : \Phi \bullet P(\tau)) \} \\ [P] &== \{ \Phi : \mathcal{CC} \mid (\forall \tau : \Phi \bullet P(\tau)) \} \end{aligned}$$

Commonly-required syntactic abbreviations are introduced so that predicate P may refer to properties of the intervals it is defining, specifically the starting time α , finishing time ω , and duration δ . For some interval I , let $\inf I$ be its infimum, and $\sup I$ be its supremum.

$$\begin{aligned} \alpha &== \inf \Phi \\ \omega &== \sup \Phi \\ \delta &== \omega - \alpha \end{aligned}$$

Recall that the infimum of a left-infinite interval $(-\infty, b)$ is $-\infty$, and the supremum of a right-infinite interval (a, ∞) is ∞ , even though no interval may *contain* these values [10, §2.3].

As in previous work [2], syntactic shorthands for combinations of these brackets are defined trivially as unions. For instance, the set of all right-open intervals is defined as follows.

$$\llbracket P \rangle == [P] \cup \langle P \rangle$$

Thus, the left-hand endpoint may be open or closed. Similarly for sets of right-closed $\llbracket P]$, left-open $\langle P]$, and left-closed $[P]$ intervals. The final case is where no constraints are placed on the endpoints.

$$\llbracket P] == \langle P \rangle \cup [P] \cup \langle P] \cup [P]$$

We now extend this notation to define maximal intervals. Following previous convention [6], we indicate a maximal endpoint by adding an extra vertical line to the brackets defined above. For instance, the left-maximal open intervals in which predicate P holds are denoted $\llbracket P \rrbracket$. Intuitively, intervals in set $\llbracket P \rrbracket$ are those intervals from set $\langle P \rangle$ which cannot be extended any further to the left because they will overlap with times at which P is false, or they already extend to $-\infty$. Let \subset denote proper subset (with \subseteq denoting subset).

Definition 5 (Maximal Intervals) *Left-maximal intervals are defined as follows.*

$$\begin{aligned} \langle P \rangle &== \{I : \langle P \rangle \mid (\neg \exists J : \langle P \rangle \bullet I \subset J \wedge \sup I = \sup J)\} \\ \llbracket P \rrbracket &== \{I : \llbracket P \rrbracket \mid (\neg \exists J : \llbracket P \rrbracket \bullet I \subset J \wedge \sup I = \sup J)\} \\ \lllbracket P \lllbracket &== \{I : \lllbracket P \lllbracket \mid (\neg \exists J : \lllbracket P \lllbracket \bullet I \subset J \wedge \sup I = \sup J)\} \end{aligned}$$

Similarly for right-closed and right-unspecified intervals. Also, similarly for right-maximal intervals, except that $\inf I = \inf J$ replaces $\sup I = \sup J$. Combinations for maximal right-open intervals are defined as follows.

$$\begin{aligned} \langle P \rangle &== \{I : \langle P \rangle \mid (\neg \exists J : \langle P \rangle \bullet I \subset J)\} \\ \llbracket P \rrbracket &== \{I : \llbracket P \rrbracket \mid (\neg \exists J : \llbracket P \rrbracket \bullet I \subset J)\} \\ \lllbracket P \lllbracket &== \{I : \lllbracket P \lllbracket \mid (\neg \exists J : \lllbracket P \lllbracket \bullet I \subset J)\} \end{aligned}$$

Similarly for other right-maximal intervals.

Thus, for instance, the definition of $\llbracket P \rrbracket$ states that for each left-maximal interval I there is no interval J in which P holds that can be extended further leftwards than I . The condition $\sup I = \sup J$ ensures that the right-hand endpoint of J is not extended.

In previous work using finite intervals [2], concatenation was defined in Z as follows. The definition proves to be equally applicable to infinite intervals. Let \mathbb{I} be the set of all possible intervals, i.e., $\mathbb{I} = \mathcal{OO} \cup \mathcal{CC} \cup \mathcal{CO} \cup \mathcal{OC}$.

Definition 6 (Concatenation) *Let X and Y be sets of intervals.*

$$X ; Y == \{x : X ; y : Y ; z : \mathbb{I} \mid z = x \cup y \wedge (\forall t_1 : x ; t_2 : y \bullet t_1 < t_2) \bullet z\}$$

In other words, an interval x from set X can be joined to an interval y from set Y , to form a new interval z in set ' $X ; Y$ ', provided that (a) x and y meet so that z can be formed from their union, and (b) all points t_1 in x occur before all points t_2 in y so that the two subintervals do not overlap.

With these definitions in place, we can now revisit the motivational example from Section 2. Property (3), which required the gas to be switched off no more than 3 seconds after the flame went out, can now be expressed more concisely as follows.

$$\lllbracket \neg \text{flame} \rrlbracket \cap \lllbracket \text{gas} > 0 \rrlbracket \subseteq \lllbracket \delta \leq 3 \rrlbracket \quad (5)$$

The left-hand endpoints of the ‘ \neg *flame*’ intervals on the left are now guaranteed to mark times at which the flame goes out. There is thus no need to introduce a preceding ‘*flame*’ interval. Furthermore, the right-hand endpoints of the ‘*gas* $>$ 0’ intervals on the left will all occur at points where the gas was switched off, so there is no need to state ‘*gas* = 0’ in the following interval.

Note that property (3) did not require the gas to be switched off following the flame going out; if the gas stays on indefinitely there is no ‘*gas* = 0’ subinterval and the property says nothing. However, property (5) prevents this behaviour because the ‘*gas* $>$ 0’ interval on the left-hand side could extend to ∞ , whereas there is a ‘ $\delta \leq 3$ ’ constraint on the right-hand side.

Similarly, property (4), which states that periods in which gas is not leaking must have a minimum duration of 30 seconds, can be considerably simplified.

$$\llbracket \neg Leak \rrbracket \subseteq \llbracket \delta \geq 30 \rrbracket \quad (6)$$

Again, the endpoints of the intervals on the left will all occur at times where the state of the *Leak* predicate changes, or $\pm\infty$ if the ‘ $\neg Leak$ ’ property remains true indefinitely.

3.3 Discussion

Introducing infinite intervals to the calculus raises a number of subtle issues. For instance, from property (6) it follows that if there is any time $t : \mathbb{R}$ after which predicate $\neg Leak$ is forever true, then no matter how large t is, the interval $[t, \infty)$ satisfies the property because the interval’s duration is $\infty - t = \infty$ [10, §2.3].

Care must be taken with maximal intervals when choosing whether an endpoint should be open or closed. For example, let x be the identity function ($\lambda t : \mathbb{T} \bullet t$). Then, $\langle 4 \leq x \rangle = \langle 4 < x \rangle = \{(4, \infty)\}$ and $\llbracket 4 \leq x \rrbracket = \llbracket 4 < x \rrbracket = \{\{4, \infty)\}$. However, $\llbracket 4 < x \rrbracket = \emptyset$. In the latter case, no matter what closed interval with infimum greater than 4 is chosen, it can always be extended slightly to the left while keeping predicate $4 < x$ true, so no maximal closed interval that has this property can be found. In practice, this problem can be solved by using ‘unspecified’ endpoint brackets. The set $\llbracket 4 < x \rrbracket = \{(4, \infty)\}$ contains all valid matches of endpoint closure with the predicate.

Other unusual results are possible by mixing maximal brackets with constraints on the interval duration. For instance, $\llbracket \delta < 1 \rrbracket = \emptyset$. In this case, no matter what length interval less than 1 is chosen, it can always be extended slightly to either the left or right while keeping predicate $\delta < 1$ true, so no maximal interval that has this property can be found.

It is also interesting to note that

$$\langle P \wedge \alpha \geq 10 \rangle \neq \langle P \rangle \cap \langle \alpha \geq 10 \rangle.$$

For instance, if P is true from times 5 to 15, exclusive, then $\langle P \wedge \alpha \geq 10 \rangle = \{(10, 15)\}$, but $\langle P \rangle \cap \langle \alpha \geq 10 \rangle = \{(5, 15)\} \cap \langle \alpha \geq 10 \rangle = \emptyset$.

However, maximal brackets satisfy certain intuitively reasonable conditions. Let P be a predicate, with Φ (and therefore α , ω and δ) not free in P . Then the following conditions are satisfied:

- Each interval is contained in a maximal interval, i.e., if $I \in \llbracket P \rrbracket$, then there exists $I' \in \llbracket P \rrbracket$ such that $I \subseteq I'$.
- Maximal intervals are disjoint, i.e., if $I, I' \in \llbracket P \rrbracket$ such that $I \cap I' \neq \emptyset$, then $I = I'$.
- $\llbracket P \rrbracket = \llbracket P \rrbracket \cap \llbracket P \rrbracket$.

If Φ is free in P , then counterexamples can be found.

3.4 Laws for Maximal Intervals

In previous work, a number of laws were presented for the Timed Interval Calculus [2, 1], themselves derived from laws defined for the Duration Calculus [9, 3]. These laws were stated for non-maximal intervals only. Although some of these laws can be reused for maximal intervals, many cannot. For instance, given two predicates P and Q , the equivalence $\llbracket P \rrbracket \cap \llbracket Q \rrbracket = \llbracket P \wedge Q \rrbracket$ expresses an elegant relationship between logical conjunction and set intersection. However, in general, we have only the weaker Law 4 below ($\llbracket P \rrbracket \cap \llbracket Q \rrbracket \subseteq \llbracket P \wedge Q \rrbracket$) for maximal intervals. Similarly, the relationship between logical implication and subset for non-maximal intervals is that if $P \Rightarrow Q$, then $\llbracket P \rrbracket \subseteq \llbracket Q \rrbracket$. However, for maximal intervals, $\llbracket P \rrbracket \not\subseteq \llbracket Q \rrbracket$ in general, because Q may hold over a longer period of time than P , and hence set $\llbracket Q \rrbracket$ does not contain all of the maximal subintervals in which P holds. Thus maximal intervals do not enjoy many of the set-theoretic properties of their non-maximal predecessors.

Nevertheless, we may still state many useful laws applicable to sets of maximal intervals. The most important new law, Law 1, states that sets of maximal intervals are always subsets of non-maximal intervals. This allows some existing laws to be reused for maximal intervals. Let P and Q be predicates. Let S , T and U be sets of time intervals.

- Law 1** $\llbracket P \rrbracket \subseteq \llbracket P \rrbracket$
- Law 2** If $P \Leftrightarrow Q$, then $\llbracket P \rrbracket = \llbracket Q \rrbracket$.
- Law 3** $\llbracket true \rrbracket = \{(-\infty, \infty)\}$ and $\llbracket false \rrbracket = \emptyset$
- Law 4** $\llbracket P \rrbracket \cap \llbracket Q \rrbracket \subseteq \llbracket P \wedge Q \rrbracket$
- Law 5** If $S \subseteq S'$ and $T \subseteq T'$ then $S ; T \subseteq S' ; T'$.
- Law 6** $(S ; T) ; U = S ; (T ; U)$
- Law 7** $S ; \emptyset = \emptyset ; S = \emptyset$
- Law 8** $(S \cup T) ; U = (S ; U) \cup (T ; U)$
 $S ; (T \cup U) = (S ; T) \cup (S ; U)$
- Law 9** $(S \cap T) ; U \subseteq (S ; U) \cap (T ; U)$
 $S ; (T \cap U) \subseteq (S ; T) \cap (S ; U)$
- Law 10** If Φ is not free in P , then $\llbracket P \wedge \delta > 0 \rrbracket = \llbracket P \rrbracket ; \llbracket P \rrbracket$.
- Law 11** If Φ is not free in P , and $r : \mathbb{R}^+$, then $\llbracket P \wedge \delta = r \rrbracket = \llbracket P \wedge \delta = r \rrbracket$ and $(\llbracket P \wedge \delta = r \rrbracket) = (P \wedge \delta = r)$.
- Law 12** $\llbracket P \rrbracket ; \llbracket P \rrbracket = \llbracket P \rrbracket ; \llbracket P \rrbracket = \emptyset$
- Law 13** $\llbracket \neg P \rrbracket ; \llbracket P \rrbracket = \llbracket \neg P \rrbracket ; \llbracket P \rrbracket = \llbracket \neg P \rrbracket ; \llbracket P \rrbracket$
- Law 14** $\llbracket \neg P \rrbracket ; (\llbracket P \rrbracket \cap S) = \llbracket \neg P \rrbracket ; (\llbracket P \rrbracket \cap S) = \llbracket \neg P \rrbracket ; (\llbracket P \rrbracket \cap S)$

Laws 5 to 9 [2] hold for any interval, while Laws 2 to 4 and 10 are modifications of existing laws. Laws 11 to 14 are entirely new.

Law 11 reminds us that the use of the special values α , ω , δ or Φ within maximal brackets may yield counter-intuitive results. Since it constrains the length δ of the intervals, then the requirement that one of the endpoints is maximal has no effect. The starting time α of intervals on the left may occur *after* the point at which the predicate P became true, because such an interval cannot be extended further leftwards without making predicate $\delta = r$ false.

Law 12 shows that if a bracket is maximal with respect to a predicate, then the same predicate cannot hold beyond the bracket. Law 13 shows that at a point of concatenation where a predicate changes from false to true, the intervals are inevitably maximal. Law 14 is a generalisation of Law 13. These laws confirm that maximal brackets can express any requirement described in the $\llbracket P \rrbracket; \llbracket \neg P \rrbracket$ style. However, the converse is not necessarily the case, because maximal brackets can express properties of infinite behaviours that cannot be stated using the concatenation operator.

Indeed, the relationship between maximal and ordinary brackets can be demonstrated by proving that property (6), which is expressed using maximal brackets, implies property (4), which is expressed in the concatenation style. That is, we wish to prove the following relationship

$$\begin{aligned} \llbracket \neg Leak \rrbracket &\subseteq \llbracket \delta \geq 30 \rrbracket \\ \Rightarrow \llbracket Leak \rrbracket; \llbracket \neg Leak \rrbracket; \llbracket Leak \rrbracket &\subseteq \llbracket \delta \geq 30 \rrbracket. \end{aligned}$$

The proof is straightforward:

$$\begin{aligned} &\llbracket Leak \rrbracket; \llbracket \neg Leak \rrbracket; \llbracket Leak \rrbracket \\ &= \text{'Law 13'} \\ &\llbracket Leak \rrbracket; \llbracket \neg Leak \rrbracket; \llbracket Leak \rrbracket \\ &\subseteq \text{'property (6) and Law 5'} \\ &\llbracket Leak \rrbracket; \llbracket \delta \geq 30 \rrbracket; \llbracket Leak \rrbracket \\ &\subseteq \text{'definition of ;'} \\ &\llbracket \delta \geq 30 \rrbracket \end{aligned}$$

Note that property (6) expresses constraints on infinite intervals that are not even defined by property (4).

Similarly, we can prove that property (5) implies property (3). That is, we prove

$$\begin{aligned} \llbracket \neg flame \rrbracket \cap \llbracket gas > 0 \rrbracket &\subseteq \llbracket \delta \leq 3 \rrbracket \\ \Rightarrow \llbracket flame \rrbracket; \llbracket \neg flame \wedge gas > 0 \rrbracket; \llbracket gas = 0 \rrbracket &\subseteq \llbracket flame \rrbracket; \llbracket \delta \leq 3 \rrbracket; \llbracket gas = 0 \rrbracket. \end{aligned}$$

The proof is equally straightforward:

$$\llbracket flame \rrbracket; \llbracket \neg flame \wedge gas > 0 \rrbracket; \llbracket gas = 0 \rrbracket$$

$$\begin{aligned}
&= \text{'conjunctivity, [2, Law 3]'} \\
&\quad \llbracket \textit{flame} \rrbracket ; (\llbracket \neg \textit{flame} \rrbracket \cap \llbracket \textit{gas} > 0 \rrbracket) ; \llbracket \textit{gas} = 0 \rrbracket \\
&= \text{'Law 14 twice'} \\
&\quad \llbracket \textit{flame} \rrbracket ; (\llbracket \neg \textit{flame} \rrbracket \cap \llbracket \textit{gas} > 0 \rrbracket) ; \llbracket \textit{gas} = 0 \rrbracket \\
&\subseteq \text{'property (3) and Law 5'} \\
&\quad \llbracket \textit{flame} \rrbracket ; \llbracket \delta \leq 3 \rrbracket ; \llbracket \textit{gas} = 0 \rrbracket
\end{aligned}$$

Note that unlike property (5), property (3) permits the gas to remain on indefinitely after the flame has gone out.

4 Conclusion

We have presented a new definition of maximal intervals for the Timed Interval Calculus, in order to improve its expressive power for event-oriented specification and reasoning. In doing so it proved necessary to introduce infinite intervals to the formalism, to cater for properties that remain unchanged forever.

Acknowledgements We wish to thank Keith Duddy, Luke Everett and Colin Millerchip for their earlier definitions of maximal intervals, Stephen Grundon for commenting on this work, and the anonymous referees for suggesting improvements. This research was funded by the Information Technology Division of the Defence Science and Technology Organisation.

References

1. C. J. Fidge. Modelling discrete behaviour in a continuous-time formalism. In K. Araki, A. Galloway, and K. Taguchi, editors, *IFM'99: Proceedings of the First International Conference on Integrated Formal Methods*, pages 170–188. Springer-Verlag, June 1999.
2. C. J. Fidge, I. J. Hayes, A. P. Martin, and A. K. Wabenhurst. A set-theoretic model for real-time specification and reasoning. In J. Jeuring, editor, *Mathematics of Program Construction (MPC'98)*, volume 1422 of *Lecture Notes in Computer Science*, pages 188–206. Springer-Verlag, 1998.
3. M. R. Hansen and Zhou Chaochen. Duration calculus: Logical foundations. *Formal Aspects of Computing*, 9(3):283–330, 1997.
4. B. Mahony and I. J. Hayes. A case study in timed refinement: A central heater. In J. M. Morris and R. C. Shaw, editors, *Fourth Refinement Workshop*, pages 138–149. Springer-Verlag, 1991.
5. B. Mahony and I. J. Hayes. Using continuous real functions to model timed histories. In *Proc. Sixth Australian Software Engineering Conference (ASWEC'91)*, pages 257–270, Sydney, July 1991.
6. B. Mahony, C. Millerchip, and I. J. Hayes. A boiler control system: Overview of a case-study in timed refinement. In D. Del Bel Belluz and H. C. Ratz, editors, *Software Safety: Everybody's Business—Proc. 1993 International Workshop on Design and Review of Software Controlled Safety-Related Systems*, pages 189–208, 1994.

7. B. P. Mahony and I. J. Hayes. A case-study in timed refinement: A mine pump. *IEEE Transactions on Software Engineering*, 18(9):817–826, September 1992.
8. B. Moszkowski. *Executing Temporal Logic Programs*. Cambridge University Press, 1986.
9. A. P. Ravn. *Design of Embedded Real-Time Computing Systems*. PhD thesis, Department of Computer Science, Technical University of Denmark, 1995.
10. H. L. Royden. *Real Analysis*. Collier Macmillan, 3rd edition, 1988.
11. J. M. Spivey. *The Z Notation: A Reference Manual*. Prentice Hall International, 1989.
12. B. von Karger. Temporal algebra. Institut für Informatik und Praktische Mathematik, Christian-Albrechts-Universität Kiel, September 1997. Habilitation thesis.
13. Zhou Chaochen. Duration calculi: An overview. In D. Björner, M. Broy, and I. Pottosin, editors, *Formal Methods in Programming and Their Applications*, volume 735 of *Lecture Notes in Computer Science*, pages 256–266. Springer-Verlag, 1993. Extended abstract.
14. Zhou Chaochen, C. A. R. Hoare, and A. P. Ravn. A calculus of durations. *Information Processing Letters*, 40:269–276, December 1991.