Reasoning about Discontinuities in the Event Calculus

Rob Miller
Department of Computing,
Imperial College of Science, Technology & Medicine,
180 Queen's Gate, London SW7 2BZ, U.K.
rsm@doc.ic.ac.uk
http://www.doc.ic.ac.uk/~rsm/

Murray Shanahan
Department of Computer Science,
Queen Mary & Westfield College,
Mile End Road, London E1 4NS, U.K.
mps@dcs.qmw.ac.uk
http://www.dcs.qmw.ac.uk/~mps/

This paper appears in the proceedings of KR'96, but without Appendix B.

Abstract

This paper describes a logic-based formalism which combines techniques for reasoning about actions with standard mathematical techniques for modelling dynamic systems using the differential calculus. The formalism inherits a robust solution to the frame problem which can handle concurrency, non-determinism, domain constraints and narrative. It also incorporates a mechanism for reasoning about the boundary conditions associated with systems of differential equations defined over various intervals. This mechanism overcomes a number of drawbacks of previous systems.

1 INTRODUCTION

Solutions to the frame problem now exist which can handle a variety of phenomena, such as narrative, concurrent action and non-deterministic action (see (Shanahan 1997) for a general discussion). However, the topic of continuous change has received relatively little attention in the Reasoning about Action literature. In particular, a satisfactory general framework has yet to be developed which reconciles logic-based techniques for reasoning about action with the standard mathematical approach to modelling dynamic systems, using differential calculus.

Some previous logic-based approaches to reasoning about action do allow limited types of mathematical expressions involving continuously varying parameters to be embedded within domain descriptions. For example, Shanahan (1990) presents a logic programming approach to representing continuous change, based on the event calculus of Kowalski and Sergot (1986). Further work along these lines has been done by Van Belleghem *et al.* (1994), whose treatment allows a wider class of mathematical expressions, by Shanahan (1995), where a full predicate calculus version using circumscription is presented, and by Herrmann and Thielscher (1996), whose notion of a *process* generalises Shanahan's notion of a *trajectory*. However, none of these frameworks incorporates the notion of a derivative function. Hence any differential equations representing the domain have to be solved before they can be added to the model.

In contrast, Sandewall (1989a, 1989b) presents an approach to continuous change which combines logic and differential equations. Sandewall identifies the need to incorporate mechanisms within such frameworks to deal with the boundary conditions usually associated with sets of such equations. He advocates a default reasoning method, based on Shoham's notion of chronological minimisation (Shoham 1988), to generate new boundary conditions when an action or event transforms one mathematical model into another. However, as shown by Rayner (1991), Sandewall's approach leaves open the possibility of anomalous models. Miller (1996) introduces techniques for incorporating differential equations within a circumscribed situation calculus. He avoids Rayner's anomalous models through the use of a (minimised) *Breaks* predicate, which describes when actions cause discontinuities in particular parameters.

This paper draws on the techniques of (Miller 1996) in order to construct an event calculus resembling that of (Shanahan 1997, Ch. 16), but which allows continuous change to be described using arbitrary systems of differential equations. Unlike existing versions of the situation calculus which can handle continuous change, such as (Miller 1996) and (Reiter 1996), this calculus inherits the ability to handle domain constraints, concurrent actions, and actions with non-deterministic effects from the event calculus (Shanahan 1997, Ch. 16).

2 TWO EXAMPLES

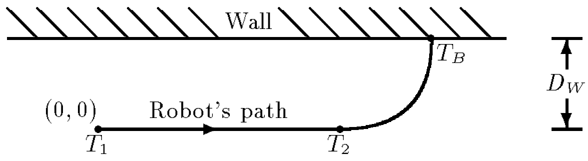
Both the examples below help to illustrate how arbitrary systems of simultaneous differential equations can be incorporated within an axiomatic description of a given domain. The first, adapted from (Shanahan 1996), concerns the movement of a mobile robot, which will stop whenever it bumps against an object (in this case, a wall). It thus shows how the formalism represents actions (events) which are "triggered" when certain conditions occur. The second is a modification of the "water tanks" example from (Miller 1996). It shows how the formalism supports reasoning about concurrently performed actions (a tap is opened, and some water is simultaneously scooped from a tank). A key feature of both these examples is that, as is usual in the mathematical modelling of dynamic systems, the continuously changing aspects of the domain are represented as a system of (possibly *simultaneous* or *coupled*) differential equations, together with one or more sets of *boundary* conditions (or *initial* conditions). In such models, the complete set of boundary conditions may not be listed – some conditions are implicit in the accompanying physical description of the domain, or are "common sense".

In this section we concentrate on showing how both examples can be modelled "conventionally" using differential calculus, highlighting some of the reasoning processes involved in obtaining explicit mathematical expressions for parameters' behaviours over time. In both cases we go into some mathematical detail, primarily in order to analyse why and how specific values are assigned to arbitrary constants of integration during the mathematical modelling process. This analysis is central to the understanding of the rest of the paper. As we shall see, these value assignments are based partly on "common sense", and it was incomplete modelling of this implicit assignment process which lead to problems with anomalous models in the work of Sandewall (1989a, 1989b).

In both examples, dashes ($'$) are used to refer to derivatives of parameters – for example, P' refers to the first derivative with respect to time of P.

2.1 THE MOBILE ROBOT EXAMPLE

The robot in this example can start (and stop) moving at any constant speed in the direction it is facing (its "bearing"). It can also (continuously) vary its bearing by turning at any constant angular velocity, possibly while moving. It can therefore follow any path consisting of a series of circle fragments and straight lines. It has sensors which will cause it to stop if it bumps against any object. We will represent the following scenario. At time 0 the robot is stationary at the point $(0, 0)$, facing east, and has angular velocity 0. At time T_1 it starts moving forward (east) at speed S^+. At time T_2 it also starts to turn in a anticlockwise direction with angular speed A^+. Some time later it bumps into the only other object in the domain – a wall (of "infinite length") running east-west, at a distance D_W due north from the origin.



By idealising the robot as a moving point and the wall as a fixed line, a mathematical model of this domain can be formulated in a simple way, using the parameters (real-valued functions of time) *Bearing*, *NSP* ("North-South-Position"), and *EWP* ("East-West-Position"). We can write down the following mathematical constraints between these parameters using the "constants" S (forward speed) and A (angular speed), whose values are determined directly and instantaneously by the robot's actions, or by the event of bumping into the wall.

$$EWP'(t) = S.\cos(Bearing(t)) \quad (1)$$
$$NSP'(t) = S.\sin(Bearing(t)) \quad (2)$$
$$Bearing'(t) = A \quad (3)$$

Let T_B be the time that the robot bumps against the wall. We know that $EWP(0) = NSP(0) = Bearing(0) = 0$, that $NSP(T_B) = D_W$, and that, because of the robot's actions and the effect of bumping into the wall,

$$\text{for } 0 \leq t \leq T_1: \quad A = 0 \text{ and } S = 0 \quad (4)$$
$$\text{for } T_1 < t \leq T_2: \quad A = 0 \text{ and } S = S^+ \quad (5)$$
$$\text{for } T_2 < t \leq T_B: \quad A = A^+ \text{ and } S = S^+ \quad (6)$$
$$\text{for } T_B < t: \quad A = 0 \text{ and } S = 0 \quad (7)$$

The formalism described in this paper allows us to include equations (1)–(3) directly in the domain-dependent axiomatisation, rather than first having to solve them to determine an explicit or stratified set of expressions for *EWP*, *NSP* and *Bearing*, as in (Shanahan 1990) and related approaches. It also allows us to infer (4)–(7) from more general knowledge, both about the effects and timings of the robot's actions, and about the general hypothetical circumstances under which the robot will bump into the wall.

It is important to note, however, that from a strictly mathematical point of view, expressions (1)–(7) do not contain enough information to uniquely determine the position of the robot at a given time T (even when considered together with information about the values of each parameter at time 0). An assumption also has to be made that EWP, NSP and $Bearing$ are continuous at times T_1, T_2 and T_B. The simultaneous solution to (1)–(3) for any given value of S and A (obtained by integrating (3), substituting the answer in (1) and (2), and integrating again) is

$$\begin{aligned} EWP(t) &= \tfrac{S}{A}\sin(At + C_1) + C_2 \\ NSP(t) &= \tfrac{-S}{A}\cos(At + C_1) + C_3 \\ Bearing(t) &= At + C_1 \end{aligned}$$

where C_1, C_2 and C_3 are arbitrary constants of integration. In the time interval $[0, T_1]$ (when $S = A = 0$), the values of these constants can be computed using knowledge about the initial values of the three parameters, but in the intervals $(T_1, T_2]$, $(T_2, T_B]$ and (T_B, ∞), C_1, C_2 and C_3 can only be given specific numerical values by making continuity assumptions. Fortunately, it is "common sense" that the robot cannot instantaneously shift its own position or bearing. However, it is not difficult to imagine another scenario where some external action (such as giving the robot a good shove from behind) *does* cause an instantaneous shift in position (at least at the level of detail at which we wish to model the domain).

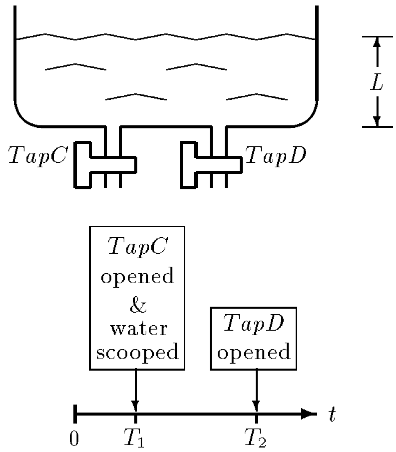
The formalism described below provides a general purpose default reasoning mechanism which allows us to infer the extra common sense information that the parameters EWP, NSP and $Bearing$ are continuous at times T_1, T_2 and T_B (when particular actions or events cause discontinuities in some or all of their derivatives, or in other unrelated parameters), without forbidding, in principle, the existence of other actions or events that could potentially cause discontinuities in the robot's position or orientation. Putting all this together, it allows us to infer the explicit values of EWP, NSP and $Bearing$ at any time t, and (assuming that $D_W < \frac{2S^+}{A^+}$, so that the robot doesn't turn in a full circle and miss the wall) it allows us to infer that

$$T_B = T_2 + \tfrac{1}{A^+}\arccos(1 - \tfrac{A^+.D_W}{S^+})$$

The formalism is "elaboration tolerant" in the sense that, at some later date, we can represent new knowledge about other effects of the robot's actions, or about other occurrences of actions or events at specific times, simply by adding to the existing set of domain-dependent axioms.

2.2 THE WATER TANK EXAMPLE

This example involves an open-top water tank. In the bottom of the tank are two taps, $TapC$ and $TapD$, both of which (when open) discharge water out of the tank at a rate proportional to the level of water in it (i.e. at a rate proportional to the water pressure at the bottom of the tank). Initially, both taps are closed, and the level of water in the tank is L. At time T_1, half the water is (instantaneously) scooped out of the tank, and, simultaneously, $TapC$ is opened. At time T_2, $TapD$ is also opened.



A mathematical model of this domain can be formulated using three parameters. $Level(t)$ is the function representing the level of water in the tank, and $FlowC(t)$ and $FlowD(t)$ represent the water flow through $TapC$ and $TapD$ respectively. Let K be the constant of proportionality between the level of water and the flow through either of the taps when open. The following equations are each applicable in the circumstances indicated

Always:	$Level'(t) = -(FlowC(t) + FlowD(t))$	(8)
$TapC$ closed:	$FlowC(t) = 0$	(9)
$TapC$ open:	$FlowC(t) = K.Level(t)$	(10)
$TapD$ closed:	$FlowD(t) = 0$	(11)
$TapD$ open:	$FlowD(t) = K.Level(t)$	(12)

In the interval $[0, T_1]$ the relevant equations are (8), (9) and (11). Their simultaneous solution is

$$Level(t) = C_4 \quad \text{(S1)}$$
$$FlowC(t) = 0 \quad \text{(S2)}$$
$$FlowD(t) = 0 \quad \text{(S3)}$$

where C_4 is an arbitrary constant of integration. In fact, since we have also been given the initial condition $Level(0) = L$, we know that $C_4 = L$.

At time T_1, $TapC$ is opened, so that in the interval $(T_1, T_2]$ the equations to solve are (8), (10) and (11). Their simultaneous solution is

$$\begin{aligned} Level(t) &= C_5.e^{-Kt} \\ FlowC(t) &= C_5.Ke^{-Kt} \\ FlowD(t) &= 0 \end{aligned}$$

where C_5 is an arbitrary constant of integration. We also know that the water level in the tank is instantaneously decreased from L to $\frac{L}{2}$ at T_1 by a "scoop" action. Hence, with a little algebra, it is easy to see that $C_5 = \frac{L}{2}e^{KT_1}$. So the overall solution for $t \in (T_1, T_2]$ is

$$Level(t) = \frac{L}{2}e^{-K(t-T_1)} \tag{S4}$$
$$FlowC(t) = \frac{LK}{2}e^{-K(t-T_1)} \tag{S5}$$
$$FlowD(t) = 0 \tag{S6}$$

In the interval (T_2, ∞), both $TapC$ and $TapD$ are open, and the equations to solve are (8), (10) and (12). Their simultaneous solution is

$$\begin{aligned} Level(t) &= C_6.e^{-2Kt} \\ FlowC(t) &= C_6.Ke^{-2Kt} \\ FlowD(t) &= C_6.Ke^{-2Kt} \end{aligned}$$

where again C_6 is an arbitrary real valued constant.

What value should C_6 take? Everyday knowledge about taps and tanks tells us that $Level(t)$ is continuous at T_2. In other words, we know that by itself the action of turning on $TapD$ will only cause water to disappear from the tank gradually, not instantaneously. (Although, at the level of physical detail we have chosen to incorporate in our mathematical model, the action of scooping water from the tank *does* cause the water level to drop instantaneously.) Hence $C_6 = \frac{L}{2}e^{K(T_1+T_2)}$, and the overall solution for $t \in (T_2, \infty)$ is

$$Level(t) = \frac{L}{2}e^{K(T_1+T_2-2t)} \tag{S7}$$
$$FlowC(t) = \frac{LK}{2}e^{K(T_1+T_2-2t)} \tag{S8}$$
$$FlowD(t) = \frac{LK}{2}e^{K(T_1+T_2-2t)} \tag{S9}$$

Notice that, if instead we had wished to keep $FlowD$ and its derivatives continuous at T_2, we could have made the assignment $C_6 = 0$. Alternatively, if we had wished to keep the first derivatives $Level'$ and $FlowC'$ continuous, we could have made the assignment $C_6 = \frac{L}{4}e^{K(T_1+T_2)}$. The important general point here is that in many mathematical models, it is possible to trade discontinuities in some parameters for discontinuities in others, using alternative assignments of values to (arbitrary) constants of integration. It is only common sense, or extra knowledge about the physical reality of the domain being modelled, that allows us to pick the right assignments from the different sets of possibilities.

The formalism described below allows us to include equations (8)–(12), along with their conditions of applicability, directly in the domain-dependent axiomatisation. The default reasoning method it incorporates allows us to infer that there is a discontinuity in $Level$ at T_1 (an instantaneous change from the value L to the value $\frac{L}{2}$ caused by the action of scooping water from the tank), but that $Level$ is continuous at T_2 (although its derivatives are not). In doing so, it correctly eliminates the anomalous models of the domain (which would otherwise be sanctioned by (8)–(12)) in which all (or half) the remaining water in the tank instantaneously disappears at T_2, thus illegally "trading" the discontinuity in $FlowD$ (or the discontinuities in $Level'$ and $FlowC'$) at T_2 for discontinuities in $Level$ and $FlowC$. Hence it allows us to correctly infer the values of $Level$, $FlowC$, $FlowD$ and their derivatives at any time t.

3 AN EXTENDED EVENT CALCULUS

The event calculus presented in this section is written in a sorted predicate calculus, with sorts as summarised in the following table.

NAME OF SORT	SYMBOL	VARIABLES
Actions	$\mathcal{A}$	$a, a_1, a_2, \ldots$
Fluents	$\mathcal{F}$	$f, f_1, f_2, \ldots$
Times	$\mathcal{T}$	$t, t_1, t_2, \ldots$
Parameters	$\mathcal{P}$	$p, p_1, p_2, \ldots$
Reals	$\mathcal{R}$	$r, r_1, r_2, \ldots$
Domain objects	$\mathcal{X}$	$x, x_1, x_2, \ldots$

Models will be considered only in which terms of sort $\mathcal{R}$ and $\mathcal{T}$ are interpreted as real and non-negative real numbers respectively. The sorting of the predicate symbols[1] in the language can be understood from their

[1] As mentioned previously, this formalism builds on several previous axiomatisations. However, some predicate and function names have been changed for the sake of clarity. *InitialisedTrue* and *InitialisedFalse* correspond to $Initially_P$ and $Initially_N$ in (Shanahan 1997), and *BreaksTo* and *Value* are analogous to *InstantEffect* and *Function* in (Miller 1996).

arguments in the axioms below, in which all variables are assumed to be universally quantified with maximum scope unless otherwise stated. The function symbols $Value : \mathcal{P} \times \mathcal{T} \mapsto \mathcal{R}$ and $\delta : \mathcal{P} \mapsto \mathcal{P}$ are also introduced. The term $Value(\delta(P), T)$ represents the numerical value of the first derivative of parameter P at time T.

The six core event calculus axioms (EC1)–(EC6) below, which do not directly concern continuous change, are domain-independent, i.e. included in every theory. For the sake of generality, this part of the axiomatisation includes a mechanism, inspired by Kartha and Lifschitz's work (1994) and Sandewall's notion of *occlusion* (Sandewall 1994), for dynamically adding fluents to (and removing fluents from) the "frame" (i.e. the set of fluents subject to the "commonsense law of inertia"). It was first introduced into the event calculus by Shanahan (1995). At time 0, exactly those fluents which have been *initialised true* or *initialised false* belong to the frame. These mechanisms are useful, for example, in dealing with domain constraints and non-deterministic actions (Shanahan 1995). However, in the example domains used in this paper, all fluents are permanently subject to the commonsense law of inertia, so that the (universally quantified) sentences $[InitialisedTrue(f) \vee InitialisedFalse(f)]$ and $\neg Releases(a, f, t)$ may safely be assumed or added to the axiomatisation. Hence axioms (EC1)–(EC6) simply express the following: (1) Fluents which initially hold, or which have been initiated by an occurrence of an action, continue to hold until an occurrence of an action which terminates them. (2) Fluents which do not initially hold, or which have been terminated by an occurrence of an action, continue not to hold until an occurrence of an action which initiates them.

$$HoldsAt(f, t) \leftarrow [InitialisedTrue(f) \wedge \neg Clipped(0, f, t)] \quad \text{(EC1)}$$

$$\neg HoldsAt(f, t) \leftarrow [InitialisedFalse(f) \wedge \neg Declipped(0, f, t)] \quad \text{(EC2)}$$

$$HoldsAt(f, t_2) \leftarrow [Happens(a, t_1) \wedge Initiates(a, f, t_1) \wedge t_1 < t_2 \wedge \neg Clipped(t_1, f, t_2)] \quad \text{(EC3)}$$

$$\neg HoldsAt(f, t_2) \leftarrow [Happens(a, t_1) \wedge Terminates(a, f, t_1) \wedge t_1 < t_2 \wedge \neg Declipped(t_1, f, t_2)] \quad \text{(EC4)}$$

$$Clipped(t_1, f, t_2) \leftrightarrow \exists a, t[Happens(a, t) \wedge t_1 < t < t_2 \wedge [Terminates(a, f, t) \vee Releases(a, f, t)]] \quad \text{(EC5)}$$

$$Declipped(t_1, f, t_2) \leftrightarrow \exists a, t[Happens(a, t) \wedge t_1 < t < t_2 \wedge [Initiates(a, f, t) \vee Releases(a, f, t)]] \quad \text{(EC6)}$$

This basic event calculus can be extended to deal with continuous change as follows. To respect the convention that actions take effect immediately *after* they occur, it is necessary to axiomatise the mathematical constraint that, at every time-point (including those at which actions occur), the function associated with each parameter is left-hand continuous:

$$LeftContinuous(p, t) \quad \text{(EC7)}$$

To describe instantaneous changes in the values of parameters at times when actions occur, and discontinuities in their corresponding functions of time, the predicates *BreaksTo* and *Breaks* are introduced. Both are minimised. $BreaksTo(A, P, T, R)$ should be read as 'at time T, an occurrence of action A will cause parameter P to instantaneously take on value R'. More precisely, Axiom (EC10) below states that if A also happens at time T, then R is the value of the right-hand limit of P at T. *BreaksTo* is used, for example, to describe the effects of a "scoop" action on the parameter *Level* in the water tank example. $Breaks(A, P, T)$ can be read as 'at time T, action A potentially causes a discontinuity in parameter P'. The following domain-independent axioms make direct use of *BreaksTo* and *Breaks*. Axioms (EC8) and (EC9) can be likened to 'frame axioms' for parameters. Axiom (EC12) states the relationship between *BreaksTo* and *Breaks*, and Axiom (EC12) states that if an action potentially causes a discontinuity in a given parameter, it also potentially causes discontinuities in its higher derivatives.

$$\neg[Happens(a, t) \wedge Breaks(a, p, t)] \rightarrow Continuous(p, t) \quad \text{(EC8)}$$

$$\neg[Happens(a, t) \wedge Breaks(a, \delta(p), t)] \rightarrow Differentiable(p, t) \quad \text{(EC9)}$$

$$[BreaksTo(a, p, t, r) \wedge Happens(a, t)] \rightarrow RightLimit(p, t, r) \quad \text{(EC10)}$$

$$BreaksTo(a,p,t,r) \rightarrow Breaks(a,p,t) \quad \text{(EC11)}$$

$$Breaks(a,p,t) \rightarrow Breaks(a,\delta(p),t) \quad \text{(EC12)}$$

For any given time point T, it is useful to be able to refer to the next point after T at which an action occurs, if there is such a point. Axioms (EC13), (EC14) and (EC15) state that if any action occurs at any time point after T, then the term $Next(T)$ refers to the least such time point. (Such points are somewhat analogous to the "least natural time points" discussed in (Reiter 1996).)

$$t < Next(t) \quad \text{(EC13)}$$

$$[t < t_1 \wedge t_1 < Next(t)] \rightarrow \neg Happens(a,t_1) \quad \text{(EC14)}$$

$$[Happens(a_1,t_1) \wedge t < t_1] \rightarrow \exists a. Happens(a, Next(t)) \quad \text{(EC15)}$$

Finally, the standard mathematical definitions of *Continuous*, *Differentiable*, *LeftContinuous* and *RightLimit* are straightforwardly axiomatised using the function symbols *Value* and δ:

$$Continuous(p,t) \leftrightarrow \forall r \exists t_1 \forall t_2 [[|t - t_2| < t_1 \wedge 0 < r] \rightarrow |Value(p,t) - Value(p,t_2)| < r] \quad \text{(A1)}$$

$$Differentiable(p,t) \leftrightarrow \forall r \exists t_1 \forall t_2 [[0 < |t - t_2| < t_1 \wedge 0 < r] \rightarrow |(\frac{Value(p,t) - Value(p,t_2)}{t - t_2}) - Value(\delta(p),t)| < r] \quad \text{(A2)}$$

$$LeftContinuous(p,t) \leftrightarrow \forall r \exists t_1 \forall t_2 [[t_2 < t \wedge (t - t_2) < t_1 \wedge 0 < r] \rightarrow |Value(p,t) - Value(p,t_2)| < r] \quad \text{(A3)}$$

$$RightLimit(p,t,r) \leftrightarrow \forall r_1 \exists t_1 \forall t_2 [[t < t_2 \wedge (t_2 - t) < t_1 \wedge 0 < r_1] \rightarrow |Value(p,t_2) - r| < r_1] \quad \text{(A4)}$$

4 DOMAIN-DEPENDENT AXIOMS

4.1 AN AXIOMATISATION OF THE ROBOT EXAMPLE

The robot described in Section 2 can start to move forward or start to turn at any speed, so that two parameterised action symbols are needed, $ChangeSpeed : \mathcal{R} \mapsto \mathcal{A}$ and $StartTurn : \mathcal{R} \mapsto \mathcal{A}$, where, for example, $ChangeSpeed(S)$ signifies the action of "changing the forward speed to S". We can describe the properties that these actions actions initiate and terminate using the parameterised fluent symbols $Moving : \mathcal{R} \mapsto \mathcal{F}$ and $Turning : \mathcal{R} \mapsto \mathcal{F}$, as follows

$$Initiates(ChangeSpeed(r), Moving(r), t) \quad \text{(R1)}$$
$$Initiates(StartTurn(r), Turning(r), t) \quad \text{(R2)}$$
$$Terminates(ChangeSpeed(r_1), Moving(r_2), t) \leftarrow r_1 \neq r_2 \quad \text{(R3)}$$
$$Terminates(StartTurn(r_1), Turning(r_2), t) \leftarrow r_1 \neq r_2 \quad \text{(R4)}$$

The following axioms express the fact that a *ChangeSpeed* action can cause discontinuities in the first derivatives of the parameters *NSP* and *EWP*, whereas a *StartTurn* action can cause discontinuities in the first derivatives of all three parameters in the domain:

$$Breaks(ChangeSpeed(r), \delta(NSP), t) \quad \text{(R5)}$$
$$Breaks(StartTurn(r), \delta(NSP), t) \quad \text{(R6)}$$
$$Breaks(ChangeSpeed(r), \delta(EWP), t) \quad \text{(R7)}$$
$$Breaks(StartTurn(r), \delta(EWP), t) \quad \text{(R8)}$$
$$Breaks(StartTurn(r), \delta(Bearing), t) \quad \text{(R9)}$$

The constraints (1)–(3) of Section 2 can now be axiomatised as follows:

$$Value(\delta(EWP),t) = r.\cos(Value(Bearing,t)) \leftarrow HoldsAt(Moving(r),t) \quad \text{(R10)}$$
$$Value(\delta(NSP),t) = r.\sin(Value(Bearing,t)) \leftarrow HoldsAt(Moving(r),t) \quad \text{(R11)}$$
$$Value(\delta(Bearing),t) = r \leftarrow HoldsAt(Turning(r),t) \quad \text{(R12)}$$

Happens can be used to state that the robot changes speed at time T_1, starts to turn at time T_2, and stops moving whenever it hits the wall:

$$Happens(ChangeSpeed(S^+), T_1) \quad \text{(R13)}$$
$$Happens(StartTurn(A^+), T_2) \quad \text{(R14)}$$
$$[Happens(ChangeSpeed(0),t) \wedge Happens(StartTurn(0),t)] \leftarrow [Value(NSP,t) = D_W \wedge Value(\delta(NSP),t) > 0] \quad \text{(R15)}$$

Finally, axioms are needed stating various initial conditions, and expressing uniqueness-of-names properties (using the "UNA[...]" notation from (Baker 1991)) for all action, fluent and parameter symbols.

$$[\mathit{InitialisedTrue}(\mathit{Moving}(0)) \wedge \mathit{InitialisedTrue}(\mathit{Turning}(0))] \quad \text{(R16)}$$
$$[\mathit{InitialisedFalse}(\mathit{Moving}(r)) \wedge \mathit{InitialisedFalse}(\mathit{Turning}(r))] \leftarrow r \neq 0 \quad \text{(R17)}$$
$$\mathit{Value}(\mathit{NSP}, 0) = 0 \wedge \mathit{Value}(\mathit{EWP}, 0) = 0 \wedge \mathit{Value}(\mathit{Bearing}, 0) = 0 \quad \text{(R18)}$$
$$\text{UNA}[\mathit{ChangeSpeed}, \mathit{StartTurn}] \quad \text{(R19)}$$
$$\text{UNA}[\mathit{Moving}, \mathit{Turning}] \quad \text{(R20)}$$
$$\text{UNA}[\mathit{Bearing}, \mathit{NSP}, \mathit{EWP}, \delta] \quad \text{(R21)}$$

4.2 AN AXIOMATISATION OF THE WATER TANK EXAMPLE

The following constant symbols will be used to axiomatise the water tanks example. $TurnOnC$, $TurnOnD$ and $Scoop$ of sort $\mathcal{A}$, $OpenC$ and $OpenD$ of sort $\mathcal{F}$, and $Level$, $FlowC$ and $FlowD$ of sort $\mathcal{P}$. The (direct) effects of turning on either tap can be described as follows:

$$\mathit{Initiates}(\mathit{TurnOnC}, \mathit{OpenC}, t) \quad \text{(T1)}$$
$$\mathit{Initiates}(\mathit{TurnOnD}, \mathit{OpenD}, t) \quad \text{(T2)}$$

The action of scooping water from the tank has no effect on the fluents in the domain, but instantaneously effects the parameter $Level$. If the level in the tank is greater than or equal to $\frac{L}{2}$, a $Scoop$ action reduces the level by $\frac{L}{2}$. For the purpose of illustration, we will further suppose that if the level in the tank is less than $\frac{L}{2}$, a scoop action removes all the water from the tank:

$$\mathit{BreaksTo}(\mathit{Scoop}, \mathit{Level}, t, \mathit{Value}(\mathit{Level}, t) - \tfrac{L}{2}) \leftarrow \mathit{Value}(\mathit{Level}, t) \geq \tfrac{L}{2} \quad \text{(T3)}$$
$$\mathit{BreaksTo}(\mathit{Scoop}, \mathit{Level}, t, 0) \leftarrow \mathit{Value}(\mathit{Level}, t) < \tfrac{L}{2} \quad \text{(T4)}$$

As well as causing a discontinuity in $FlowC$, the action $TurnOnC$ causes a discontinuity in the first derivative $Level'$, and, if $TapD$ is open, in the first derivative $FlowD'$. The effects of $TurnOnD$ are analogous:

$$\mathit{Breaks}(\mathit{TurnOnC}, \mathit{FlowC}, t) \quad \text{(T5)}$$
$$\mathit{Breaks}(\mathit{TurnOnC}, \delta(\mathit{Level}), t) \quad \text{(T6)}$$
$$\mathit{Breaks}(\mathit{TurnOnC}, \delta(\mathit{FlowD}), t) \leftarrow \mathit{HoldsAt}(\mathit{OpenD}, t) \quad \text{(T7)}$$
$$\mathit{Breaks}(\mathit{TurnOnD}, \mathit{FlowD}, t) \quad \text{(T8)}$$
$$\mathit{Breaks}(\mathit{TurnOnD}, \delta(\mathit{Level}), t) \quad \text{(T9)}$$
$$\mathit{Breaks}(\mathit{TurnOnD}, \delta(\mathit{FlowC}), t) \leftarrow \mathit{HoldsAt}(\mathit{OpenC}, t) \quad \text{(T10)}$$

Constraints (8)–(12) of Section 2 are axiomatised as follows:

$$\mathit{Value}(\delta(\mathit{Level}), t) = -(\mathit{Value}(\mathit{FlowC}, t) + \mathit{Value}(\mathit{FlowD}, t)) \quad \text{(T11)}$$
$$\mathit{HoldsAt}(\mathit{OpenC}, t) \rightarrow \mathit{Value}(\mathit{FlowC}, t) = K.\mathit{Value}(\mathit{Level}, t) \quad \text{(T12)}$$
$$\neg \mathit{HoldsAt}(\mathit{OpenC}, t) \rightarrow \mathit{Value}(\mathit{FlowC}, t) = 0 \quad \text{(T13)}$$
$$\mathit{HoldsAt}(\mathit{OpenD}, t) \rightarrow \mathit{Value}(\mathit{FlowD}, t) = K.\mathit{Value}(\mathit{Level}, t) \quad \text{(T14)}$$
$$\neg \mathit{HoldsAt}(\mathit{OpenD}, t) \rightarrow \mathit{Value}(\mathit{FlowD}, t) = 0 \quad \text{(T15)}$$

The action occurrences are:

$$\mathit{Happens}(\mathit{Scoop}, T_1) \quad \text{(T16)}$$
$$\mathit{Happens}(\mathit{TurnOnC}, T_1) \quad \text{(T17)}$$
$$\mathit{Happens}(\mathit{TurnOnD}, T_2) \quad \text{(T18)}$$

Finally, the initial conditions and uniqueness-of-names axioms are:

$$\mathit{InitialisedFalse}(\mathit{OpenC}) \quad \text{(T19)}$$
$$\mathit{InitialisedFalse}(\mathit{OpenD}) \quad \text{(T20)}$$
$$\mathit{Value}(\mathit{Level}, 0) = L \quad \text{(T21)}$$
$$\text{UNA}[\mathit{TurnOnC}, \mathit{TurnOnD}, \mathit{Scoop}] \quad \text{(T22)}$$
$$\text{UNA}[\mathit{OpenC}, \mathit{OpenD}] \quad \text{(T23)}$$
$$\text{UNA}[\mathit{Level}, \mathit{FlowC}, \mathit{FlowD}, \delta] \quad \text{(T24)}$$

5 THE CIRCUMSCRIPTION POLICY $\mathrm{CIRC}_{\mathrm{CEC}}$

By themselves, the axioms in Section 4.1 are not sufficient to infer the robot's trajectory. In general, a default reasoning mechanism will also be required which models various default assumptions about such domains. The circumscription policy used here is inspired by a solution to the frame problem described by Kartha and Lifschitz (1995), which is related to Sandewall's idea of *filter preferential entailment* (Sandewall 1989b). This has been adapted for use with the event calculus, and extended so that it also models the assumptions that by default a given action does not occur at a given time point, and that by default a given action occurrence does not result in a discontinuity for a given parameter. Given a collection of domain-dependent axioms $\mathcal{D}$ similar to those in the previous section, the circumscription policy is:

$$
\begin{aligned}
&CIRC[Nar(\mathcal{D})\ ;\ Happens]\\
&\quad\wedge CIRC[Eff(\mathcal{D})\ ;\ Initiates, Terminates, Releases]\\
&\quad\wedge CIRC[(Inst(\mathcal{D}) \wedge \text{(EC11)} \wedge \text{(EC12)})\ ;\\
&\qquad\qquad Breaks\ ;\ BreaksTo]\\
&\quad\wedge Con(\mathcal{D})\ \wedge\ Una(\mathcal{D})\ \wedge\ [\text{(EC1)} \wedge \ldots \wedge \text{(EC15)}]
\end{aligned}
$$

We will abbreviate this to $CIRC_{CEC}[\mathcal{D}]$. In $CIRC_{CEC}[\mathcal{D}]$, the term "$Nar(\mathcal{D})$" stands for (the conjunction of) those domain-specific axioms describing the "narrative" (e.g. *Happens* facts and statements about the initial values of fluents or parameters), "$Eff(\mathcal{D})$" stands for those axioms describing the effects of actions on fluents (using *Initiates*, *Terminates* and *Releases*), "$Inst(\mathcal{D})$" stands for those axioms describing the instantaneous effects of actions on parameters (using *Breaks* and *BreaksTo*), "$Con(\mathcal{D})$" stands for axioms describing mathematical constraints between parameters during different circumstances (e.g. when a tap is open, its flow is proportional to the water level in the tank), and "$Una(\mathcal{D})$" stands for the uniqueness-of-names axioms. So, if $\mathcal{D}_R$ is the set of axioms describing the robot,

$$
\begin{aligned}
Nar(\mathcal{D}_R) &= [\text{(R13)} \wedge \ldots \wedge \text{(R18)}]\\
Eff(\mathcal{D}_R) &= [\text{(R1)} \wedge \ldots \wedge \text{(R4)}]\\
Inst(\mathcal{D}_R) &= [\text{(R5)} \wedge \ldots \wedge \text{(R9)}]\\
Con(\mathcal{D}_R) &= [\text{(R10)} \wedge \ldots \wedge \text{(R12)}]\\
Una(\mathcal{D}_R) &= [\text{(R19)} \wedge \ldots \wedge \text{(R21)}]
\end{aligned}
$$

and if $\mathcal{D}_T$ is the set of axioms describing the water tank example,

$$
\begin{aligned}
Nar(\mathcal{D}_T) &= [\text{(T16)} \wedge \ldots \wedge \text{(T21)}]\\
Eff(\mathcal{D}_T) &= [\text{(T1)} \wedge \text{(T2)}]\\
Inst(\mathcal{D}_T) &= [\text{(T3)} \wedge \ldots \wedge \text{(T10)}]\\
Con(\mathcal{D}_T) &= [\text{(T11)} \wedge \ldots \wedge \text{(T15)}]\\
Una(\mathcal{D}_T) &= [\text{(T22)} \wedge \ldots \wedge \text{(T24)}]
\end{aligned}
$$

Whenever $Nar(\mathcal{D})$, $Eff(\mathcal{D})$ and $Inst(\mathcal{D})$ are of a certain general form, we can prove general properties of $CIRC_{CEC}[\mathcal{D}]$ which allow its consequences to be computed using classical deduction. The three propositions below are applicable to a wide class of domains which includes both $\mathcal{D}_R$ and $\mathcal{D}_T$. (Strictly speaking, to fit the conditions of the propositions, some of the domain-dependent axioms in $\mathcal{D}_R$ and $\mathcal{D}_T$ must be re-written in a slightly different form. For example, (T3) is re-written as

$$
\begin{aligned}
&BreaksTo(a,p,t,r) \leftarrow \qquad\qquad \text{(T3}'\text{)}\\
&\quad [Value(Level,t) \geq \tfrac{L}{2}\ \wedge\ a = Scoop\\
&\quad \wedge\ p = Level\ \wedge\ r = Value(Level,t) - \tfrac{L}{2}]
\end{aligned}
$$

and similar syntactic transformations are applied to the other clauses partially defining *Initiates*, *Terminates*, *Happens*, *Breaks* or *BreaksTo*.)

Proposition 1 Let S be the conjunction of (EC11) and (EC12) with the following sentences:

$$
\begin{aligned}
&Breaks(a,p,t) \leftarrow \Phi_1(a,p,t) && \text{(S1)}\\
&\qquad \vdots && \vdots\\
&Breaks(a,p,t) \leftarrow \Phi_k(a,p,t) && \text{(Sk)}\\
&BreaksTo(a,p,t,r) \leftarrow \Phi_{k+1}(a,p,t,r) && \text{(Sk+1)}\\
&\qquad \vdots && \vdots\\
&BreaksTo(a,p,t,r) \leftarrow \Phi_m(a,p,t,r) && \text{(Sm)}
\end{aligned}
$$

where a, p and t are the only variables which appear free in the formulae $\Phi_1(a,p,t), \ldots, \Phi_k(a,p,t)$, where a, p, t and r are the only variables which appear free in the formulae $\Phi_{k+1}(a,p,t,r)$, $\ldots, \Phi_m(a,p,t,r)$, and where none of $\Phi_1(a,p,t), \ldots$, $\Phi_k(a,p,t)$ or $\Phi_{k+1}(a,p,t,r), \ldots, \Phi_m(a,p,t,r)$ mention the predicates *Breaks* or *BreaksTo*. Then $CIRC[S\ ;\ Breaks\ ;\ BreaksTo]$ entails the following sentence S_{comp}:

$$
\begin{aligned}
&Breaks(a,p,t) \leftrightarrow\\
&\quad [\exists p_1[p = \delta(p_1)\ \wedge\ Breaks(a,p_1,t)] \vee\\
&\quad \Phi_1(a,p,t)\ \vee \ldots \vee\ \Phi_k(a,p,t) \vee\\
&\quad \exists r[\Phi_{k+1}(a,p,t,r)\ \vee \ldots \vee\ \Phi_m(a,p,t,r)]]
\end{aligned}
$$

Proof: (*Notation:* Let $\overline{x}$ be the tuple of free variables in the formula $\Phi(\overline{x})$. Then in the following proof, $\overline{\Phi}$ refers to the *predicate expression* $\lambda\overline{x}.\Phi(\overline{x})$ (see (Lifschitz 1995) for definition). Hence, given a model M and an appropriately sorted tuple of domain objects $\overline{\chi}$, the statement $\overline{\chi} \in M[\![\overline{\Phi}]\!]$ signifies that $M,v \models \overline{\Phi}(\overline{x})$ for all variable assignments v such that $v(\overline{x}) = \overline{\chi}$.)

The if half of S_{comp} follows directly from S. It remains to prove the only-if half. Suppose there is some model M of S which does not satisfy the only-if half of S_{comp}. Then there must be some $\langle\alpha,\rho,\tau\rangle \in M[\![Breaks]\!]$ such that:

(i) there is no ρ' such that $\rho = M[\![\delta]\!](\rho')$ and $\langle\alpha,\rho',\tau\rangle \in M[\![Breaks]\!]$

(ii) for all $i \leq k$, $\langle\alpha,\rho,\tau\rangle \notin M[\![\overline{\Phi_i}]\!]$

(iii) for all $k < i \leq m$, there is no π such that $\langle\alpha,\rho,\tau,\pi\rangle \in M[\![\overline{\Phi_i}]\!]$

Furthermore, since M is a model of S, it follows that for each parameter ρ'' such that $\rho = M[\![\delta]\!]^n(\rho'')$ for some $n \geq 1$:

(iv) $\langle \alpha, \rho'', \tau \rangle \notin M[\![Breaks]\!]$

(v) for all $i \leq k$, $\langle \alpha, \rho'', \tau \rangle \notin M[\![\overline{\Phi_i}]\!]$

(vi) for all $k < i \leq m$, there is no π such that $\langle \alpha, \rho'', \tau, \pi \rangle \in M[\![\overline{\Phi_i}]\!]$

(Otherwise, we could put $\rho' = M[\![\delta]\!]^{n-1}(\rho'')$ and (i) would not be satisfied.) Hence we can construct a smaller model than M by removing $\langle \alpha, \rho, \tau \rangle$ from $M[\![Breaks]\!]$. In order to satisfy (E13), this necessitates the additional removal of all tuples of the form $\langle \alpha, \rho, \tau, \pi \rangle$ from $M[\![BreaksTo]\!]$. More precisely, let M' be an interpretation obtained from M in the following way:

- M' agrees with M on the interpretation of all predicate, constant and function symbols except *Breaks* and *BreaksTo*.
- $\langle \alpha', \rho', \tau' \rangle \in M'[\![Breaks]\!]$ if and only if $\langle \alpha', \rho', \tau' \rangle \in M[\![Breaks]\!]$ and $\langle \alpha', \rho', \tau' \rangle \neq \langle \alpha, \rho, \tau \rangle$.
- $\langle \alpha', \rho', \tau', \pi \rangle \in M'[\![BreaksTo]\!]$ if and only if $\langle \alpha', \rho', \tau', \pi \rangle \in M[\![BreaksTo]\!]$ and $\langle \alpha', \rho', \tau' \rangle \neq \langle \alpha, \rho, \tau \rangle$.

It is easily verified that M' is also a model of S. Since $M'[\![Breaks]\!]$ is a strict subset of $M[\![Breaks]\!]$, M' is preferable to M according to the circumscription policy. Therefore M cannot be a model of $CIRC[S\ ;\ Breaks\ ;\ BreaksTo]$, and the proposition holds. □

Proposition 2 Let S be the conjunction of the following sentences:

$$Happens(a,t) \leftarrow \Phi_1(a,t) \quad \text{(S1)}$$
$$\vdots$$
$$Happens(a,t) \leftarrow \Phi_n(a,t) \quad \text{(Sn)}$$

where a and t are the only variables which (possibly) appear free in the formulae $\Phi_1(a,t), \ldots, \Phi_n(a,t)$, and where none of $\Phi_1(a,t), \ldots, \Phi_n(a,t)$ mention the predicate *Happens*. Then $CIRC[S\ ;\ Happens]$ entails the following sentence:

$$Happens(a,t) \leftrightarrow [\Phi_1(a,t) \vee \ldots \vee \Phi_n(a,t)]$$

Proof: The proposition follows directly from Proposition 3.1.1 in (Lifschitz 1995). □

Proposition 3 Let S be the conjunction of the following sentences:

$$Initiates(a,f,t) \leftarrow \Phi_1(a,f,t) \quad \text{(S1)}$$
$$\vdots$$
$$Initiates(a,f,t) \leftarrow \Phi_k(a,f,t) \quad \text{(Sk)}$$
$$Terminates(a,f,t) \leftarrow \Phi_{k+1}(a,f,t) \quad \text{(Sk+1)}$$
$$\vdots$$
$$Terminates(a,f,t) \leftarrow \Phi_m(a,f,t) \quad \text{(Sm)}$$
$$Releases(a,f,t) \leftarrow \Phi_{m+1}(a,f,t) \quad \text{(Sm+1)}$$
$$\vdots$$
$$Releases(a,f,t) \leftarrow \Phi_n(a,f,t) \quad \text{(Sn)}$$

where a, f and t are the only variables which (possibly) appear free in the formulae $\Phi_1(a,f,t), \ldots, \Phi_n(a,f,t)$, and where none of $\Phi_1(a,f,t), \ldots, \Phi_n(a,f,t)$ mention the predicates *Initiates*, *Terminates* or *Releases*.

Then $CIRC[S\ ;\ Initiates, Terminates, Releases]$ entails the following three sentences:

$$Initiates(a,f,t) \leftrightarrow [\Phi_1(a,f,t) \vee \ldots \vee \Phi_k(a,f,t)]$$

$$Terminates(a,f,t) \leftrightarrow [\Phi_{k+1}(a,f,t) \vee \ldots \vee \Phi_m(a,f,t)]$$

$$Releases(a,f,t) \leftrightarrow [\Phi_{m+1}(a,f,t) \vee \ldots \vee \Phi_n(a,f,t)]$$

Proof: The proposition follows directly from Propositions 3.1.1 and 7.1.1 in (Lifschitz 1995). □

Using these results, it is not hard to generate classical derivations of sentences of the form $HoldsAt(F,T)$, $Happens(A,T)$ and $Value(P,T)=R$ as required. For example, an outline derivation of the sentence

$$Happens(ChangeSpeed(0), T_2 + \tfrac{1}{A^+}\arccos(1 - \tfrac{A^+.D_W}{S^+}))$$

from $CIRC_{CEC}[\mathcal{D}_R]$ is given in Appendix B.

6 SUMMARY AND DISCUSSION

A logical formalism for representing both discrete and continuous change has been presented which overcomes a number of drawbacks in existing logic-based formalisms. It permits the use of arbitrary formulae of the differential calculus without giving rise to anomalous models. In addition, it incorporates a solution to the frame problem which is robust in the presence of narrative, concurrent actions, non-deterministic actions, and domain constraints.

It is important to incorporate the notion of a derivative function in any comprehensive formalism for modelling domains with continuous change. Differential calculus is the primary tool for mathematical modelling in mainstream science and engineering. When a reasonably complex dynamic system is represented as a set of differential equations it is often not possible to obtain an analytical solution. Instead, numerical methods may be used, and the present formalism is a step towards integrating such computational techniques with systems for automated reasoning about actions. Furthermore, the information about the continuously varying aspects of a domain may be incomplete, in which case it may be more appropriate to use computational methods from Qualitative Reasoning (Weld 1990) (Kuipers 1994). Again, the notion of a derivative function is fundamental to the semantics of such systems.

The mechanisms incorporated in the formalism for reasoning about boundary conditions, using the predicates *Breaks* and *BreaksTo*, are similar to those introduced in (Miller 1996). However, whereas the discussion in (Miller 1996) was restricted to a particular case study, Propositions 1, 2 and 3 have allowed the effects of the circumscription policy $CIRC_{CEC}$ to be characterised for a wide class of domains. Moreover, they allow the use of standard, first-order proof-theoretic techniques (as opposed to the model-theoretic arguments of (Miller 1996), (Sandewall 1989a) or (Sandewall 1989b)) to ascertain logical consequences of a given domain description. Indeed, the overall structure of the example derivation given in Appendix B points towards a particular approach to designing algorithms for temporal projection, which would be sound with respect to the logical specification presented here. Such algorithms would compute forward in time in alternating "steps", each step being either a single time point identified by the *Next* function (see Section 3), or an open interval between such time points.

A great deal of work has already been done on algorithms which perform this kind of computation in the Qualitative Reasoning community (Weld 1990) (Kuipers 1994). In the terminology of qualitative reasoning, time points identified by the *Next* function may often coincide with *distinguished* or *landmark time-points* with respect to some continuously varying parameter. Algorithms for reasoning about landmark time-points and the associated landmark values of particular parameters are embedded in, for example, QSIM (Kuipers 1986), and in systems based on Qualitative Process Theory (Forbus 1984). Furthermore, qualitative process theory incorporates the ability to handle actions (Forbus 1989). Although some effort has been made to reconcile logic-based work in Reasoning about Action with that in Qualitative Reasoning (Crawford & Etherington 1992) (van Belleghem *et al* 1994), the two fields have yet to be properly integrated. Attempts to axiomatise qualitative reasoning are valuable here (Davis 1992), and it is hoped that the present paper can serve to further work in this vein.

Of course, the emphasis in Qualitative Reasoning is on reasoning with incomplete or qualitative information about relationships between parameters. In the present formalism, such information would manifest itself in the use of the inequality predicate and existentially quantified numerical variables in constraints between parameters, or in the use of appropriately defined "qualitative predicates". This is discussed further in Appendix A. Domain dependent qualitative constraints would appear in axioms analogous to (R10)–(R12) in the "*Con*" part of the theory (see Section 5). The fact that $Con(\mathcal{D})$ is outside the scope of any circumscription indicates that the minimisation policy is equally applicable to either qualitative or quantitative domain descriptions. It is therefore hoped that, as well as integrating the notions of discrete and continuous change, the present formalism is a step towards providing a unifying conceptual framework for qualitative, semi-qualitative and quantitative reasoning about continuous change. But further work needs to be done in order to substantiate this claim.

Acknowledgement

This research was funded by the U.K. Engineering and Physical Sciences Research Council (EPSRC).

References

A. Baker (1991), *Nonmonotonic Reasoning in the Framework of the Situation Calculus*, A.I. vol. 49, page 5, Elsevier Science Publishers.

J. Crawford and D. Etherington (1992), *Formalizing Reasoning about Change: A Qualitative Reasoning Approach*, Proceedings AAAI'92, pages 577-583.

E. Davis (1992a), *Axiomatising Qualitative Process Theory*, Proceedings KR'92, Morgan Kaufmann, pages 177-188.

E. Davis (1992b), *Infinite Loops in Finite Time: Some Observations*, Proceedings KR'92, Morgan Kaufmann.

K. Forbus (1984), *Qualitative Process Theory*, in Artificial Intelligence 24, reprinted in Weld and de Kleer (eds.), Readings in Qualitative Reasoning about Physical Systems, Morgan Kaufmann (1990), 1984.

K. Forbus (1989), *Introducing Actions into Qualitative Simulation*, Proceedings IJCAI'89, pages 1273-1278.

C. Herrmann and M. Thielscher (1996), *Reasoning* *about Continuous Processes*, Proceedings AAAI'96.

G.N. Kartha and V. Lifschitz (1994), *Actions with Indirect Effects (Preliminary Report)*, Proceedings KR'94, pages 341-350.

G.N. Kartha and V. Lifschitz (1995), *A Simple Formalization of Actions Using Circumscription*, Proceedings IJCAI'95, pages 1970-1975.

R. Kowalski and M. Sergot (1986), *A Logic-Based Calculus of Events*, New Generation Computing, vol. 4, page 267.

B. Kuipers (1986), *Qualitative Simulation*, A.I. vol. 29, pages 289-338, Elsevier Science Publishers.

B. Kuipers (1994), *Qualitative Reasoning: Modeling and Simulation with Incomplete Knowledge*, MIT Press.

V. Lifschitz (1995), *Circumscription*, in Handbook of Logic in Artificial Intelligence, ed.s D. Gabbay, C. Hogger and J.A. Robinson, Oxford University Press, pages 297-352.

R. Miller (1995), *Situation Calculus Specifications for Event Calculus Logic Programs*, in Proceedings of the Third International Conference on Logic Programming and Non-monotonic Reasoning, Lexington, KY, USA, pub. Springer Verlag.

R. Miller (1996), *A Case Study in Reasoning about Actions and Continuous Change*, Proceedings ECAI'96, pub. John Wiley & Sons, Ltd.

R. Miller and M. Shanahan (1994), *Narratives in the Situation Calculus*, in Journal of Logic and Computation, Special Issue on Actions and Processes, vol. 4 no. 5, Oxford University Press.

R. Miller and M. Shanahan (1996), *Reasoning about Discontinuities in the Event Calculus (Extended Version)*, `http://www-lp.doc.ic.ac.uk/UserPages/staff/rsm/abstract12.html`.

J. Pinto (1994), *Temporal Reasoning in the Situation Calculus*, PhD. Thesis, University of Toronto.

M. Rayner (1991), *On the Applicability of Nonmonotonic Logic to Formal Reasoning in Continuous Time*, A.I. vol. 49, pages 345-360, Elsevier Science.

R. Reiter (1991), *The Frame Problem in the Situation Calculus: a Simple Solution (Sometimes) and a Completeness Result for Goal Regression*, in Artificial Intelligence and Mathematical Theory of Computation: Papers in Honour of John McCarthy, ed. V. Lifschitz, Academic Press, page 418.

R. Reiter (1996), *Natural Actions, Concurrency and Continuous Time in the Situation Calculus*, Proceedings KR'96 (this proceedings), Morgan Kaufmann.

E. Sandewall (1989a), *Combining Logic and Differential Equations for Describing Real World Systems*, Proceedings KR'89, Morgan Kaufman.

E. Sandewall (1989b), *Filter Preferential Entailment for the Logic of Action in Almost Continuous Worlds*, Proceedings IJCAI'89, pages 894-899.

E. Sandewall (1994), *Features and Fluents*, Oxford University Press.

M. Shanahan (1990), *Representing Continuous Change in the Event Calculus*, Proceedings ECAI'90, pages 598-603.

M. Shanahan (1995), *A Circumscriptive Calculus of Events*, A.I. vol. 77, pages 249-284, Elsevier Science.

M. Shanahan (1996), *Robotics and the Common Sense Informatic Situation*, in Proceedings ECAI'96, pub. John Wiley & Sons, Ltd.

M. Shanahan (1997), *Solving the Frame Problem: A Mathematical Investigation of the Common Sense Law of Inertia*, MIT Press, (to appear).

Y. Shoham (1988), *Reasoning About Change: Time and Causation from the Standpoint of Artificial Intelligence*, MIT Press.

K. Van Belleghem, M. Deneker and D. De Schreye (1994), *Representing Continuous Change in the Abductive Event Calculus*, in Proceedings 1994 International Conference on Logic Programming, ed. P. Van Hentenrijck, pages 225-240.

D. Weld and J. de Kleer (editors) (1990), *Qualitative Reasoning about Physical Systems*, Morgan Kaufmann.

APPENDIX A – REPRESENTING QUALITATVE INFORMATION ABOUT PARAMETER BEHAVIOUR

This appendix outlines some preliminary ideas for incorporating qualitative information about parameter behaviour in domain descriptions, along the lines of (Kuipers 1986) and (Kuipers 1994). The discussion here is speculative – a more thorough investigation is needed to fully integrate the notions and ontologies of QR with those of action based logical formalisms.

Kuipers lists seven qualitative relationships between parameters as being particularly important as regards the qualitative description of a system's behaviour. These relationships are *add*, *mult*, *minus*, *d/dt*, *constant*, M^+ (monotonically increasing) and M^- (monotonically decreasing). These notions can be included in the present framework by appropriately extending the axiom set (A1)–(A4):

$$Add(p_1, p_2, p_3, t) \leftrightarrow \quad \text{(Q1)}$$
$$Value(p_1, t) + Value(p_2, t) = Value(p_3, t)$$

$$Mult(p_1, p_2, p_3, t) \leftrightarrow \quad \text{(Q2)}$$
$$Value(p_1, t).Value(p_2, t) = Value(p_3, t)$$

$$Minus(p_1, p_2, t) \leftrightarrow \quad \text{(Q3)}$$
$$Value(p_1, t) = -Value(p_2, t)$$

$$Derivative(p_1, p_2, t) \leftrightarrow \quad \text{(Q4)}$$
$$Value(\delta(p_1), t) = Value(p_2, t)$$

$$Constant(p, t) \leftrightarrow Value(\delta(p), t) = 0 \quad \text{(Q5)}$$

$$MonInc(p_1, p_2, t) \leftrightarrow \quad \text{(Q6)}$$
$$[[Value(\delta(p_1), t) > 0 \leftrightarrow Value(\delta(p_2), t) > 0] \wedge$$
$$[Value(\delta(p_1), t) < 0 \leftrightarrow Value(\delta(p_2), t) < 0]]$$

$$MonDec(p_1, p_2, t) \leftrightarrow \quad \text{(Q7)}$$
$$[[Value(\delta(p_1), t) > 0 \leftrightarrow Value(\delta(p_2), t) < 0] \wedge$$
$$[Value(\delta(p_1), t) < 0 \leftrightarrow Value(\delta(p_2), t) > 0]]$$

With such extra definitions, it is easy to include qualitative constraints in domain descriptions. For example, in the water tank example we might replace axiom (T12) with the sentence

$$HoldsAt(OpenC, t) \rightarrow MonInc(FlowC, Level, t)$$

to reflect the fact that, although we know that the flow through tap C decreases as the water level decreases, we do not know the exact mathematical relationship between the two parameters in this circumstance.

Another notion central to QR is that of a *landmark value* (w.r.t. a particular parameter). A landmark value, or simply "landmark", is a particular "qualitatively important" value for some parameter, typically where the first derivative changes sign. Symbolic names may be used for landmarks, since their exact numerical values may be unknown. In the present context, landmarks can be represented by extra constant or function symbols, along with suitable sets of ordering declarations. For example, qualitatively important values for the parameter *Level* in the water tank example are 0 and the top of the tank. Hence the "quantity space" for *Level* can be described by the sentence

$$0 < Top$$

where *Top* is an extra constant symbol of sort $\mathcal{R}$. In the terminology of QSIM, this quantity space gives rise to 15 possible qualitative values for the parameter *Level*, each of the form $\langle qmag, qdir \rangle$, where *qmag* is chosen from the set $\{(-\infty, 0), 0, (0, Top), Top, (Top, \infty)\}$, and *qdir* is chosen from the set $\{Inc, Std, Dec\}$ ("increasing", "steady" or "decreasing"). Qualitative values can be represented as fluents within the present framework by incorporating (6) domain independent axioms of the following form:

$$HoldsAt(QV(p, \langle (r_1, r_2), Inc \rangle), t) \leftrightarrow \quad \text{(Q8)}$$
$$[r_1 < Value(p, t) < r_2 \wedge Value(\delta(p), t) > 0]$$

$$HoldsAt(QV(p, \langle r, Inc \rangle), t) \leftrightarrow \quad \text{(Q9)}$$
$$[Value(p, t) = r \wedge Value(\delta(p), t) > 0]$$

The fluent $QV(Level, \langle (0, Top), Inc \rangle)$ is not included in the "frame". Conceptually, its truth value at any instant of time is instead determined by the values of *Level* and $\delta(Level)$. With such notation, axiom (T21) stating the initial value of *Level* could be replaced with an assertion such as

$$HoldsAt(QV(Level, \langle (0, Top), Std \rangle), 0)$$

or

$$\exists x. HoldsAt(QV(Level, \langle (0, Top), x \rangle), 0)$$

Similarly, we could describe when triggered events such as as an "overflow" occur with axioms of the form

$$Happens(Overflow, t) \leftarrow$$
$$HoldsAt(QV(Level, \langle Top, Inc \rangle), t)$$

Computationally, for domains involving both discrete and continuous change a hybrid system seems feasible, in which an event calculus style mechanism (perhaps using logic programming) determines the instantaneous changes in system configuration at times (identified by the *Next* function) when actions (events) occur, and where a QSIM style algorithm determines the possible evolutions of the continuously varying aspects of the domain between such time points. The interaction between these two components would be in both directions – actions could impose new and remove existing qualitative constraints between parameters, and achievement of landmark values by particular parameters could trigger particular action occurrences.

APPENDIX B – AN EXAMPLE OUTLINE DERIVATION FROM $CIRC_{CEC}[\mathcal{D}_R]$

This appendix gives an outline derivation of the sentence

$$Happens(ChangeSpeed(0), T_2 + \frac{1}{A^+} \arccos(1 - \frac{A^+ . D_W}{S^+})).$$

from $CIRC_{CEC}[\mathcal{D}_R]$ – in other words, a derivation of the fact that the robot stops moving (because it bumps into the wall) at time T_B.

The derivation is given in several stages. Sentences given parenthesised ( ) labels are (domain-independent or domain-dependent) axioms or are direct consequences of Propositions 1, 2 or 3, whereas sentences given square-bracketed [ ] labels are derived.

As regards the topic of this paper, the key steps in the derivation below are [iii] to [iv], [xii] to [xiii], [xvi] to [xvii], [ii(a)] to [iii(a)], and [xxvi] to [xxvii]. These are inferences of expressions of the form "$t \in I \rightarrow Value(P,t) = \int f(t)dt$" from expressions of the form "$t \in I \rightarrow Value(\delta(P),t) = f(t)$". Such derivation steps are only legitimate if it is first shown (using axiom (EC9)) that P is differentiable over the time interval I in question. It is only in this case that axiom (A2) ensures that "$\delta(P)$" is the "derivative of P" in the standard mathematical sense (as well as purely syntactically).

In derivation steps [iv], [xiii], [xvii], [iii(a)] and [xxvii], terms of the form "$\int f(t)dt$" are written after they have been symbolically integrated[2], and so contain existentially quantified variables representing "arbitrary constants of integration". These are assigned specific values in steps [v], [xv], [xix], [v(a)] and [xxix], using information about the particular parameters' values at the greatest lower bounds of the intervals in question, together with the information (derived using (EC8)) that these parameters are continuous at these greatest lower bounds.

Steps [ii], [xi], [xii], [ii(a)] and [xxvi] are derived using the "basic" event calculus of axioms (EC1)–(EC6), and so details are omitted.

[2]More precisely, "$t \in I \rightarrow Value(P,t) = \int f(t)dt$" can be written

$$t, t_1 \in I \rightarrow Value(P,t) = Value(P,t_1) + \int_{t_1}^{t} f(t)dt$$

It is the lack of information about the exact numerical value of "$Value(P,t_1)$" for any $t_1 \in I$ that leads to "arbitrary constants of integration" when such "indefinite integrals" are symbolically integrated.

In the first stage of the derivation, "completion" sentences are listed for the predicates *Breaks*, *Happens*, *Initiates*, *Terminates* and *Releases*.

Stage 1: Completions of *Breaks*, *Happens*, *Initiates*, *Terminates* and *Releases*

By Proposition 1 and the definition of $CIRC_{CEC}[\mathcal{D}_R]$:

$$\begin{aligned} & Breaks(a,p,t) \leftrightarrow \\ & \quad [\exists p_1[p = \delta(p_1) \wedge Breaks(a,p_1,t)] \vee \\ & \quad \exists r[a = ChangeSpeed(r) \wedge p = \delta(NSP)] \vee \\ & \quad \exists r[a = StartTurn(r) \wedge p = \delta(NSP)] \vee \\ & \quad \exists r[a = ChangeSpeed(r) \wedge p = \delta(EWP)] \vee \\ & \quad \exists r[a = StartTurn(r) \wedge p = \delta(EWP)] \vee \\ & \quad \exists r[a = StartTurn(r) \wedge p = \delta(Bearing)]] \end{aligned} \quad \text{(CB)}$$

By Proposition 2 and the definition of $CIRC_{CEC}[\mathcal{D}_R]$:

$$\begin{aligned} & Happens(a,t) \leftrightarrow \\ & \quad [[a = ChangeSpeed(S^+) \wedge t = T_1] \vee \\ & \quad [a = StartTurn(A^+) \wedge t = T_2] \vee \\ & \quad [a = ChangeSpeed(0) \wedge Value(NSP,t) = D_W \\ & \qquad \wedge Value(\delta(NSP),t) > 0] \vee \\ & \quad [a = StartTurn(0) \wedge Value(NSP,t) = D_W \\ & \qquad \wedge Value(\delta(NSP),t) > 0]] \end{aligned} \quad \text{(CH)}$$

By Proposition 3 and the definition of $CIRC_{CEC}[\mathcal{D}_R]$:

$$\begin{aligned} & Initiates(a,f,t) \leftrightarrow \\ & \quad [\exists r[a = ChangeSpeed(r) \wedge f = Moving(r)] \vee \\ & \quad \exists r[a = StartTurn(r) \wedge f = Turning(r)]] \end{aligned} \quad \text{(CI)}$$

$$\begin{aligned} & Terminates(a,f,t) \leftrightarrow \\ & \quad [\exists r_1,r_2[a = ChangeSpeed(r_1) \wedge \\ & \qquad f = Moving(r_2) \wedge r_1 \neq r_2] \vee \\ & \quad \exists r_1,r_2[a = StartTurn(r_1) \wedge f = Turning(r_2) \wedge r_1 \neq r_2]] \end{aligned} \quad \text{(CT)}$$

$$\neg Releases(a,f,t) \quad \text{(CR)}$$

Stage 2: Derivation of $Next(0) = T_1$ and Values of Parameters, etc. at T_1

By (CH) and (R18):

$$\neg Happens(a,0) \quad \text{[i]}$$

By (R16), (EC1), (EC5), (EC14) and [i]:

$$\begin{aligned} & [0 \leq t \wedge t \leq Next(0)] \rightarrow \\ & \quad [HoldsAt(Moving(0),t) \wedge HoldsAt(Turning(0),t)] \end{aligned} \quad \text{[ii]}$$

By [ii], (R10), (R11) and (R12):

$$\begin{aligned} & [0 \leq t \wedge t \leq Next(0)] \rightarrow [Value(\delta(NSP),t) = 0 \wedge \\ & \quad Value(\delta(EWP),t) = 0 \wedge Value(\delta(Bearing),t) = 0] \end{aligned} \quad \text{[iii]}$$

By (EC14), (EC9) [i], [iii] and integration:

$$\begin{aligned} & \exists r_1, r_2, r_3 \forall t[[0 \leq t \wedge t < Next(0)] \rightarrow [Value(NSP,t) = r_1 \\ & \quad \wedge Value(EWP,t) = r_2 \wedge Value(Bearing,t) = r_3]] \end{aligned} \quad \text{[iv]}$$

By (R18) and [iv]:

$$[0 \leq t \ \wedge \ t < Next(0)] \ \rightarrow \ [Value(NSP, t) = 0 \wedge Value(EWP, t) = 0 \wedge Value(Bearing, t) = 0] \quad \text{[v]}$$

By [v] and (EC7):

$$[Value(NSP, Next(0)) = 0 \wedge Value(EWP, Next(0)) = 0 \wedge Value(Bearing, Next(0)) = 0] \quad \text{[vi]}$$

By (CH) and (EC15):

$$\exists a. Happens(a, Next(0)) \quad \text{[vii]}$$

By (EC14), [vii] and (CH):

$$Next(0) = T_1 \quad \text{[viii]}$$

By [viii] and [vi]:

$$[Value(NSP, T_1) = 0 \wedge Value(EWP, T_1) = 0 \wedge Value(Bearing, T_1) = 0] \quad \text{[ix]}$$

By [ix] and (CH):

$$Happens(a, T_1) \ \leftrightarrow \ a = ChangeSpeed(S^+) \quad \text{[x]}$$

Stage 3: Derivation of $Next(T_1) = T_2$ and Values of Parameters, etc. at T_2

By (R1), (R10), (R11), (EC3), (EC5), (CT), (CR) and [X]:

$$[T_1 < t \ \wedge \ t \leq Next(T_1)] \ \rightarrow [Value(\delta(EWP), t) = S^+ . \cos(Value(Bearing, t)) \wedge Value(\delta(NSP), t) = S^+ . \sin(Value(Bearing, t))] \quad \text{[xi]}$$

By (R16), (EC1), (EC5), (EC14) and [x]:

$$[T_1 < t \ \wedge \ t \leq Next(T_1)] \ \rightarrow \ Value(\delta(Bearing), t) = 0 \quad \text{[xii]}$$

By (EC14), (EC9), [xii] and integration:

$$\exists r \forall t [[T_1 < t \ \wedge \ t < Next(T_1)] \ \rightarrow \ Value(Bearing, t) = r] \quad \text{[xiii]}$$

By (CB) and (R21):

$$\neg Breaks(ChangeSpeed(S^+), Bearing, T_1) \quad \text{[xiv]}$$

By (EC8), [xiv], [x], [xiii] and [ix]:

$$[T_1 < t \ \wedge \ t < Next(T_1)] \ \rightarrow \ Value(Bearing, t) = 0 \quad \text{[xv]}$$

By [xv] and [xi]:

$$[T_1 \leq t \ \wedge \ t \leq Next(T_1)] \ \rightarrow \ [Value(\delta(EWP), t) = S^+ \wedge Value(\delta(NSP), t) = 0] \quad \text{[xvi]}$$

By (EC14), (EC9), [xvi] and integration:

$$\exists r \forall t [[T_1 < t \ \wedge \ t < Next(T_1)] \ \rightarrow \wedge \ [Value(EWP, t) = S^+ . t + r_1 \ \wedge \ Value(NSP, t) = r_2]] \quad \text{[xvii]}$$

By (CB) and (R21):

$$\neg Breaks(ChangeSpeed(S^+), NSP, T_1) \wedge \neg Breaks(ChangeSpeed(S^+), EWP, T_1) \quad \text{[xviii]}$$

By (EC8), [xviii], [x], [xvii] and [ix]:

$$[T_1 < t \ \wedge \ t \leq Next(T_1)] \ \rightarrow [Value(EWP, t) = S^+(t - T_1) \ \wedge \ Value(NSP, t) = 0] \quad \text{[xix]}$$

By (EC7), [xix] and [xv]:

$$Value(EWP, Next(T_1)) = S^+(Next(T_1) - T_1) \wedge Value(NSP, Next(T_1)) = 0 \ \wedge \ Value(Bearing, Next(T_1)) = 0 \quad \text{[xx]}$$

By (CH) and (EC15):

$$\exists a. Happens(a, Next(T_1)) \quad \text{[xxi]}$$

By (EC14), [xxi] and (CH):

$$Next(T_1) = T_2 \quad \text{[xxii]}$$

By [xx] and [xxii]:

$$Value(EWP, T_2) = S^+(T_2 - T_1) \wedge Value(NSP, T_2) = 0 \ \wedge \ Value(Bearing, T_2) = 0 \quad \text{[xxiii]}$$

By [xxiii] and (CH):

$$Happens(a, T_1) \ \leftrightarrow \ a = StartTurn(S^+) \quad \text{[xxiv]}$$

Stage 4: Derivation (by contradiction) of $\exists a, t[Happens(a, t) \ \wedge \ T_2 < t]$

**** beginning of sub-derivation ****

Assumption:

$$\neg \exists a, t[Happens(a, t) \ \wedge \ T_2 < t] \quad \text{(a)}$$

By (EC8), (EC9) and (a):

$$T_2 < t \ \rightarrow \ [Continuous(p, t) \ \wedge \ Differentiable(p, t)] \quad \text{[i(a)]}$$

By (R10), (R11), (R12), (EC3), (EC5), (CI), (CT), (CR), [xxiv] and [x]:

$$T_2 < t \ \rightarrow [Value(\delta(EWP), t) = S^+ . \cos(Value(Bearing, t)) \wedge Value(\delta(NSP), t) = S^+ . \sin(Value(Bearing, t)) \wedge Value(\delta(Bearing), t) = A^+] \quad \text{[ii(a)]}$$

By [i(a)], [ii(a)] and integration:

$$\exists r_1, r_2, r_3 \forall t [T_2 < t \ \rightarrow [Value(EWP, t) = \frac{S^+}{A^+} \sin(A^+ . t + r_1) + r_2 \ \wedge Value(NSP, t) = \frac{-S^+}{A^+} \cos(A^+ . t + r_1) + r_3 \ \wedge Value((Bearing, t) = A^+ . t + r_1]] \quad \text{[iii(a)]}$$

By (CB) and (R21):

$$\neg Breaks(a, NSP, T_2) \ \wedge\ \neg Breaks(a, EWP, T_2) \quad \neg Breaks(a, Bearing, T_2) \qquad [iv(a)]$$

By [iv(a)], [iii(a)], [xxiii] and (EC8):

$$T_2 < t \ \rightarrow \ [Value(EWP, t) = \frac{S^+}{A^+}\sin(A^+.t - A^+.T_2) + S^+.(T_2 - T_1) \ \wedge\ Value(NSP, t) = \frac{-S^+}{A^+}\cos(A^+.t - A^+.T_2) + \frac{S^+}{A^+} \ \wedge\ Value((Bearing, t) = A^+.t - A^+.T_2] \qquad [v(a)]$$

By [v(a)], [ii(a)] and (CH):

$$Happens(ChangeSpeed(0), T_2 + \frac{1}{A^+}\arccos(1 - \frac{A^+.D_W}{S^+})) \qquad [vi(a)]$$

By (a) and [vi(a)]:

$$\bot \qquad [vii(a)]$$

**** end of sub-derivation ****

By sub-derivation leading from (a) to [vii(a)] above:

$$\exists a, t[Happens(a, t) \ \wedge\ T_2 < t] \qquad [xxv]$$

Stage 5: Derivation of $Happens(ChangeSpeed(0), T_2 + \frac{1}{A^+}\arccos(1 - \frac{A^+.D_W}{S^+}))$

By (R10), (R11), (R12), (EC3), (EC5), (CI), (CT), (CR), [xxiv] and [x]:

$$[T_2 < t \ \wedge\ t \leq Next(T_2)] \ \rightarrow \ [Value(\delta(EWP), t) = S^+.\cos(Value(Bearing, t)) \ \wedge\ Value(\delta(NSP), t) = S^+.\sin(Value(Bearing, t)) \ \wedge\ Value(\delta(Bearing), t) = A^+] \qquad [xxvi]$$

By (EC9), (EC14), [xxvi] and integration:

$$\exists r_1, r_2, r_3 \forall t[[T_2 < t \ \wedge\ t < Next(T_2)] \ \rightarrow \ [Value(EWP, t) = \frac{S^+}{A^+}\sin(A^+.t + r_1) + r_2 \ \wedge\ Value(NSP, t) = \frac{-S^+}{A^+}\cos(A^+.t + r_1) + r_3 \ \wedge\ Value((Bearing, t) = A^+.t + r_1]] \qquad [xxvii]$$

By (CB) and (R21):

$$\neg Breaks(a, NSP, T_2) \ \wedge\ \neg Breaks(a, EWP, T_2) \ \wedge\ \neg Breaks(a, Bearing, T_2) \qquad [xxviii]$$

By [xxviii], [xxvii], [xxiii] and (EC8):

$$[T_2 < t \ \wedge\ t < Next(T_2)] \ \rightarrow \ [Value(EWP, t) = \frac{S^+}{A^+}\sin(A^+.t - A^+.T_2) + S^+.(T_2 - T_1) \ \wedge\ Value(NSP, t) = \frac{-S^+}{A^+}\cos(A^+.t - A^+.T_2) + \frac{S^+}{A^+} \ \wedge\ Value((Bearing, t) = A^+.t - A^+.T_2] \qquad [xxix]$$

By [xxv], (EC15) and (CH):

$$Happens(ChangeSpeed(0), Next(T_2)) \qquad [xxx]$$

By [xxix] and (EC7):

$$Value(NSP, Next(T_2)) = \frac{-S^+}{A^+}\cos(A^+.Next(T_2) - A^+.T_2) + \frac{S^+}{A^+} \qquad [xxxi]$$

By [xxx] and (CH):

$$Value(NSP, Next(T_2)) = D_W \qquad [xxxii]$$

By [xxx], [xxxi] and [xxxii]:

$$Happens(ChangeSpeed(0), T_2 + \frac{1}{A^+}\arccos(1 - \frac{A^+.D_W}{S^+})) \qquad [xxxiii]$$