Completeness of the Accumulation Calculus

Yu Huiqun, Song Guoxin

Department of Computer Science and Engineering
East China University of Science and Technology
Shanghai 200237, P.R.China

Sun Yongqiang

Department of Computer Science and Engineering
Jiaotong University, Shanghai 200030, P.R.China

Abstract

The accumulation calculus (AC for short) [6] is an interval based temporal logic to specify and reason about hybrid real-time systems. This paper presents a formal proof system for AC, and proves that the system is complete relative to that of Interval Temporal Logic (ITL for short) on real domain [1].

Keywords: Interval Temporal Logic, the accumulation calculus, real-time systems, completeness.

1 Introduction

Since the landmark paper[5], various types of temporal logics have been developed and then applied to specific computer science areas [3, 2, 4, 7]. Lamport's TLA is the generalization of Pnueli's simple logic by allowing actions as atomic formulas; ITL on real domain reasons about temporal variables, which are functions from intervals to reals: $Intv \rightarrow \mathbf{R}$, where $Intv$ is the set of all closed intervals, including point intervals. The duration calculus is an extension of ITL and real arithmetic, which allows description of finitely variable Boolean functions of time, in terms of their integrals in bounded closed intervals.

AC investigates finitely variable real valued functions of time instead of Boolean valued step functions of time, which find application in specifying and reasoning about 'quantitative' properties of hybrid real-time systems. For example, a home heating system consists of a control system, a gas burner and a room. One performance requirement is that during the period of normal working of the heater, the room temperature should be kept within ε units of the required temperature T_r. Let $T(t)$, $Hin(t)$, $Hout(t)$ be room temperature, rate of heat inflow from gas burner, and rate of heat outflow from the room at time t respectively. The physical law which governs the room temperature is

$$T(e) \;=\; T(b) \;+\; \int_b^e Hin(t)\; dt \;-\; \int_b^e Hout(t)\; dt$$

By introducing the notion of sampling in AC, the performance requirement can be simply specified [6].

To reason out such kind of requirements, a formal deduction system for AC is needed. The aim of this paper is to present a formal proof system for AC, and prove that the proof system of AC is complete relative to that of ITL. For detailed presentation of ITL, the reader is referred to [1]. Section 2 presents formal definition of AC; Section 3 gives the proof of relative completeness; Section 4 summaries the paper.

2 The Accumulation Calculus

2.1 Syntax

Accumulation calculus is a modal logic with alphabet of symbols:

- the global variables: x, x_1, x_2, ...
- the state variables: v, v_1, v_2, ...
- the n-ary function symbols: f, f_1, f_2, ...
- the n-ary predicate symbols: p, p_1, p_2, ...
- the connectives: $\neg$, $\vee$, $\wedge$, $\Rightarrow$, $\Leftrightarrow$, ; , $\Diamond$, $\Box$
- the quantifiers: $\forall$, $\exists$
- the truth symbols: *true*, *false*
- special symbols: $\rceil$, $\lceil$, ℓ, $\int$, $+$, $-$, $*$,), (

The sets of global variables, state variables, function symbols and predicate symbols are denoted by X, V, F, and Pr respectively, each of which can be infinite.

Definition 2.1 (States) *The set S of* states *is inductively defined by:*

- *Every constant is a state;*
- *Every state variable is a state;*
- *If r_1, r_2 are states, so are $(r_1 + r_2)$, $(r_1 - r_2)$, and $(r_1 * r_2)$.*

Definition 2.2 (Accumulations) *For any state r , $\int r$ is an* accumulation.

Definition 2.3 (Terms) *The set TERM of* terms *is inductively defined by:*

(1) Accumulations, and global variables are terms;

(2) If $t_1, ..., t_n$ are terms and $f \in F$ is an n-ary function symbol, then $f(t_1, ..., t_n)$ is also a term.

Definition 2.4 (Formulas) *The set WFF of all* (well-formed) formulas *is inductively defined by:*

(1) The truth symbols true and false are formulas;
Every propositional constant from Pr is a formula;
If $t_1, ..., t_n \in TERM$ and $p \in Pr$ is a n-ary predicate symbol, then $p(t_1, ..., t_n)$ is a formula.

(2) If w is a formula, then $(\neg w)$ is a formula;
If w is a formula and $x \in X$, then $(\forall x.w)$ is a formula;
If w_1 and w_2 are formulas, then so are $(w_1 \vee w_2)$ and $(w_1; w_2)$.

Remarks For legibility, conventions in accordance with mathematical logic usage are permitted. In addition, there are following abbreviations:

- $\ell \stackrel{\text{def}}{=} \int 1$
- $\lceil\,\rceil \stackrel{\text{def}}{=} (\ell = 0)$
- $\lceil r = x \rceil \stackrel{\text{def}}{=} \Box(\int r = x * \ell) \wedge (\ell > 0)$
- $\lceil r \rceil \stackrel{\text{def}}{=} \exists x. \lceil r = x \rceil$

2.2 Semantics

Definition 2.5 (Interpretation) *An* interpretation *is a pair I=($\mathbf{R}$, I_0), where $\mathbf{R}$ is the set of real numbers and I_0 is a mapping which assigns*

- *to every constant c$\in$F an element $I_0(c) \in \mathbf{R}$;*
- *to every function symbol $f \in F$ of arity $n \geq 1$ a total function $I_0(f)$:$\mathbf{R}^n \to \mathbf{R}$;*
- *to every propositional constant a$\in$ Pr an element $I_0(a) \in$Bool;*
- *to every predicate symbol p$\in$ Pr of arity $n \geq 1$ a predicate $I_0(p)$:$\mathbf{R}^n \to$Bool.*

We assume standard meanings associated with ordinary function symbols and predicate symbols. In many cases, we use the same symbols for both syntactic entities and semantic ones, i.e. we allow for overload of symbols when it does not cause confusion. For example, '+' is used as both syntactic symbol and semantic symbol, whose meaning (interpretation) is the ordinary arithmetic addition.

Definition 2.6 (Valuation) *A* valuation ϕ *is a functional of signature:*

$$\phi : V \to (\mathbf{R} \to \mathbf{R})$$

We require that each $\phi(v)$ is a step function.

Definition 2.7 (Assignment) *An* assignment γ *is a function giving values to global variables, i.e. γ:X$\to$R. The set of all valuations is denoted by Γ.*

Two assignments γ and γ' are called *x-equivalent* if they associate the same value with any global variable y different from x.

Definition 2.8 (Structure) *A* structure θ *of the calculus is a triple θ=(I,ϕ,γ).*

Semantics of States The meaning of states in an interpretation and a valuation is a functional of signature:

S$\to$ ($\mathbf{R} \to \mathbf{R}$).

The value of state r at time t in ϕ of I (denoted as ϕ_I(r)(t)) is defined by:

$$\phi_I(c)(t) = I_0(c)$$
$$\phi_I(v)(t) = \phi(v)(t)$$
$$\phi_I(r_1 \star r_2)(t) = \phi_I(r_1)(t) \star \phi_I(r_2)(t), \text{ where } \star \in \{+, -, *\}.$$

Semantics of Terms The meaning of terms in structure θ is a functional of signature:

$TERM \rightarrow (Intv \rightarrow \mathbf{R})$,

where $Intv$ is the set of all closed intervals. We use $\theta_{[b,e]}(t)$ to denote the functional value for term t in the structure θ and time interval [b,e]. The semantics of terms is defined by:

$$\begin{array}{l}
\theta_{[b,e]}(x) = \gamma(x) \\
\theta_{[b,e]}(\ell) = e - b \\
\theta_{[b,e]}(\int r) = \int_b^e \phi_I(r)(t)dt \\
\theta_{[b,e]}(f(t_1, ..., t_n)) = I_0(f)(\theta_{[b,e]}(t_1), ..., \theta_{[b,e]}(t_n))
\end{array}$$

Semantics of formulas The meaning of formulas in structure θ is a functional of signature:

WFF$\rightarrow (Intv \rightarrow Bool)$.

We use $\theta_{[b,e]}(w)$ to denote the functional value for formula w in the structure θ and time interval [b,e]. The semantics of formulas is defined by:

$$\begin{array}{l}
\theta_{[b,e]}(true) = tt \\
\theta_{[b,e]}(false) = ff \\
\theta_{[b,e]}(p) = I_0(p) \\
\theta_{[b,e]}(p(t_1, ..., t_n)) = I_0(p)(\theta_{[b,e]}(t_1), ..., \theta_{[b,e]}(t_n)) \\
\theta_{[b,e]}(\neg w) = tt \quad iff \quad \theta_{[b,e]}(w) = ff \\
\theta_{[b,e]}(w_1 \vee w_2) = tt \quad iff \quad \theta_{[b,e]}(w_1) = tt \ or \ \theta_{[b,e]}(w_2) = tt \\
\theta_{[b,e]}(w_1; w_2) = tt \quad iff \quad \theta_{[b,m]}(w_1) = tt \ and \ \theta_{[m,e]}(w_2) = tt, \\
\qquad\qquad for \ some \ m \in [b, e] \\
\theta_{[b,e]}(\forall x.w) = tt \quad iff \quad \theta'_{[b,e]}(w) = tt, \ for \ all \ \theta' = (I, \phi, \gamma'), \\
\qquad\qquad where \ \gamma' \ is \ assignment \ x - equivalent \ to \ \gamma.
\end{array}$$

Definition 2.9 (Valid) *A formula w is called* valid in an valuation ϕ *of I (denoted by* $\phi \models w$*) if* $\theta_{[b,e]}(w)=tt$ *for any time interval [b, e] and for any assignment. A formula is called* valid *(denoted by* $\models w$*) if it is valid in all valuations of I.*

2.3 Proof System

As AC is an extension of ITL, which is an extension of modal logic, which again is an extension of predicate logic of reals, we adopt all axioms and inference rules from these calculi. In addition, there are several axioms and rules which are specific to AC. They are given as follow.

Axioms and Rules We list the axioms and proof rules of AC. Let r range over states. We will list some simple theorems of mathematical analysis, which are sufficiently useful to be taken as axioms in AC.

Axiom 1 $\int c = c * \ell$

Axiom 2 $\int(r_1 + r_2) = \int r_1 + \int r_2$

Axiom 3 $\int(r_1 - r_2) = \int r_1 - \int r_2$

An axiom about the '$*$' operator for states is as follows.

Axiom 4 $\lceil r_1 \rceil \wedge \lceil r_2 \rceil \Rightarrow (\ell * \int(r_1 * r_2) = \int r_1 * \int r_2)$

The basic axiom relating ';' and '$\int$' is that the accumulation of a state in an interval is the sum of its accumulation in its parts.

Axiom 5 $(\int r = x; \int r = y) \Rightarrow \int r = x + y$

There are two induction rules which extend hypotheses over adjacent sub-intervals. These rules are based on the assumption that states have finite variability.

Induction Rules: Let $R(X)$ be an accumulation formula containing the formula letter X, and let r be any state.

- If $\vdash R(\lceil\ \rceil)$ and $R(X) \vdash R(X \vee (\lceil r \rceil; X))$, then $R(true)$.
- If $\vdash R(\lceil\ \rceil)$ and $R(X) \vdash R(X \vee (X; \lceil r \rceil))$, then $R(true)$.

Following the induction rules, we have theorem:

Theorem 2.10 *For any state r,*

$$\lceil\ \rceil \vee (\lceil r \rceil; true),$$
and $$\lceil\ \rceil \vee (true; \lceil r \rceil)\ .$$

3 Completeness of the Accumulation Calculus

Before proving completeness of AC, we give the soundness theorem, some definitions and lemmas.

Theorem 3.1 (Soundness) *If* $\vdash W$ *then* $\models W$

The proof of soundness is by induction on the axioms and rules, which is quite conventional and is omitted. Now we sketch out the proof of relative completeness of AC. The main idea is to prove that for each AC formula W we find a corresponding ITL formula F_W such that $\models W\ iff\ \models F_W$.

Definition 3.2 (Subexpression) *u is a* subexpression *of a state w if*

- $u = w$, *or*
- $w = w_1 \star w_2$ *and u is a subexpression of* w_1 *or of* w_2, *where* $\star \in \{+, -, *\}$.

Let an arbitrary accumulation formula W be given. Let Sub denote the set of all the subexpressions of states occurring in W. We select $|Sub|$ many temporal variables and put them in one-to-one correspondence with these subexpressions [1]. We can index the selected variables with these subexpressions. A finite set of interval formulas is then constructed as follows.

$$\begin{aligned}
H_1 &\stackrel{\text{def}}{=} \{v_{[c]} \ = \ c * \ell \mid c \in Sub\} \\
H_2 &\stackrel{\text{def}}{=} \{v_{[r_1+r_2]} \ = \ v_{[r_1]} + v_{[r_2]} \mid (r_1 + r_2) \in Sub\} \\
H_3 &\stackrel{\text{def}}{=} \{v_{[r_1-r_2]} \ = \ v_{[r_1]} - v_{[r_2]} \mid (r_1 - r_2) \in Sub\} \\
H_4 &\stackrel{\text{def}}{=} \{\lceil v_{[r_1]} \rceil \wedge \lceil v_{[r_2]} \rceil \Rightarrow (\ell * v_{[r_1 * r_2]} \ = \ v_{[r_1]} * v_{[r_2]}) \mid (r_1 * r_2) \in Sub\} \\
H_5 &\stackrel{\text{def}}{=} \{\forall x, y.(v_{[r]} = x; v_{[r]} = y \Rightarrow v_{[r]} = x + y) \mid r \in Sub\} \\
H_6 &\stackrel{\text{def}}{=} \{\lceil\ \rceil \vee (\lceil v_{[r]} \rceil; true) \mid r \in Sub\} \\
H_7 &\stackrel{\text{def}}{=} \{\lceil\ \rceil \vee (true; \lceil v_{[r]} \rceil) \mid r \in Sub\}
\end{aligned}$$

[1] $|Sub|$ denotes the cardinality of the set Sub, which is obviously finite.

where we define $\lceil v_{[r]} \rceil$ by $\exists x. \Box (v_{[r]} = x * \ell) \land \ell > 0$.

Define H to be conjunction of all the formulas in H_1 to H_7, and let F be the interval formula obtained from W by replacing each accumulation $\int r$ with $v_{[r]}$. Define F_W to be $\Box H \Rightarrow F$. In the following we assume I is the interpretation given as in Section 2.2.

Definition 3.3 (H-pair) *We call* $(\phi, [b,e])$ *an H-pair in* I *if*

$$\theta_{[c,d]}(\Box H) = tt,$$

for every assignment γ *and subinterval* $[c,d]$ *of* $[b,e]$*, where* $\theta = (I, \phi, \gamma)$*.*

Lemma 3.4 *Given an H-pair* $(\phi, [b,e])$ *in* I *. For any subexpression* $r \in Sub$ *and time interval* $[b,e]$*,* $b < e$*, there is a finite partition* $b = t_0 < t_1 < \ \dots \ < t_n = e$ *of* $[b,e]$ *such that*

$\theta_{[t_{i-1},t_i]}(\lceil v_{[r]} \rceil) \ = \ tt$*, for every assignment* γ*,* $i = 1, \ \dots \ , n$*.*
where $\theta = (I, \phi, \gamma)$*.*

Proof: For any $t : \ b < t < e$, there are (by H_6 and H_7) t' and t" such that $b \le t' < t < t" \le e$ and

$$\left.\begin{array}{rcl} \theta_{[t',t]}(\lceil v_{[r]} \rceil) & = & tt \\ \theta_{[t,t"]}(\lceil v_{[r]} \rceil) & = & tt \end{array}\right\} \qquad (\dagger)$$

Thus there is an open interval $(t', t")$ covering t (but not b and e) such that the closed interval $[t', t"]$ has the property $(\dagger)$.

For the left end point b, there is (by H_6) a t" such that $b < t" \le e$ and

$\theta_{[b,t"]}(\lceil v_{[r]} \rceil) \ = \ tt \qquad (\dagger_b)$

Thus there is an open interval $(t', t")$ covering b such that the closed interval $[b, t"]$ has the above property $(\dagger_b)$. (Select arbitrary $t' < b$.)

Similarly for e there is (by H_7) a t' such that $b \le t' < e$ and

$\theta_{[t',e]}(\lceil v_{[r]} \rceil) \ = \ tt \qquad (\dagger_e)$

Thus there is an open interval $(t', t")$ covering e such that the closed interval $[t', e]$ has the above property $(\dagger_e)$. (Select arbitrary $t" > e$)

So we have an infinite collection of open intervals covering the closed and bounded interval $[b,e]$. Then by Heine-Borels theorem there is a finite sub-collection $C = \{I_1, \ .. \ I_m\}$ of the open intervals covering $[b,e]$.

STEP 1: Select the open interval $I_i = (a_i, b_i)$ from C covering b. Then the closed interval $[b, b_i]$ satisfies by $(\dagger_b)$: $\theta_{[b,b_i]}(\lceil v_{[r]} \rceil) \ = \ tt$

STEP 2: If $b_i = e$ we have proved the lemma.

Otherwise $b_i < e$. Select an open interval $I_j = (a_j, b_j)$ from C covering b_i.

If $e < b_j$, then by $(\dagger_e)$ the closed interval $[b_i, e]$ satisfies: $\theta_{[b_i,e]}(\lceil v_{[r]} \rceil) \ = \ tt$ and the lemma is proved.

If $b_j \le e$, then the closed interval $[b_i, b_j]$ will by $(\dagger)$ satisfy one of the followings

1. $\theta_{[b_i,b_j]}(\lceil v_{[r]} \rceil) \ = \ tt$
2. $\theta_{[b_i,m]}(\lceil v_{[r]} \rceil) \ = \ tt$ and $\theta_{[m,b_j]}(\lceil v_{[r]} \rceil) \ = \ tt$, for some $m : \ b_i < m < b_j$

Repeat now STEP 2, until the required partition of $[b,e]$ is achieved. This terminates since there is a finite number m of the open intervals in C.

$|\Box$

Lemma 3.5 *Given an H-pair* $(\phi, [b,e])$ *in* I. *There is a valuation* ϕ' *such that for any subexpression* $r \in Sub$ *and time interval* $[b,e]$, $b < e$, *there is a finite partition* $b = t_0 < t_1 < \ \dots\ t_n = e$ *of* $[b,e]$ *such that*

$$\theta_{[t_{i-1},t_i]}(\lceil v_{[r]} \rceil) \ = \ tt, \textit{ for every assignment } \gamma,\ i = 1,\ \dots\ , n.$$

and, for any $t \in [b,e]$, *there is an* $i \in \{1,\ \dots\ n\}$ *such that* $t_{i-1} \leq t < t_i$ *and*

$$\phi'_I(r)(t) \ = \ x, \ \textit{if} \ \theta_{[t_{i-1},t_i]}(v_{[r]}) = x * (t_i - t_{i-1}).$$

where $\theta = (I, \phi, \gamma)$.

Proof: Define a valuation ϕ' as follows: For any state variable $Y \notin Sub$ let $\phi'(Y)(t) = 0$ for $t \in \mathbf{R}$. For any state variable $X \in Sub$, there is by lemma 3.4 a finite partition $b = t_0 < t_1 < \ \dots\ < t_n = e$ of $[b,e]$ such that

$$\theta_{[t_{i-1},t_i]}(\lceil v_{[X]} \rceil) \ = \ tt, \text{ for } i = 1,\ \dots\ , n.$$

where $\theta = (I, \phi, \gamma)$. Let

$$\phi'(X)(t) \ = \ x \text{ if } t_{i-1} < t < t_i \text{ and } \theta_{[t_{i-1},t_i]}(v_{[X]}) = x * (t_i - t_{i-1}), \text{ for some } i \in \{1,\ \dots\ , n\}.$$

Each such function has only a finite number of discontinuity points in any time interval, so ϕ' is a valuation.

We prove the remain parts of the lemma by structural induction on $r \in Sub$. If $r \notin Sub$ there is nothing to prove, so assume below that $r \in Sub$. The cases when r is constant c or state variable X are trivial, so consider the following cases.

- CASE 1: r has the form $r_1 + r_2$.
  By combination of the induction hypotheses for r_1 and r_2 , we get a finite partition $b = t_0 < t_1 < \ \dots\ t_n$ of $[b,e]$ and in each $[t_{i-1}, t_i]$ there exist x and y such that
  $$\begin{aligned} \theta_{[t_{i-1},t_i]}(v_{[r_1]}) &= x * (t_i - t_{i-1}), \\ \theta_{[t_{i-1},t_i]}(v_{[r_2]}) &= y * (t_i - t_{i-1}), \\ \phi'(r_1)(t) &= x, \\ \phi'(r_2)(t) &= y, \end{aligned}$$
  where $t \in [t_{i-1}, t_i]$.
  By H_2,
  $\theta_{[t_{i-1},t_i]}(v_{[r]})$
  $= \theta_{[t_{i-1},t_i]}(v_{[r_1]}) + \theta_{[t_{i-1},t_i]}(v_{[r_2]})$
  $= (x + y) * (t_i - t_{i-1})$
  and
  $\phi'(r)(t) = \phi'(r_1 + r_2)(t) = \phi'(r_1)(t) + \phi'(r_2)(t) = x + y$

- CASE 2: r has the form $r_1 - r_2$.
  Similar to CASE 1.

- CASE 3: r has the form $r_1 * r_2$.
  Similar to CASE 1 and CASE 2 .

$|\square$

Lemma 3.6 *For a given H-pair* $(\phi, [b,e])$ *in* I, *let* ϕ' *be the valuation given by lemma 3.5. Then any assignment* γ *in* ϕ' *satisfies*

$$\theta'_{[c,d]}(\textstyle\int r) \ = \ \phi_{[c,d]}(v_{[r]})$$

where $\theta' = (I, \phi', \gamma)$, $r \in Sub$ *and* $[c,d]$ *is a sub-interval of* $[b,e]$.

Proof: Suppose $c = d$. Then $\theta'_{[c,d]}(\int r) = 0$. From H_1, we have $\phi_{[c,d]}(v_{[r]}) = x*(d-c) = 0$. The lemma is proved.

Suppose $c < d$. Since $(\gamma, [b, e])$ is an H-pair in ϕ, so is $(\gamma, [c, d])$. By lemma 3.5, there is a finite partition $c = t_0 < t_1 < \ \dots\ < t_n = d$ of $[c, d]$ such that

$\theta_{[t_{i-1},t_i]}(\lceil v_{[r]} \rceil) = tt$, for $i = 1, \ \dots\ , n$.

and, for any $t \in [c, d]$, there is an $i \in \{1, \ \dots\ n\}$ such that $t_{i-1} \le t < t_i$ and

$\phi'_I(r)(t) = x, \ if\ \theta_{[t_{i-1},t_i]}(v_{[r]}) = x * (t_i - t_{i-1})$.

Thus $\int_{t_{i-1}}^{t_i} \phi'_I(r)(t)dt = \phi_{[t_{i-1},t_i]}(v_{[r]})$, for $i = 1, \ \dots\ , n$, and by H_5:

$$\theta'_{[c,d]}(\int r) = \sum_{i=1}^{n} \phi_{[t_{i-1},t_i]}(v_{[r]}) = \phi_{[c,d]}(v_{[r]})$$

$|\Box$

Lemma 3.7 $\models W \ \ iff \ \ \models \Box H \Rightarrow F$

Proof: We first prove that $\models W$ implies $\models \Box H \Rightarrow F$. Suppose $\not\models \Box H \Rightarrow F$, i.e. there is an H-pair $(\phi, [b, e])$ in I such that $\theta_{[b,e]}(F) = ff$. By lemma 3.6, there is a valuation ϕ' and an assignment γ' such that for any $r \in Sub$

$$\theta'_{[c,d]}(\int r) = \phi_{[c,d]}(v_{[r]}),$$

for any sub-interval $[c, d]$ of $[b, e]$, where $\theta' = (I, \phi', \gamma')$. Since $\theta_{[b,e]}(F) = ff$ we have $\theta_{[b,e]}(W) = ff$ and hence $\not\models W$.

We now prove that $\models \Box H \Rightarrow F$ implies $\models W$. Suppose $\not\models W$, i.e. there is a valuation ϕ and an assignment γ such that $\theta_{[b,e]}(W) = ff$, where $\theta = (I, \phi, \gamma)$. Construct a valuation ϕ' such that $\phi'_{[c,d]}(v_{[r]}) = \theta_{[c,d]}(\int r)$ for all $r \in Sub$ and interval $[c, d]$. By construction, $\theta'_{[b,e]}(F) = ff$, where $\theta' = (I, \phi', \gamma)$, and from the soundness theorem of AC, we have $\theta'_{[b,e]}(\Box H) = tt$. So $\not\models \Box H \Rightarrow F$.
$|\Box$

The following completeness theorem we name *relative completeness* since we assume that valid interval formulas of the form $(\Box H) \Rightarrow F$ are provable. That is, we assume a new rule for the calculus:

> If F is a valid interval formula, in which only the temporal variables: $v_{[r_1]}, \ \dots\ v_{[r_n]}$ occur, we take
>
> W
>
> as an axiom, where W is obtained from F by replace each temporal variable $v_{[r_i]}$ with an accumulation $\int r_i$.

Theorem 3.8 (Relative completeness) $\models W$ *implies* $\vdash W$.

Proof: Suppose $\models W$. By lemma 3.7 we get $\models \Box H \Rightarrow F$. Let H' be obtained from H by replacing each $v_{[r]}$ by $\int r$. Thus H' is a conjunction of axioms of AC. By the $\Box$-generalization

of modal logic, we have $\vdash \Box H'$. By the above assumption we have $\vdash \Box H' \Rightarrow W$. Therefore by modus ponens $\vdash W$.
$|\Box$

4 Summary

AC is an interval based temporal logic introduced in [6] to specify and reason about hybrid real-time systems. This paper presents a formal proof system for AC, and proves that the system is complete relative to that of ITL on real domain.

Acknowledgements: The authors are grateful to Zhou Chaochen and Shen Enshao for their valuable suggestions, and the referees for excellent comments.

References

[1] M.R. Hansen and Zhou Chaochen. Semantics and completeness of duration calculus. In J.W. de Bakker, C. Huizing, W.P. de Roever, and G. Rozenberg, editors, *Real-time:Theory in Practice, LNCS 600*, pages 209–225. Springer-Verlag, 1992.

[2] L. Lamport. Temporal Logic of Actions. Technical Report 79, Systems Research Center of DEC, USA, 1991.

[3] Z. Manna and A. Pnueli. Models for reactivity. *Acta Informatica*, 30:609–678, 1993.

[4] B. Moszkowski. A temporal logic for multi-level reasoning about hardware. *IEEE Computer*, 18(2):10–19, 1985.

[5] A. Pnueli. The temporal logic of programs. In *18th Ann. IEEE Symp. on Foundations of Computer Science*, pages 46–57, 1977.

[6] Yu Huiqun, P.K. Pandya, and Sun Yongqiang. A calculus for hybrid sampled data systems. In H. Langmaack, W.-P. de Rover, and J. Vytopil, editors, *Formal Techniques in Real-Time and Fault-Tolerant Systems, LNCS 863*, pages 716–737. Springer Verlag, 1994.

[7] Zhou Chaochen, C.A.R. Hoare, and A.P. Ravn. A calculus of durations. *Information Processing Letters*, 40(5):269–276, 1991.

Yu Huiqun received the Ph.D. degree in computer science from Shanghai Jiaotong University. He is currently a Lecturer in the Department of Computer Science and Engineering at East China University of Science and Technology at Shanghai. His areas of interest are formal methods, concurrency, reliable real-time systems, semantics of programming languages.

Song Guoxin is a Professor in the Department of Computer Science and Engineering at East China University of Science and Technology at Shanghai. His current interests are formal methods, object-oriented programming and parallel engineering.

Sun Yongqiang is a Professor in the Department of Computer Science and Engineering at Shanghai Jiaotong University. His current interests are formal technique of software development, new paradigm of programming languages, and parallel processing.